Improved Integrality Gap Upper Bounds
for TSP with Distances One and Two

Matthias Mnich†  Tobias Mömke‡

December 10, 2013

Abstract

We study the structure of solutions to linear programming formulations for the traveling
salesperson problem (TSP).

We perform a detailed analysis of the support of the subtour elimination linear program-
ming relaxation, which leads to algorithms that find 2-matchings with few components in
polynomial time. The number of components directly leads to integrality gap upper bounds
for the TSP with distances one and two, for both undirected and directed graphs.

Our main results for fractionally Hamiltonian instances are:

• For undirected instances we obtain an integrality gap upper bound of 5/4 without
  any further restrictions, of 7/6 if the optimal LP solution is half-integral, and of 10/9
  if there is an optimal solution that is a basic solution of the fractional 2-matching
  polytope.

• For directed instances we obtain an integrality gap upper bound of 3/2, and of 4/3 if
  given an optimal 1/2-integral solution.

Additionally, we show that relying on the structure of the support is not an artefact of our
algorithm, but is necessary under standard complexity-theoretic assumptions: we show that
finding improved solutions via local search is $W[1]$-hard for $k$-edge change neighborhoods
even for the TSP with distances one and two, which strengthens a result of Dániel Marx.

Keywords. Integrality gap, traveling salesperson problem, subtour elimination.

1 Introduction

The traveling salesperson problem (TSP) in metric graphs is one of the most fundamental
NP-hard optimization problems. Given an undirected or directed graph $G$ with a metric on its
edges, we seek a tour $T$ (Hamiltonian cycle) of minimum cost in $G$, where the cost of $T$ is the
sum of costs of edges traversed by $T$.

Despite a vast body of research, the best approximation algorithm for metric TSP is still
Christofides’ algorithm [9] from 1976, which has a performance guarantee of 3/2. Recall that
the performance guarantee or approximation ratio of an algorithm for a problem is defined as
a number $\alpha$ such that, in polynomial time, the algorithm computes a solution whose value is
within a factor $\alpha$ of the optimal value. Generally the bound 3/2 is not believed to be tight.
However, the currently largest known lower bound on the performance guarantee obtainable in
polynomial time is as low as 123/122 [13].

*The research was partially funded by Deutsche Forschungsgemeinschaft grant BL511/10-1.
†Cluster of Excellence, Saarbrücken, Germany. m.mnich@mmci.uni-saarland.de
‡Saarland University, Saarbrücken, Germany. moemke@cs.uni-saarland.de
One of the most promising techniques to obtain an improved performance guarantee is to use a linear programming (LP) formulation of TSP. Upper bounds on the integrality gap of the LP usually translate to approximation guarantees. In this context, the subtour elimination relaxation (SER), or Held-Karp relaxation [11], is particularly important. Its integrality gap is between $4/3$ and $3/2$, and the value $4/3$ is conjectured to be tight [10]. For relevant special cases, the conjecture is known to be true [6][16]. It is also known that SER has a close relation to 2-matchings, as was pointed out for instance by Schalekamp et al. [19] in the context of perfect 2-matchings. We write that the approximation ratio of an algorithm is $\alpha$ with respect to SER to express that the result provides both an integrality gap upper bound and an approximation ratio of $\alpha$.

### 1.1 Our Contributions

We investigate the structure of the support graph of solutions to SER, that is, the graph of edges with non-zero value in an optimal solution to SER. Our structural insights allow us to give an approximation algorithm for TSP variants that finds a 2-matching (i.e., a collection of paths and cycles) in the SER support graph that aims to minimize the number of components for both directed and undirected instances. While we consider our structural findings to be valuable by themselves, they have a direct impact on the integrality gap of SER for several TSP variants. In particular, we obtain improved integrality gap upper bounds for the asymmetric and symmetric TSP with distances one and two, a classical and well-studied variant of TSP [2, 4, 5, 14, 17, 18, 20, 21]. We refer to the symmetric variant (in undirected graphs) as (1, 2)-STSP and to the asymmetric variant as (1, 2)-ATSP.

We first, in Sect. 2, prove that from an algorithmic point of view we can restrict attention to fractionally Hamiltonian instances of (1, 2)-STSP and (1, 2)-ATSP; that is, we may assume that if the instance has order $n$, there is a fractional solution to SER of cost $n$. This extends a result of Qian et al. [18]. For integrality gap upper bounds of SER with respect to (1, 2)-STSP and (1, 2)-ATSP, we show that we can restrict our focus to fractionally Hamiltonian instances, if there are are arbitrarily large integrality gap instances. We therefore conjecture that analyzing fractionally Hamiltonian instances is sufficient to obtain integrality upper bounds for general instances. Our conjecture is backed by an infinite class of instances for (1, 2)-STSP with integrality gap $10/9$ obtained by Williamson [21]. We also present an infinite class of (1, 2)-ATSP instances with integrality gap $6/5$.

With the restriction to fractionally Hamiltonian instances, it suffices to find a 2-matching with few components in the support graph of an optimal SER solution to derive integrality gap upper bounds for TSP. To this end, we define certain types of improvements that extend the improvements used by Berman and Karpinski [2] in their approximation algorithm for (1, 2)-STSP.

In Sect. 3 we show that we always can transform a 2-matching in the support to another 2-matching that does not contain isolated vertices without increasing the number of components. The proof is based on directed alternating paths and a novel flow formulation, where the flow conservation is ensured due to the constraints of SER. Based on this result and further improvements based on alternating paths, in Sect. 6 we find in polynomial time a 2-matching with at most $n/4$ components, which implies an integrality gap upper bound of $5/4$ for (1, 2)-STSP. The proof assigns LP values to components such that each component collects a total value of at least 4.

Next, we consider the half-integrality conjecture. The conjecture says that it suffices to consider half-integral instances of SER to derive integrality gap upper bounds for general instances. It is implied by a recent conjecture of Schalekamp et al. [19], that there are integrality gap instances of SER that at the same time are basic solutions to the fractional perfect 2-matching polytope. These basic solutions are relatively well understood and they have a quite specific structure [1]. In particular, they are half-integral (all LP values are multiples of 1/2)
and they are subcubic (all degrees in the support are at most three). We show that if the half-integrality conjecture is true and at the same time the integrality gap instances are fractionally Hamiltonian, the integrality gap is at most $7/6$.

The conjecture of Schalekamp et al. [19] motivates to consider instances of $(1,2)$-STSP that are fractionally Hamiltonian and admit subcubic supports of optimal basic solutions. For those, in Sect. 7, we prove a tight integrality gap upper bound of $10/9$. These results are orthogonal to those of Schalekamp et al. [19], but as mentioned above we have a certain believe that fractionally Hamiltonian instances capture the essence of the problem.

Our preparation for $(1,2)$-STSP translates to $(1,2)$-ATSP in a natural way. We obtain, in Sect. 8, an integrality gap upper bound of $3/2$ for general fractionally Hamiltonian instances, and $4/3$ if additionally there is a half-integral optimal solution.

In Sect. 9 we strengthen a result of Marx [15] about finding cheaper solutions to a $(1,2)$-STSP instance by local search. Precisely, we show that it is $W[1]$-hard to find an improved solution compared to a given tour by changing at most $k$ edges, even in undirected TSP instances with distances one and two. This hardness result says that a brute force search of all subsets of $k$ edges—in time $n^{O(k)}$ for $(1,2)$-STSP instances of order $n$—is essentially optimal, unless many canonical NP-complete problems admit subexponential-time algorithms. Such an intractability result was known before only for TSP instances with three distinct city distances, due to Marx [15]. This result suggests that a simple search for local improvements is not efficient, and therefore that it is essential to analyze the specific structure of the support. Our proof is very similar to that of Marx, with some simplifications.

### 1.2 Overview of Techniques

To show that the subtour elimination support graph contains a 2-matching with few components, we apply a sequence of local improvements. One important step is that we can exclude the solution computed by our algorithm to contain isolated vertices. We use an induction that creates a tree of alternating paths, and show by a flow argument that it is always possible to increase the size of the tree unless there is an improved 2-matching (i.e., with fewer components).

All five of our integrality gap upper bound results use an accounting technique that distributes an amount of $n$ coins to components of the 2-matching. We ensure that for each component, there are sufficiently many coins and therefore the total number of components cannot exceed the aimed-for upper bound. However, we employ two entirely different schemes in order to provide a distribution of coins. Our result for degree-3-bounded support graphs (the proof of Theorem 3) initially distributes the coins to the vertices, that is, there is one coin for each vertex. Then it redistributes the coins to the components. A similar accounting technique has been used by Berman and Karpinski [2]. However, we exploit properties of of the subtour elimination constraints in order to ensure the existence of fractional coins that are available provided that no local improvements are possible.

The remaining integrality gap upper bound results use the LP values of the subtour elimination relaxation directly. One way to see the technique is to initially distribute $n$ coins to the edges, where each edge obtains a fraction of the coin according to its LP value. We formulate the distribution of values to components by using a new linear program. The linear program takes the SER solution $x^*$ and the aimed-for number of coins per component as parameter and this way we reduce the analysis to finding a feasible solution to the linear program. In order to find a feasible solution to the new linear program, we split the edges of the SER support graph into sub-edges such that each sub-edge $e$ has the same LP value $x^*_e$. In the resulting multigraph, we obtain a collection of disjoint alternating paths and use that certain types of alternating paths lead to improved 2-matchings.
1.3 Related Work

Both (1, 2)-STSP and (1, 2)-ATSP are well-studied from the approximation point of view. For (1, 2)-STSP, it is NP-hard to obtain a performance guarantee better than 535/534 [14]. On the positive side, Papadimitriou and Yannakakis [17] presented an approximation algorithm for (1, 2)-STSP with performance ratio 7/6. This was improved by Bläser and Ram [5] to a 65/56-approximation algorithm. The currently best approximation factor is 8/7, due Berman and Karpinski [2].

The best known integrality gap lower bound of (1, 2)-STSP is 10/9, due to Williamson [21]. Recently, Qian et al. [18] showed an integrality gap upper bound of 19/15 for (1, 2)-STSP, and of 7/6 if the integrality gap is attained by a basic solution of the fractional 2-matching polytope. With the additional assumption that a certain type of modification maintains the 2-vertex connectedness of the support graph, they were able to show a tight integrality gap of 10/9.

For (1, 2)-ATSP, it is NP-hard to obtain a performance ratio better than 207/206 [14]. The first non-trivial approximation algorithm for (1, 2)-ATSP was given by Vishwanathan [20], with an approximation factor of 17/12. This was improved to 4/3 by Bläser and Manthey [4]. The currently best approximation factor is 5/4, and is due to Bläser [3].

2 Preparatory Work

2.1 Notation

Let us start with some notation that will be used throughout the paper. For a graph $G$, let $V(G)$ denote its set of vertices and $E(G)$ its set of edges (resp. arcs, if $G$ is directed). For vertex sets $S, S' \subseteq V(G)$, define $\delta(G)(S, S') := \{(v, w) : v \in S, w \in S'\}$ as the set of edges between $S$ and $S'$. We use $\delta(G)(S) := \delta(G)(S, V(G) \setminus S)$. For a single vertex $v$, we write $\delta(G)(v)$ instead of $\delta(G)(\{v\})$. We define $\mathcal{N}_G(v) = \{u \in V(G) : \{u, v\} \in \delta(G)(v)\}$ to be the neighborhood of $v$. For each vertex $v \in V(G)$, let $\deg_G(v) := |\mathcal{N}_G(v)|$ be its degree in $G$.

In directed graphs, we sometimes need to distinguish between arcs leaving a set of vertices and those entering the set, which we mark by the superscript $+$ or $-$. For a digraph $G$ with possibly bidirected arcs but no loops, we define $\delta(G)(S, S') := \{(v, w), (w, v) \in E(G) : v \in S, w \in S'\}$, $\delta^+(G)(S, S') := \{(v, w) \in E(G) : v \in S, w \in S'\}$, and $\delta^-(G)(S, S') := \{(w, v) \in E(G) : v \in S, w \in S'\}$. Analogous to the undirected case, we define $\deg^+_G(S) := \delta^+(G)(S, V(G) \setminus S)$ and $\deg^-_G(S) := \delta^-(G)(S, V(G) \setminus S)$. We define $\mathcal{N}^+_G(v) = \{u \in V(G) : (u, v) \in \delta^+(G)(v)\}$ or $(v, u) \in \delta^+_G(v)$, $\mathcal{N}^-_G(v) = \{u \in V(G) : (v, u) \in \delta^+_G(v)\}$, and $\mathcal{N}^-_G(v) = \{u \in V(G) : (u, v) \in \delta^-_G(v)\}$. For each vertex $v \in V(G)$, let $\deg^-_G(v) := |\mathcal{N}^-_G(v)|$ and $\deg^+_G(v) := |\mathcal{N}^+_G(v)|$ be its in-degree and out-degree, and let $\deg_G(v) = \deg^+_G(v) + \deg^-_G(v)$ be its degree.

2.2 Subtour Elimination Linear Program

For an instance $G$ of (1, 2)-STSP, we define a linear programming relaxation of an integer linear program that models an optimal TSP tour in $G$.

Given an undirected graph $G$, we introduce one variable $x_e$ for each edge $e \in E(G)$. The variable $x_e$ models whether edge $e$ belongs to the TSP tour ($x_e = 1$) or not ($x_e = 0$). We then
Lemma 1. Let \( G \) contain many cost two edges. Hence we modify Proposition 1 as follows:

We refer to the first set \( (1) \) of constraints as the equality constraints, and to the second set \( (2) \) as the subtour elimination constraints. The equality constraints model that each node of a tour must be connected to other nodes by exactly two edges, and the subtour elimination constraints model that any non-empty proper subset of nodes must be connected by at least two edges with the remaining set of nodes.

If \( G \) is a directed graph, we replace the constraints \( (1)-(2) \) by

The constraints on the right hand side of \( (1) \) are redundant, but we keep them for convenience. We will refer to the polyhedra of both linear programs as \( \text{SER}(G) \), where the constraints depend on whether \( G \) is directed or not. The cost of a solution \( x \) to \( \text{SER}(G) \) is defined as \( \text{cost}(x) = \sum_{e \in E(G)} \text{cost}(e)x_e \). For a set of edges \( E' \subseteq E(G) \), we write \( x(E') = \sum_{e \in E'} x_e \). We refer to the value of an optimal solution \( x^* \) to \( \text{SER}(G) \) by \( \text{Opt}_{\text{SER}}(G) = \text{cost}(x^*) \), and write \( \text{Opt}(G) \) for the cost of an optimal integral solution of \( \text{SER}(G) \). We write \( \text{SER} \) as shorthand for “subtour elimination linear programming relaxation”, for both the directed and undirected version.

The integrality gap of \( \text{SER}(G) \) is \( \text{Opt}(G)/\text{Opt}_{\text{SER}}(G) \), and the integrality gap of \( \text{SER} \) is the supremum over the integrality gaps of all instances \( G \). The support graph of a solution \( x \in \text{SER}(G) \) is the graph \( G_x \) with \( V(G_x) = V(G) \) and \( E(G_x) = \{ e \in E(G) : x_e > 0 \} \). Qian et al. [18] proved the following.

**Proposition 1** (Qian et al. [18]). To show integrality gap upper bounds of \( \text{SER} \) for \((1,2)\)-STSP, it suffices to consider graphs \( G \) of order \( n \) with \( \text{Opt}_{\text{SER}}(G) = n + \varepsilon \) and \( \varepsilon < 1 \).

We chose this formulation of the lemma, since it is possible that the integrality gap lower bound requires an infinite sequence of graphs.

The main idea of the proof of Lemma 1 is that, if \([\text{Opt}_{\text{SER}}(G)] = k \) for some \( k \geq 1 \), the LP values of all cost two edges sum up to at least \( k \). We transform \( G \) into a graph \( G' \) with \( k \) auxiliary vertices such that each of them has a cost one edge to each vertex of \( G \) that is incident to a cost two edge of the support graph. Then \( \text{Opt}_{\text{SER}}(G) = \text{Opt}_{\text{SER}}(G') \), the integral optimum stays unchanged, but the order of \( G' \) is increased. The reasoning directly translates to directed graphs.

For our results, even a small value \( \varepsilon > 0 \) can be harmful, as the support graph can still contain many cost two edges. Hence we modify Proposition 1 as follows:

**Lemma 1.** Let \( G = (V,E) \) be a graph of order \( n \) that forms an instance of \((1,2)\)-STSP or \((1,2)\)-ATSP. Then there is a graph \( G' \) of order \( n' \geq n \) such that

- \( \text{Opt}(G') \geq \text{Opt}(G) \),
• $\text{Opt}_{\text{SER}}(G) + 1 > \text{Opt}_{\text{SER}}(G') = n'$, and

• for any optimal solution $x'$ to $\text{SER}(G')$, the support graph $G'_x$ of $x'$ only contains edges of cost one, i.e., $x'_e = 0$ for all $e$ with $\text{cost}(e) = 2$.

If there is a half-integral solution $x$ to $\text{SER}(G)$ with $\text{cost}(x) = \text{Opt}_{\text{SER}}(G)$, also $x'$ can be made half-integral.

Proof. Let $k \geq 0$ be an integer such that $n + k - 1 < \text{Opt}_{\text{SER}}(G) \leq n + k$. Then we construct $G' = (V', E')$ as follows. We introduce a set $S$ of $k$ auxiliary vertices and set $V' = V \cup S$. We add all edges to $E'$ such that $G'$ is a complete graph. We set the weights as follows. For all $e \in E$, the weight is the same as in $G$. For each $e \in \delta_G(V, S)$, we set $\text{cost}(e) = 1$. For all remaining new edges $e'$, we set $\text{cost}(e') = 2$.

Let us first show that $\text{Opt}(G') \geq \text{Opt}(G)$. To this end, let $T$ be a tour in $G'$. Then each vertex $v$ from $S$ has exactly two adjacent vertices $s$ and $t$ from $V$ in $T$. We remove both edges incident to $v$, and introduce the edge $\{s, t\}$ (if $G$ is undirected) or one of the arcs $(s, t), (t, s)$ (if $G$ is directed) such that we obtain a new tour that does not contain $v$. We apply this transformation to each vertex in $S$. Clearly, the result is a solution for $G$, and none of the replacements can increase the cost.

To show the second claim, let $x^*$ be a basic solution to $\text{SER}(G)$ of cost $\text{Opt}_{\text{SER}}(G)$. We construct a solution $x'$ to $\text{SER}(G')$ as follows. We state the construction for directed graphs; for the undirected version, we simply replace arcs by edges. First we set $x' = x^*$. Then for each arc $e = (s, t)$ in $G$ with $\text{cost}(e) = 2$ and $x^*_e > 0$, we set $x'_e = 0$, and for each vertex $v \in S$, we set $x'_{(s,v)} = x'_{(v,t)} = x^*_v/k$. As a result, $x'_e = 0$ for all edges $e$ of cost two, and $x'(\delta(v)) = x'(\delta(w))$ for all $v, w \in S$. We define $a := \sum_{v \in S} (2 - x'(\delta(v)))/2$. If $a = 0$, there is nothing more to do. Otherwise, let $F \subseteq E$ be a set of edges such that $\text{cost}(e) = 1$ for all $e \in F$, $x^*(F) \geq a$, and there is an edge $f = (s', t') \in F$ with $x^*(F \setminus \{f\}) < a$.

Before we finish the construction of $x'$, let us show that the set $F$ exists. Let $E = E_1 \cup E_2$ be such that, if $\text{cost}(e) = 1$, $e \in E_1$ and otherwise, if $\text{cost}(e) = 2$, $e \in E_2$. We have

$$
\varepsilon := \text{Opt}_{\text{SER}}(G) - (k + n - 1)
= \text{Opt}_{\text{SER}}(G) - [\text{Opt}_{\text{SER}}(G)]
= \{x^*(E_1) + 2x^*(E_2)\} - [x^*(E_1) + 2x^*(E_2)]
= x^*(E_2) - [x^*(E_2)],
$$

where we used that $a > 0$ and $x^*(E_1 \cup E_2) = n$. In particular, $\varepsilon \in (0, 1)$, and therefore $x^*(E_2) < n$. We obtain

$$
a = \sum_{v \in S} (2 - x'(\delta(v)))/2 = k - \sum_{v \in S} x'(\delta(v))/2 = k - x^*(E_2)
= k - \text{Opt}_{\text{SER}}(G) + n = 1 - (\text{Opt}_{\text{SER}}(G) - (k + n - 1)) = 1 - \varepsilon.
$$

Therefore, $x^*(E_1) = n - x^*(E_2) = n - [x^*(E_2)] - \varepsilon \geq 1 - \varepsilon = a$, and we can choose $F$ to be a minimal subset of $E_1$ with $x^*(F) \geq a$.

We continue with the construction of $x'$. Analogous to the case of cost two arcs, for each arc $e = (s, t)$ in $F \setminus \{f\}$ we set $x'_e = 0$, and for each vertex $v \in S$ we set $x'_{(s,v)} = x'_{(v,t)} = x^*_v/k$. Finally, we update $a$ such that the increased values of $x'$ are considered, set $x'_f = x^*_f - a$, and, for each $v \in S$, we set $x'_{(s,v)} = x'_{(v,t)} = x^*_v/k$. This completes the construction of $x'$.

To show that $x'$ is a feasible solution to $\text{SER}(G')$, we observe that $x'(\delta(v)) = 2$ for all $v \in S$. This is the case, because by construction all vertices of $S$ are handled in exactly the same way and $x'(\delta(S)) = 2(x^*(E_2) + x^*(F \setminus \{f\}) + a) = 2k$. Analogously, $x'(\delta^-(v)) = x'(\delta^+(v)) = 1$ for all $v \in S$. Thus, the equality constraints for all vertices in $S$ are satisfied. For the equality
constraints of the vertices in \( V \), let us fix a vertex \( v \in V \). For each edge \( e = (u, v) \in E \), each of the \( k \) edges to vertices in \( S \) obtained a value of \( (x^*_e - x'_e)/k \) in \( x' \). The case \( e = (v, u) \) is analogous, and thus \( x'(\delta(v)) = x^*(\delta(v)) = 2 \), \( x'(\delta^- (v)) = x^*(\delta^- (v)) = 1 \), and \( x'(\delta^+ (v)) = x^*(\delta^+ (v)) = 1 \).

We now analyze the subtour elimination constraints. Since \( S \) forms an independent set in the support of \( x' \), all subtour elimination constraints for subsets of \( S \) are satisfied. Instead, if we can redirect both of them to one auxiliary vertex.

By construction, the support of \( x' \) only contains edges of cost one. Therefore, \( \sum_{e \in E'} \text{cost}(e) x_e = \sum_{e \in E'} x_e = n' \), which is optimal.

Conversely, let \( x'' \) be any optimal solution to \( \text{SER}(G') \). Then

\[
\sum_{e \in E(G')} \text{cost}(e) x''_e = n' = \sum_{e \in E(G')} x_e,
\]

which can only be true if \( x''_e = 0 \) for all \( e \) with \( \text{cost}(e) = 2 \).

To show the last claim, note that in the proof we equally distributed the value \( x^*_e \) to the arcs incident to auxiliary vertices only for convenience of notation. Instead, if \( x^*_e = 1 \), we can use a single auxiliary vertex exclusively to replace \( x^*_e \) and, for two arcs \( e \) and \( e' \) with \( x^*_e = x^*_{e'} = 1/2 \), we can redirect both of them to one auxiliary vertex.

### 2.2.1 Integrality Gap Lower Bounds

As a corollary of Lemma 1, we obtain the following.

**Corollary 1.** Let \( \alpha \) be the integrality gap of \( \text{SER} \) (for either \((1, 2)\)-\text{STSP} or \((1, 2)\)-\text{ATSP}). If for any \( \varepsilon > 0 \) there are infinitely many instances \( G \) such that \( \text{Opt}(G)/\text{Opt}_{\text{SER}}(G) > \alpha - \varepsilon \), then \( \alpha \) is the integrality gap of \( \text{SER} \) restricted to instances whose support of optimal solutions \( x^* \) only contains edges of cost one.

Note that the definition of integrality gaps does not exclude that the integrality gap is attained by an instance of finite size such that all larger instances lead to a smaller integrality gap. We see the situation stated in Corollary 1 as the most relevant setting, since from an algorithmic point of view we can solve finite instances optimally in constant time.

Furthermore, we conjecture that the instances required in Corollary 1 exist (that is, there are infinitely many instances with integrality gaps arbitrarily close to the integrality gap of \((1, 2)\)-\text{STSP} resp. \((1, 2)\)-\text{ATSP}). Given an edge \( e \) with \( x_e = 1 \), we call \( e \) a 1-edge of \( x \). Our conjecture is backed by a result by Williamson [21] that states that there is an infinite class of instances with integrality gap 10/9. For completeness, we provide a full proof of Williamson’s result, but from a more general point of view.

**Lemma 2.** Let \( G \) be an instance of \((1, 2)\)-\text{STSP} and let \( x^* \) be an optimal solution to \( \text{SER}(G) \) such that \( \text{Opt}(G)/\text{Opt}_{\text{SER}}(G) = \alpha \) and there is a degree-two vertex in the support \( G_{x^*} \). Then there are infinitely many instances \( G' \) with \( \text{Opt}(G')/\text{Opt}_{\text{SER}}(G') \geq \alpha \).

**Proof.** Let \( n > 2 \) be the order of \( G \). (If \( n \leq 2 \), the claim is trivially true.) We show how to obtain an instance \( G' = (V', E') \) of order \( 2n \) and an optimal solution \( x' \) to \( \text{SER}(G) \) such that the support of \( x' \) has a vertex \( v \) with only two adjacent vertices and \( \text{Opt}(G')/\text{Opt}_{\text{SER}}(G') = \alpha \). Since we can repeat the construction arbitrarily often, the claim follows inductively.
Without loss of generality, we assume that all edges not in the support of \( x^* \) are of cost two, since the assumption does not change cost(\( x^* \)) and cannot decrease Opt(\( G \)). Let \( s, t \) be the two vertices adjacent to \( v \). Then, due to the equality constraints, \( x_{v, s} = x'_{v, t} = 1 \) and \( s \neq t \). Furthermore, we can assume cost\( \{v, s\} = \) cost\( \{v, t\} = 1 \). Otherwise, suppose that cost\( \{v, s\} = 2 \). Decreasing the cost to one also decreases cost(\( x^* \)) by one and Opt\( \text{SER}(G) \) by at most one. Hence, the integrality gap cannot decrease. The same holds for the edge \( \{v, t\} \).

For \( i \in \{1, 2\} \), let \( G_i = (V_i, E_i) \) be a copy of \( G \) and let \( v_i, s_i, t_i \) be the copies of \( v, s, \) and \( t \). We set \( V' = V_1 \cup V_2 \). We first set the edge costs in \( E' \) to those of \( E_1 \) and \( E_2 \). For the remaining edges \( e \in \{\{u, w\} : u \in V_1, w \in V_2\} \), we set cost\( (e) = 2 \). Afterwards, we change the cost of four edges as follows (see also Fig. 1):

\[
\begin{align*}
\text{cost}(\{v_1, s_2\}) &= \text{cost}(\{v_2, s_1\}) = 1, \\
\text{cost}(\{v_1, s_1\}) &= \text{cost}(\{v_2, s_2\}) = 2\,.
\end{align*}
\]

Correspondingly, we set

\[
\begin{align*}
x'_{v_1, s_2} &= x'_{v_2, s_1} = 1, \\
x'_{v_1, s_1} &= x'_{v_2, s_2} = 0\,.
\end{align*}
\]

We set \( x' \) for all remaining edges of \( G_1 \) and \( G_2 \) according to \( x^* \). For all edges not yet considered, we set \( x' \) to zero. Clearly, cost\( (x') = 2 \text{cost}(x^*) \), and the support of \( x' \) has some vertex of degree 2.

Let us now verify that \( x' \) is a feasible solution to \( \text{SER}(G') \). For the equality constraints it suffices to consider \( v_1, v_2, s_1, \) and \( s_2 \), because the edges incident to all other vertices did not change. Since we simply replaced edges by others with identical value, all equality constraints are satisfied.

For all \( S \subseteq V_1 \) and \( S \subseteq V_2 \), the subtour elimination constraints are satisfied, where we use that any of these cuts containing \( \{s_i, v_i\} \) also contains \( \{s_i, v_{3-i}\} \), \( i \in \{1, 2\} \). Fix a set \( S \) not yet considered. If either both \( v_1 \) and \( v_2 \) or none of them is in \( S \), \( \delta_{G'}(S) \) contains all edges of a cut in either \( G_1 \) or \( G_2 \) and thus we are done. Thus, since \( \delta(S) = \delta(V' \setminus S) \), we may assume that \( v_1 \in S \) and \( v_2 \notin S \). Let \( S_1 = S \cap V_1 \) and \( S_2 = S \cap V_2 \). We obtain that

\[
\begin{align*}
x'(\delta(S)) &= x'(\delta(S_1)) + x'(\delta(S_2)) - 2x'_{v_1, s_2} \geq 2 + 2 - 2 = 2.
\end{align*}
\]

Finally, we have to argue that Opt\( (G') \geq 2 \text{Opt}(G) \). For the sake of contradiction, assume that Opt\( (G') < 2 \text{Opt}(G) \). We derive a contradiction by constructing solutions within \( G_1 \) and \( G_2 \) such that the smaller one has a cost of at most \( 2 \text{Opt}(G')/2 \). Fix an optimal solution \( C' \) for \( G' \).

Let \( k \) be the number of edges \( e = \{u, w\} \) such that cost\( (e) = 2 \), \( u \in V_1 \), and \( w \in V_2 \). We remove all of these edges and are left with \( k \) paths if \( k \geq 1 \) or one cycle if \( k = 0 \). We replace \( \{v_1, s_2\} \) by \( \{v_1, s_1\} \), and replace \( \{v_2, s_1\} \) by \( \{v_2, s_2\} \), if these are contained in \( C \). As a result, there are no edges between \( G_1 \) and \( G_2 \) left and all vertices have a degree of at most two. Note that within each of the two graphs \( G_1 \) and \( G_2 \), there is either a single cycle or a collection of paths (but not both). There are exactly \( 2k \) vertices of degree one. Therefore, we can introduce \( k \) edges of cost at most \( 2k \) in order to form Hamiltonian cycles \( C_1 \) and \( C_2 \) in \( G_1 \) and \( G_2 \). Let us rename the graphs such that cost\( (C_1) \leq \text{cost}(C_2) \). By dropping the indices, we use \( C_1 \) to form a tour \( C \) in \( G \). Since cost\( (\{v, s\}) = 1 \) in \( G \), cost\( (C) \leq \text{Opt}(G')/2 < \text{Opt}(G) \), a contradiction. \( \square \)

We note that Boyd and Pulleyblank \[7\] showed that, given a basic solution \( x \) to \( \text{SER}(G) \) for an undirected graph \( G \) of order at least three, there are at least three 1-edges of \( x \).

For (1, 2)-STSP, Fig. 1a shows a well-known 10/9 integrality gap instance. The first step of generalization by using the construction of Williamson \[21\] is depicted in Fig. 1b.

For (1, 2)-ATSP, we claim that the instance depicted in Fig. 1c provides a lower bound of 6/5 on the integrality gap. Due to its simplicity, we expect the instance to be known, but we are not aware of a previous appearance in the literature. The instance can be extended to arbitrary size, where Fig. 1d depicts the first step of generalization. However, for a generic
Suppose that any two paths \( P \) and \( P' \) in the graph \( G \), directed or undirected, intersect only in their endpoints \( v \) and \( t \). We show how to obtain an instance \( G' \) of \((1,2)\)-STSP such that the solution \( x' \) is a solution in \((1,2)\)-ATSP. For the directed graphs, there is a solution \( x' \) such that \( x' = 1/2 \) for all dashed edges \( e \) and \( x' = 1 \) otherwise. For the directed graphs, \( x' = 1/2 \) for all depicted arcs \( e \).

For the construction, we require the instance to have additional properties. Let us first verify that both the instance of Fig. 1c and Fig. 1d provide an integrality gap lower bound of 6/5. Clearly, by checking all cuts we can observe easily that assigning \( x' = 1/2 \) to each arc yields a feasible solution to \( \text{SER} \) of cost 5 resp. 10.

For the instance in Fig. 1c, we want to show that there is at least one arc of cost two. Otherwise, due to the symmetry, we can assume without loss of generality that \((v_1, t_1)\) is an arc of the integral solution with exactly one arc of cost two. By means of contradiction, let us assume that there is an integral solution with exactly one arc \( a \) of cost two.

By symmetry, again we can assume without loss of generality that \((v_1, t_1)\) is an arc of the integral solution. Then the tail of \( a \) is in the left 4-cycle, since otherwise there is a vertex in the left 4-cycle that cannot be collected without using an arc of cost two. The head of \( a \) cannot be \( v_1 \) since otherwise we created a short cycle (or used another arc of cost two). Thus, the arc \((s_2, v_1)\) must belong to the solution. By an analogue argument as above, the head of \( a \) belongs to the right 4-cycle. However, to collect \( v_2 \) the tour has to move back to the left 4-cycle and can only reach \( v_1 \) by using another arc of cost 2. Therefore, indeed any integral solution uses at least two arcs of cost two.

We now show that the construction can be extended infinitely.

**Lemma 3.** Let \( G \) be an instance of \((1,2)\)-ATSP with an optimal solution \( x^* \) to \( \text{SER}(G) \) such that \( \text{Opt}(G)/\text{Opt}_{\text{SER}}(G) = \alpha \) and a vertex \( v \) with exactly two adjacent vertices in the support \( G_{x^*} \). Suppose that any two paths \( P_1, P_2 \) with \( V(P_1) \cap V(P_2) = \{v\} \) and \( V(P_1) \cup V(P_2) = V(G) \) that either both start in \( v \) or both end in \( v \) satisfy \( \text{cost}(P_1) + \text{cost}(P_2) \geq \text{Opt}(G) - 2 \). Then there are infinitely many instances \( G' \) of \((1,2)\)-ATSP with \( \text{Opt}(G')/\text{Opt}_{\text{SER}}(G') \geq \alpha \).

**Proof.** We show how to obtain an instance \( G' = (V', E') \) of order \( 2n \) and a solution \( x' \) to \( \text{SER}(G') \) such that the support graph \( G'_{x'} \) again satisfies the conditions of the lemma and \( \text{Opt}(G')/\text{Opt}_{\text{SER}}(G') \geq \alpha \) (see also Fig. 2). The idea is similar to the proof of Lemma 2. Again, we create two copies \( G_1, G_2 \) of \( G \).

Without loss of generality, we assume that all arcs not in the support of \( x^* \) are of cost two, since the assumption does not change \( \text{cost}(x^*) \) and cannot decrease \( \text{Opt}(G) \). Let \( s \) and \( t \) be the two vertices adjacent to \( v \).

For \( i \in \{1,2\} \), let \( G_i = (V_i, E_i) \) be a copy of \( G \) and let \( v_i, s_i, t_i \) be the copies of \( v, s, t \), respectively. We set \( V(G_i') = V(G_1) \cup V(G_2) \). We first set the arc costs in \( E(G_1') \) to those of \( E(G_1) \) and \( E(G_2) \). For the remaining arcs \( e \in \{(u, w), (w, u) : u \in V(G_1), w \in V(G_2)\} \), we...
set $\text{cost}(e) = 2$. Afterwards, we change the weights of the following eight arcs:
\[
\begin{align*}
\text{cost}((s_1, v_2)) & := \text{cost}((s_1, v_1)), \\
\text{cost}((s_2, v_1)) & := \text{cost}((s_2, v_2)), \\
\text{cost}((s_1, v_1)) & := 2, \\
\text{cost}((v_1, s_1)) & := 2,
\end{align*}
\]
Let $a := x^*_{(v,s)}$. Then $x^*_{(t,v)} = a$ and $x^*_{(s,v)} = x^*_{(v,t)} = 1 - a$, again due to the equality constraints. We set
\[
\begin{align*}
x'_{(v_1, s_2), x'(v_2, s_1)} & := a, \\
x'_{(s_1, v_2), x'(s_2, v_1)} & := 1 - a,
\end{align*}
\]
For all remaining arcs of $G_1$ and $G_2$ we set $x'$ according to $x^*$. For all arcs not yet considered, we set $x'$ to zero. Clearly, $\text{cost}(x') = 2\text{cost}(x^*)$, since for each arc of $G_1$ and $G_2$ there is a corresponding arc with the same LP value and the same cost in $G'$. Let us now verify that $x'$ is a feasible solution to SER($G$). For the equality constraints it suffices to inspect $v_1, v_2, s_1, s_2, t_1, t_2$, because the arcs incident to all other vertices did not change. Since we simply replaced replaced arcs by others with both identical cost and LP value, all equality constraints are satisfied.

Similarly, for all $S \subseteq V_1$ and $S \subseteq V_2$, the subtour elimination constraints are satisfied, where we use that any cut containing $(s_1, v_1)$ also contains $(s_1, v_2)$; for all other changed arcs, the situation is analogous. If either both $v_1, v_2$ or none of $v_1, v_2$ belongs to $S$, then consider the graph where $v_1$ and $v_2$ are identified to a single vertex, keeping parallel arcs. We have $x'((\delta^- G_1(S)) \geq x'((\delta^- G_1(S \cap V(G_1))) = x'((\delta^- G_1(S \cap V(G))) \geq 1$, and thus we are done.

Finally, let us fix a set $S$ not yet considered. Since $\delta^- G_1(S) = \delta^- G_1(V(G') \setminus S)$, we may assume that $v_1 \in S$ and $v_2 \notin S$.

We have $x^*((\delta^- G_2(S \cap V(G_2))) \geq 1$, and $(v_2, s_2)$ is the only arc in $\delta^- G_2(S \cap V(G_2)) \cap E(G_2)$ whose value in $G'$ and $x'$ could have changed. Therefore, either $(s_2, v_1) \in \delta^- G_1(S)$ or $x'(\delta^- G_1(S \cap V(G_2))) \geq 1 - a$.

Analogously, we have $x^*((\delta^- G_2(S \cap V(G_1))) \geq 1$, and $(s_1, v_1)$ is the only arc in $\delta^- G_2(S \cap V(G_1)) \cap E(G_1)$ whose value in $G'$ and $x'$ could have changed. Therefore, either $(v_2, s_1) \in \delta^- G_1(S)$ or $x'(\delta^- G_1(S \cap V(G_1))) \cap E(G_1)) \geq a$.

Thus,
\[
x'(\delta^- G_1(S)) \geq \min\{x'(\delta^- G_1(S \cap V(G_2)) \cap E(G_2)), x'(\delta^- G_1(S) \cap (s_2, v_1))
\]
\[
\geq \min\{x'(\delta^- G_1(S \cap V(G_1)) \cap E(G_1)), x'(\delta^- G_1(S) \cap (v_2, s_1))
\]
\[
= 1 - a + a = 1,
\]
where $x'(\emptyset) = 0$. Therefore, $x'$ is a feasible solution to SER($G'$).
We now show that \( \text{Opt}(G') = 2\text{Opt}(G) \). To see that \( \text{Opt}(G') \leq 2\text{Opt}(G) \), consider a tour \( C \) in \( G \). If neither \((v,s)\) nor \((s,v)\) is in \( E(C) \), the cost of copies \( C_1, C_2 \) of the tours in \( G_1 \) and \( G_2 \) does not change in \( G' \), and we can combine both tours \( C_1, C_2 \) to a single tour by removing two arcs of cost two and introducing two arcs of cost at most two. Otherwise, by symmetry we may assume that \((s,v) \in E(C) \). Then in \( G' \) we replace \((s_1, v_1) \) of \( C_1 \) by \((s_1, v_2) \) and \((s_2, v_2) \) of \( C_2 \) by \((s_2, v_1) \) to obtain a new tour, whose cost again is at most \( 2\text{cost}(C) \). This shows that \( \text{Opt}(G') \leq 2\text{Opt}(G) \).

We continue to prove that \( \text{Opt}(G') \geq 2\text{Opt}(G) \). Assume, for the sake of contradiction, that there is a tour \( C \) of cost \( \text{cost}(C) < 2\text{Opt}(G) \). Let \( k \) be the number of cost two arcs in \( C \). We remove these \( k \) arcs of cost two from \( C \), and obtain a collection \( C' \) of paths (unless \( k = 0 \)).

For any subgraph \( H \) of \( G' \), we define a reverse construction \( \text{rev}(H) \) from \( H \) as follows:

- if \((s_1, v_2) \in E(H)\), remove \((s_1, v_2)\) and add \((s_1, v_1)\);
- if \((v_2, s_1) \in E(H)\), remove \((v_2, s_1)\) and add \((v_1, s_1)\);
- if \((s_2, v_1) \in E(H)\), remove \((s_2, v_1)\) and add \((s_2, v_2)\);
- if \((v_1, s_2) \in E(H)\), remove \((v_1, s_2)\) and add \((v_2, s_2)\).

The arc costs in \( \text{rev}(H) \) are reversed accordingly. Then \( \text{rev}(C') \) is a modified collection of paths and cycles without paths crossing between \( G_1 \) and \( G_2 \). For \( i = 1, 2 \), let \( C'_i \) be the restriction of \( \text{rev}(C') \) to \( G_i \).

Let us first assume that \( \{(s_1, v_2), (t_1, v_1)\} \cap \text{E}(C') < 2 \), \( \{(v_2, s_1), (v_1, t_1)\} \cap \text{E}(C') < 2 \), and \( \{(v_2, s_2), (v_1, t_2)\} \cap \text{E}(C') < 2 \). Intuitively, this means that if there are arcs crossing between \( G_1 \) and \( G_2 \), the reversed arcs have opposite directions.

Then both the in-degree and out-degree of each vertex in \( \text{rev}(C') \) is at most one. Since we did not change the number of arcs, there are still \( k \) paths and either zero, one or two cycles. By introducing \( k \) arcs of cost two, we obtain a tour \( C_1 \) in \( G_1 \) and a tour \( C_2 \) in \( G_2 \) such that \( \text{cost}(C_1) + \text{cost}(C_2) \leq \text{Opt}(G') \). Therefore, \( \min\{\text{cost}(C_1), \text{cost}(C_2)\} < \text{Opt}(G) \), a contradiction.

Alternatively, it is sufficient to consider the case \( \{(v_2, s_1), (v_1, t_1)\} \cap \text{E}(C') = 2 \) such that \( |E(C'_1)| \geq |E(C'_2)| \). Then in \( C'_1 \) there are two paths \( P'_1 \) and \( P'_2 \) starting from \( v_1 \) and \( v_2 \) and all vertices have in-degrees and out-degrees of at most one. By adding at most \( k/2 - 1 \) arcs to \( C'_1 \), we can extend \( P'_1 \) and \( P'_2 \) to \( P_1 \) and \( P_2 \) such that in \( \text{rev}(G') \) it holds \( V(P_1) \cap V(P_2) = \{v_1\}, V(P_1) \cup V(P_2) = V(G_1) \), and both \( P_1 \) and \( P_2 \) start in \( v_1 \).

However, since there are exactly as many arcs crossings from \( G_2 \) to \( G_1 \) as from \( G_1 \) to \( G_2 \), in \( G' \) there are at least two arcs of cost two from \( G_1 \) to \( G_2 \). Therefore, \( k \geq 2 \) and \( E(P_1) \cup E(P_2) \) contains at most \( k/2 - 1 \) arcs of cost two. By basic graph theory, \( |E(P_1) \cup E(P_2)| = |V(G_1)| - 1 \), and therefore \( \text{cost}(P_1) + \text{cost}(P_2) \leq |V(G_1)| - 1 + k/2 - 1 \leq \text{Opt}(G')/2 - 2 < \text{Opt}(G) - 2 \), contradicting the assumptions of the lemma.

To finish the proof, we have to show that also \( G' \) satisfies the conditions of the lemma. Again, we show the claim by means of contradiction. Suppose there are two paths \( P_1, P_2 \) in \( G' \) such that \( V(P_1) \cap V(P_2) = \{v_1\}, V(P_1) \cup V(P_2) = V(G') \) with \( \text{cost}(P_1) + \text{cost}(P_2) < \text{Opt}(G') - 2 \).

Consider the case when both \( P_1 \) and \( P_2 \) start in \( v_1 \). Let \( k \) be the number of arcs of cost two in \( E(P_1) \cup E(P_2) \). We remove all \( k \) arcs of cost two and obtain a collection \( C \) of \( k + 1 \) paths (where the component containing \( v_1 \) is counted as one path). Note that the paths in \( C \) can only pass between \( G_1 \) and \( G_2 \) via \( v_1 \) and \( v_2 \). Therefore, we can use \( k \) arcs to connect the paths of \( C \) such that we obtain two paths \( P'_1 \) and \( P'_2 \) where \( V(P'_1) \cap V(P'_2) = \{v_1\}, V(P'_1) \cup V(P'_2) = V(G'), \) both \( P'_1 \), \( P'_2 \) start in \( v_1 \), \( \text{cost}(P'_1) + \text{cost}(P'_2) < \text{Opt}(G') - 2 \), and additionally there are no cost two arcs between \( G_1 \) and \( G_2 \). Note that one of the paths, say \( P'_1 \), has to contain \( v_2 \). Thus, \( \text{rev}(P'_1) \) contains a tour in either \( G_1 \) or \( G_2 \). By renaming, let us assume that the tour is in \( G_1 \). Then \( \text{rev}(P'_2) \) and the part of \( \text{rev}(P'_1) \) in \( G_2 \) form two paths \( P''_1, P''_2 \) in \( G_2 \) such that \( V(P''_1) \cap V(P''_2) = \{v_2\}, V(P''_1) \cup V(P''_2) = V(G_2), \) both \( P''_1, P''_2 \) start in \( v_2 \) or both \( P''_1, P''_2 \).
end in $v_2$, and by assumption of the lemma, $\text{cost}(P_1) + \text{cost}(P_2) \geq \text{Opt}(G) - 2$. We obtain $\text{cost}(P_1) + \text{cost}(P_2) \geq \text{cost}(P'_1) + \text{cost}(P'_2) \geq \text{Opt}G + \text{Opt}G - 2 = \text{Opt}G'' - 2$, a contradiction.

The remaining cases where $P_1$ and $P_2$ start in $v_2$, or end in $v_1$, or end in $v_2$, are dealt with analogously. □

2.2.2 Algorithmic Point of View

Alternatively, to overcome the increase of $\text{Opt}_{\text{SER}}(G)$ in Lemma[1] we can use a slightly modified version of SER using the standard trick of introducing a target value $T$. Precisely, this means to add the constraint $\sum_{e \in E} \text{cost}(e)x_e = T$ to SER. However, we choose $T \in \mathbb{N}$ instead of $T \in \mathbb{Q}$, because the cost of any optimal solution to (1,2)-STSP or (1,2)-ATSP is an integer. It is not hard to verify that by our consideration above, for the modified linear program we can obtain a fractional Hamiltonian instance $G'$ such that a solution for $G'$ can be transformed into an equally good solution for $G$.

We use the following linear programming formulation for (1,2)-ATSP:

$$
\sum_{e \in E} \text{cost}(e)x_e = T,
\sum_{e \in \delta^+_G(v)} x_e = 1 \quad \text{and} \quad \sum_{e \in \delta^-_G(v)} x_e = 1 \quad \text{for} \quad v \in V,
\sum_{e \in \delta^+_G(S)} x_e \geq 1 \quad \text{and} \quad \sum_{e \in \delta^-_G(S)} x_e \geq 1 \quad \text{for} \quad \emptyset \neq S \subset V, \text{ and}
\quad x_e \geq 0 \quad \text{for all} \quad e \in E.
$$

The formulation for undirected graphs is analogous. We refer to the polytope of the linear program for an instance $G$ by $\text{SER}(T,G)$.

We define $\text{SER}(n,G)$ to be a simplified notation of the more precise $\text{SER}(n',G')$. That is, we implicitly assume that Corollary[2] already was applied beforehand and thus the support graph of any optimal solutions for the given instance $G$ has only edges/arcs of cost one.

There must be at least one $T \in [n,2n]$ for which $\text{SER}(T,G)$ is non-empty, and thus we have to solve the linear program at most $n$ times. Note that we cannot use binary search because $T$ is integral.

**Corollary 2.** Let $G$ be a directed or undirected graph of order $n$ that forms an instance of (1,2)-STSP or (1,2)-ATSP. Then there is a graph $G'$ of order $n' \geq n$ such that

1. $\text{Opt}(G') \geq \text{Opt}(G)$;
2. $\text{Opt}_{\text{SER}}(T,G) = \text{Opt}_{\text{SER}}(T,G') = T = n'$;
3. the support graph of any solution $x^*$ to $\text{SER}(T,G')$ only contains edges of cost one;
4. if there is a half-integral solution $x$ to $\text{SER}(G)$ with $\text{cost}(x) = \text{Opt}_{\text{SER}}(G)$, there is an optimal solution $x^*$ to $\text{SER}(T,G')$ that is half-integral.

Additionally, given a tour in $G'$ of cost $\alpha T$, one can efficiently compute a tour in $G$ with performance guarantee $\alpha$.

**Proof.** We apply Lemma[1] to obtain $G'$. Therefore, the first two claims are true. The third claim follows from

$$
\sum_{e \in E(G')} x^*_e = n' = T = \sum_{e \in E(G')} \text{cost}(e)x^*_e.
$$

Suppose that $x$ is a half-integral solution to $\text{SER}(G)$ with $\text{cost}(x) = \text{Opt}_{\text{SER}}(G)$. If $\text{cost}(x)$ is integral, $G = G'$ and there is nothing left to do. Otherwise, $\text{cost}(x) = n'' + 1/2$ for some
integers $n''$. Since $x$ is half-integral, this can only happen if there is an edge $e$ with $\text{cost}(e) = 1$ and $x_e = 1/2$. Then we set $G = G'$ except that $\text{cost}(e) = 2$ and obtain $G''$ from $G$ by applying Lemma 1.

For the last claim of the corollary, suppose we are given a tour $C'$ in $G'$ with $\text{cost}(C') = T \alpha$. Let $S$ be the set of auxiliary vertices due to the application of Lemma 1. We obtain a tour $C$ in $G$ from $C'$, by replacing all paths of length two in $C'$ that pass auxiliary vertices by direct edges/arcs. Clearly, $\text{cost}(C) \leq \text{cost}(C')$ and $C$ is a feasible tour in $G$. Then the claim follows, since $T \leq \text{Opt}(G)$.

### 2.2.3 2-Matchings in the Support Graph of SER

For the argumentation within our proofs, we change the point of view to “2-matchings”. We define a 2-matching $M$ of an undirected graph $G$ as a subgraph of $G$ such that $\deg_G(v) \leq 2$ for all $v \in V(G)$. This allows us to talk about components of a 2-matching. Note that in the literature, the term 2-matching is sometimes used for perfect 2-matchings, where all degrees are exactly two [13, 19].

For a directed graph $G$, we define a directed 2-matching $M$ to be a subgraph of $G$ such that the in-degree $\deg_G^-(v) \leq 1$ and the out-degree $\deg_G^+(v) \leq 1$ for each vertex $v \in V(G)$. In other words, we use the simpler term “directed 2-matching” for a degree-two bounded 1-transshipment.

Let $G$ be a graph of order $n$ that forms an instance of $(1, 2)$-STSP or $(1, 2)$-ATSP. Let $k$ be the minimal number such that there is a (directed) 2-matching $M$ in $G$ with $k$ components. Then it is not hard to see that, if $k \geq 2$, $\text{Opt}(G) = n + k$. This observation motivates to focus only on the number of components of $M$. In particular, we do not consider edges/arcs of cost two when showing our integrality gap upper bounds. Similar to Berman and Karpinski [2], given a (directed) 2-matching $M$, an improvement of $M$ is a transformation to a 2-matching $M'$ such that one of the following conditions is satisfied, where a singleton component of a 2-matching is a single vertex without incident edges/arcs:

1. $M'$ has fewer components than $M$.
2. $M'$ has the same number of components as $M$ and more cycles.
3. $M'$ has the same number of components and the same number of cycles as $M$, but more edges in cycles.
4. $M'$ has the same number of components and cycles as $M$ and the same number of edges in cycles but fewer singleton components.
5. $M'$ has the same number of components and cycles as $M$, the same number of edges in cycles, and the same number of singleton components but fewer components of size two.

Any improvement can only be applied linearly often and only revert improvements with higher indices, so there are at most $n^{O(1)}$ improvements in total.

If $G$ is directed, let $M$ be a directed 2-matching of $G$. A vertex $v$ is a start vertex of $M$ if $\delta^-_M(v) = \emptyset$ or the component containing $v$ is a cycle. A vertex $v$ is an end vertex of $M$ if $\delta^+_M(v) = \emptyset$ or the component containing $v$ is a cycle. The reason to define vertices in cycles to be start/end vertices is that they become path starts/ends by removing an incident arc.

If there is an arc from an end vertex to a start vertex within $M$, simply adding the arc to $M$ and possibly removing arcs from cycles leads to an improvement, unless both vertices are in one cycle (that is, we obtain a new 2-matching with fewer components or with more cycles). We call such improvements basic.

For undirected graphs, we do not distinguish between start and end vertices. Let $M$ be a 2-matching of $G$. A vertex $v$ is an end vertex of $M$ if $v$ has at most one adjacent vertex within
its component or the component containing \( v \) is a cycle. Basic improvements are analogous to those of directed graphs, but we simply require an edge between two end vertices instead of an arc from a start vertex to an end vertex.

### 3 Removing Singleton Components

Let \( G \) be an instance of \((1,2)\)-STSP or \((1,2)\)-ATSP and let \( M \) be a (directed) 2-matching \( M \) in \( G_x \) for some \( x \in \text{SER}(n, G) \). Observe that the subgraph composed of all 1-edges/arcs of \( x \) is a directed 2-matching in \( G_x \).

**Lemma 4.** There is an efficient algorithm that, given a (directed) 2-matching \( M \) with a component that is a single vertex, finds a (directed) 2-matching \( M' \) in \( G_x \) that improves \( M \). Additionally, if \( M \) contains all 1-arcs/edges of \( G_x \), then also \( M' \) does.

**Proof.** We first show the lemma for directed graphs. Afterwards, the analogous result for undirected graphs follows easily. We assume that there are no basic improvements of \( M \) as otherwise we are done.

Within the proof, we write \( \delta(S) \) as shorthand for \( \delta_{G_x}(S) \). The basic idea of how to reduce the number of singleton components is as follows. Let \( v \) be a vertex that forms a component in \( M \). If there is a vertex \( w \in N^+(v) \) such that \( w \in C \) for a component \( C \) of \( M \) and removing \( \delta_M^{-1}(w) \) does not create a singleton component, we have found a suitable transformation by including \((v, w)\) and removing \( \delta_M^{-1}(w) \). Since \( x(\delta^{-1}(w)) = 1 \), \( x(\delta_M^{-1}(w)) < 1 \) and thus we do not remove a 1-arc. However, in general we need a sequence of transformations in order to ensure that the removal does not create a singleton component. In the following, we show how to find a sequence of such transformations.

We show the claim of the lemma by induction on the size of a certain set \( S \) of vertices. Initially, \( S \) only contains \( v \). A pseudo-component \( C' \) is a path of length one such that \( C' \) is contained in a path \( P \) of \( M \), where \( C' \) contains the start-vertex of \( P \). In particular, a path of length one in \( M \) is at the same time a pseudo-component. Let \( S_{\text{start}} \) and \( S_{\text{end}} \) be the sets of start resp. end vertices in \( S \), and \( S_{\text{pseudo-end}} \) the set of pseudo-end vertices in \( S \), i.e., end vertices of pseudo-components, and let \( S_{\text{intern}} = S \setminus (S_{\text{end}} \cup S_{\text{pseudo-end}}) \) be the remaining (internal) vertices in \( S \). For a more concise notation, we define \( S_{\text{ends}} = S_{\text{end}} \cup S_{\text{pseudo-end}} \).

We use a special type of alternating path. A sequence of arcs \( Q \) is an alternating path of \( M \) if

- \( Q \) is a path with alternating orientations of its arcs, i.e., for three vertices \( \{z, y, z'\} \) such that \( z \) and \( z' \) are adjacent to \( y \) in \( Q \), the arcs are either \((z, y), (z', y)\) or \((y, z), (y, z')\);
- \( Q \) starts with a tail of an arc at an end vertex;
- no forward arc of \( Q \) is in \( E(M) \);
- all backward arcs of \( Q \) are in \( E(M) \).

For a given set of vertices \( S \), we say that \( Q \) is a start-alternating path of \( M \) and \( S \), if additionally

- all vertices of \( Q \) are in \( S \);
- all tails of arcs in \( Q \) are in \( S_{\text{start}} \);
- all heads of arcs in \( Q \) are in \( S_{\text{ends}} \);
- the path \( Q \) starts with \( v \) as tail;
- the path \( Q \) ends with the tail of an arc.
We want $S$ to maintain the following invariants:

1. $x(\delta^+(S) \cap \delta^+(S_{\text{start}})) \geq 1$.

2. $S$ contains only whole (vertex sets of) pseudo-components.

3. There is a collection $Q$ of start-alternating paths of $S$ such that each vertex of $S$ is contained in at least one of the paths and the paths form a tree $T$ with $v$ as root such that all vertices in $S_{\text{ends}} \setminus \{v\}$ have a degree of at most two within $T$.

4. There is no 1-arc in the subgraph of $G$ induced by $S$.

5. If the total number of pseudo-components in $S$ is $k$ and we remove an arbitrary start vertex from $S$, there is a decomposition of the remaining vertices in $S$ into $k-1$ pseudo-components such that there are no singleton components.

We show that we can either increase the size of $S$ or we find an improvement. Clearly, $S = \{v\}$ satisfies all invariants.

Now suppose that $S$ contains $k$ pseudo-components. Then, by Invariant 1 there is an arc $e = (u, w)$ such that $u \in S_{\text{start}}$ and $w \notin S$. If $w$ is a start vertex not in $S$, we add the arc $(u, w)$ and remove $u$ from $S$. By the induction hypothesis, this allows us to obtain $k-1$ components in $S$ and thus we are done. Thus we assume that $w$ is the internal vertex of a path $P$ in $M$. If the arc $e' \in \delta^+_M(w)$ does not contain the start vertex of $P$, we are also done: since $x(\delta^{-}(w)) = 1$ and $x_{(u,w)}>0$, $x_{e'}<1$ and the component of $P$ without $e'$ that does not contain $w$ has at least two vertices. Therefore we may add $e$ to $M$, remove $e'$, and apply the induction hypothesis.

The remaining possibility is that $w$ is a pseudo-end vertex of a pseudo-component. We claim that including the two vertices of $e'$ into $S$ still satisfies the invariants or we have found an improvement. Let $S' = S \cup \{w, s\}$, where $s$ is the second vertex of $e'$. Clearly, Invariant 2 is satisfied. Similarly, Invariant 3 follows easily since the two arcs $(u, w)$ and $(s, w)$ extend a start-alternating path or a sub-path. In the following, $Q'$ is the set of start-alternating paths in $S'$, obtained from $Q$ by adding the start-alternating path from $v$ to $s$ by extending a sub-path in $Q$ (and therefore the union of paths in $Q'$ again is a tree). Furthermore, $x_{(u,w)} < 1$, since otherwise $w$ is a start vertex and we obtain an improvement by applying the induction hypothesis. Since $x_{(u,w)} > 0$, $x_{(s,w)} < 1$ which implies that Invariant 4 is satisfied. To show that Invariant 5 is valid for $S'$, let $k$ be the number of components in $S$ and thus $k + 1$ is the number of components in $S'$. If we remove $s$, the component obtained by adding the arc $(u, w)$ with $u$ as start vertex has at least two vertices and by the induction hypothesis, removing $w$ from $S$ leads to $k - 1$ remaining components, that is, there are $k$ components left in $S'$. Removing another start vertex of $S$ also leads to $k$ components by applying the induction hypothesis to $S$ and keeping $(s, w)$ as separate component.

For the remaining Invariant 1 we first show that the lemma follows directly if there are arcs from $s$ to start vertices in $S'_{\text{start}}$. Inductively, our argument implies that there are no arcs between two start vertices within $S'$. Suppose there is an arc $a = (s, s') \in \delta^+(S_{\text{start}})$ for some vertex $s' \in S'_{\text{start}}$. Let $Q$ be the start-alternating path in $Q'$ that ends in $s$. Starting from $v$, path $Q$ has alternating forward- and backward arcs. We modify $M$ and obtain a 2-matching $M'$ by including all forward arcs of $Q$ and removing all backward arcs of $Q$. Additionally we include the arc $(s, s')$. Thus the number of added arcs is larger than the number of removed arcs and therefore it is sufficient to show that we did not create a cycle or a new singleton component. There are no new cycles in $M'$, because all vertices except $v$ and $s'$ have the same degree in $M$ and $M'$, every second vertex of $Q$ is of degree one, and both the component containing $v$ and the one containing $s'$ are paths in $M'$. We did not create singleton components because each vertex in $Q$ has an incident arc in $M'$. The vertices not in $Q$ keep all of their incident arcs of $M$.

We are now ready to show that Invariant 1 is valid in $S'$. To this end, consider the following flow network $F$; see Fig. 3.
The network $F$ has one source node $p$, one sink node $q$, and one node for each pseudo-component in $S'$. The arcs in $F$ are as follows. Let $r, r'$ be two nodes that correspond to the pseudo-components in $S'$ formed by the arc $(g, h)$ resp. $(g', h')$ (if $r$ or $r'$ is the node for the component $v$, we simply assume $g = h = v$ resp. $g' = h' = v$; we also define $v$ itself to be a node of $F$). If there is an arc $(g, h')$, in $F$ there is an arc $(r, r')$ corresponding to $(g, h')$. For each $g'' \notin S'_{\text{start}}$, if there is an arc $(g'', h)$, there is an arc $(p, r)$ from the source $p$ to $r$ corresponding to $(g'', h)$. There is an arc $(r, q)$ from $r$ to the sink for each arc in $\delta^+(g) \cap \delta(S')$.

Note that we obtained each arc of the network $F$ from exactly one arc of $G$. Within $F$, we use the names of the arcs of $G$ for the network arcs. Let us consider the flow $f$ where $f(a)$ denotes the flow of arc or set of arcs $a$. We set $f(a) = x_a$ for each arcs $a$ in $F$. We now show that $f$ respects flow conservation.

Clearly, $x(\delta^+(v)) = x(\delta^-(v)) = 1$, and thus we have flow conservation at $v$. Let $(g, h)$ be the arc of any pseudo-component within $S'$ that is not $v$ and let $r$ be its node in $F$. Since $x(\delta^-(h)) = x(\delta^+(g)) = 1$, $x(g, h) = 1 - x(\delta^-(g) \setminus \{(g, h)\})$ and thus $x(\delta^+(g) \setminus \{(g, h)\}) = x(\delta^-(h) \setminus \{(g, h)\})$. Since we have seen that there are no arcs between two start vertices within $S'$, in $F$ it holds that $x(\delta^+(g) \setminus \{(g, h)\}) = f(\delta^-(r))$ and $x(\delta^-(h) \setminus \{(g, h)\}) = f(\delta^+(r))$. Thus, $f$ also respects flow conservation at $r$. As a consequence, using that $f(\delta^+(p)) \geq f((p, v)) \geq 1$,

$$x(\delta^+(S') \cap \delta^+(S'_{\text{start}})) = \sum_{a \in \delta^-(q)} f(a) \geq 1,$$

which shows that Invariant 1 is also valid in $S'$.

For an undirected graph $G$, we generate a directed graph $\vec{G}$ by replacing each edge $e = \{u, v\}$ by the two arcs $(u, v)$ and $(v, u)$. For a given solution $x$ to SER($n, G$), we define $\vec{x}_{(u,v)} = \frac{\vec{x}(v,u)}{2}$. It is not hard to check that if $x$ is a solution in SER($n, G$), then $\vec{x}$ is a solution in SER($\vec{G}, n$). In $\vec{G}$, we can orient any path of $G$ in any direction. In particular, we can transform a given 2-matching $M$ in $G_x$ into a directed 2-matching $\vec{M}$ in $\vec{G}_x$ without changing the number of components, cycles, edges in cycles, or singleton components. Now we apply the already proved directed version of Lemma 4 to $\vec{G}$ and obtain an improved matching $\vec{M}'$. If we apply the reverse transformation and replace each 2-cycle of $\vec{G}$ by a simple edge, we obtain an improved matching $M'$ in $G_x$. In order to include all 1-edges into $M'$, we have to take some extra care. Note that since we do not have to distinguish between start and end vertices, the only possibility of a 1-edge to be in a start-alternating path is that there is a pseudo component with a 1-edge that belongs to a path in $M$ of length at least 2. However, not both of the edges of the path can be 1-edges and therefore, by reversing the orientation of all such paths, we avoid the problem. \hfill \box
4 Undirected Alternating Paths

For undirected graphs, we define alternating paths differently than for directed graphs in Sect. 3. A similar concept of alternating paths was used by Berman and Karpinski [2], but the details in our approach are different.

Let $G$ be an instance of $(1, 2)$-STSP, let $x$ be a solution to $SER(n, G)$, and let $M$ be a 2-matching in the support graph $G_x$. We call edges in $E(M)$ matching edges and edges in $E(G_x) \setminus E(M)$ connecting edges as they connect vertices that are not adjacent within $M$. Matching edges are analogous to “white edges” of Berman and Karpinski [2], and connecting edges are related to (but are different from) “black edges”.

A path $Q$ in $G_x$ with end vertices $s, t$ is an alternating path if

1. $s$ is an end vertex of $M$ and $s$ is incident to a connecting edge in $Q$;
2. $t$ is an end vertex of $M$ and $t$ is incident to a connecting edge in $Q$;
3. in $Q$, matching edges and connecting edges alternate.

An alternating path $Q$ is inward if

4. for any connecting edge $\{u, v\}$ in $Q$ such that $u, v$ are both in the same path $P$ of $M$, either $u$ is an end vertex or $\{u, v\}, \{u, u'\}$ are consecutive edges of $Q$, where $u'$ is the vertex adjacent to $u$ in $Q$ such that sub-path of $P$ between $u$ and $v$ contains $u'$.

The intuition is that the path proceeds to the inside of a cycle created by a connecting edge within a single path of $M$. Suppose $u, v, Q$, and $P$ are as in property 4 and $u'$ is adjacent to $u$ in $Q$, but $u'$ is not in the sub-path of $P$ between $u$ and $w$. Then we say that $u, u', v, \{u, u'\}$, and $\{u, v\}$ are involved in the violation of the inward property. The path $Q$ is a truncated alternating path if it has the properties 1, 3 but property 2 is violated.

A pair of vertices $s, t$ in a cycle $C$ is path-forming if there is a path $P_{st}$ from $s$ to $t$ in $G_x$ that contains all vertices of $C$ and no vertex of $V(G_x) \setminus V(C)$ (but $P_{st}$ may use edges outside $C$). We say that we apply an alternating path $Q$ if we add all connecting edges of $Q$ to $M$ and remove all matching edges of $Q$. If $s$ is an end vertex of $Q$ and $s$ is contained in a cycle $C$ of $M$, we additionally remove an edge incident to $u$ in $C$ from $M$. If both end vertices $s, t$ of $Q$ are in $C$ such that $s$ and $t$ are path-forming in $C$, instead of removing two edges we replace the cycle by a path from $s$ to $t$ that visits all vertices of $C$. If $Q$ starts and ends in the same vertex $s$, we remove all edges outside $Q$ that are incident to $s$.

Observation 1. Let $Q$ be an alternating path with two end vertices $s \neq t$ for a 2-matching $M$ such that $s$ and $t$ are end vertices of paths in $M$, and let $M'$ be the 2-matching obtained from applying $Q$ to $M$. Then all end vertices of paths in $M$ except $s$ and $t$ are also end vertices of paths in $M'$.

Proof. For any vertex $u \notin \{s, t\}$ in a path of $M$, when applying $Q$ to $M$, the number of edges incident to $u$ that are added matches the number of removed incident edges. In particular, no incident edge of $u$ is changed if $u$ is not in $Q$. After applying $Q$, both $s$ and $t$ are vertices of degree two.

As already observed by Berman and Karpinski [2], for alternating paths between two distinct end vertices of paths in $M$, we obtain a new 2-matching $M'$ that has an additional edge compared to $M$. However, applying an alternating path $Q$ may create cycles and this way, the number of components may increase. We show that such an inconvenient situation requires $Q$ to have a certain length.

Lemma 5. Let $Q$ be an alternating path in $G_x$ with end vertices $s \neq t$ for a 2-matching $M$. We can find an improved 2-matching $M'$ if
Proof of Lemma \[\text{5}.\] Suppose condition (a) is satisfied. By the definition of alternating paths, the length of \(Q\) is odd. Therefore, if condition (a) is satisfied, then \(Q\) is a single edge between two end vertices and there is a basic improvement. That is, applying \(Q\) decreases the number of components or, if \(s\) and \(t\) are the two end vertices of one path, \(M\) has the same number of components and an increase in the number of cycles.

For the remaining conditions, let us consider the following situation. Suppose there is a cycle \(C\) in \(M\) that is not in \(M\). Then \(C\) is composed of edges in \(E(M)\) and connecting edges in \(E(Q)\). In particular, \(C\) contains at least one connecting edge of \(Q\).

Suppose condition (b) is satisfied. Then \(|E(M')| > |E(M)|\) and the length of \(Q\) is at most three. Due to condition (a), we only have to consider a length of exactly three and thus \(Q\) has exactly two connecting edges. At least one of these is between two different paths of \(M\) and thus it cannot be the only connecting edge of a cycle. Therefore either the number of cycles increases by one (and the number of components stays the same) or the number of components decreases.

Suppose condition (c) is satisfied, and therefore there are at most three connecting edges in \(Q\). We have \(|E(M')| = |E(M)| + 1\) and the statement of the lemma holds, unless the number of components is increased. By contradiction, let us assume that that \(M\) has more components than \(M\). We note that the first and last edge of \(Q\) are not contained in cycles of \(M\) that only contain one connecting edge of \(Q\), since they either span between two components of \(M\) or there are matching edges of \(Q\) that prevent the situation. Thus the only possibility to increase the number of components is that we obtain one cycle with only one connecting edge and one cycle with two connecting edges such that the middle edge of \(Q\) closes a cycle \(C\) in a path \(P'\) of \(M\). As a result, the second cycle has to include all remaining edges of \(P'\) since there are only two connecting edges of \(Q\) left and they have to be incident to \(s\) and \(t\).

Suppose condition (d) is satisfied. If \(|E(M')| > |E(M)|\), the application of \(Q\) did not lead to the removal of edges in cycles of \(M\). If \(V(Q) \subseteq V(P)\) for some path \(P\) in \(M\), the inward property enforces the direction of every edge of \(Q\) and as a result, applying \(Q\) transforms \(P\) into a single cycle. If \(V(Q)\) contains vertices not in \(P\), condition (d) is satisfied and we are done.

If \(|E(M')| \leq |E(M)|\), at least one end of \(Q\), say \(s\), is contained in a cycle \(C\) of \(M\). Since after the application of \(Q\) all vertices of \(C\) are still in one component of \(M\), \(C\) is entirely included in a cycle of \(M\) or \(C\) is entirely included in a path of \(M\).

We can exclude that in \(M\) there is a cycle \(C\) that is not in \(M\) and contains only one connecting edge of \(Q\), because the inward property of \(Q\) would enforce two matching-edges of \(Q\). By analogous considerations for \(t\), we conclude that the only possibility to have a cycle in \(M\) with less than three connecting edges in \(Q\) is that both \(s\) and \(t\) are in \(C\). However, then the total number of edges in cycles is increased without changing the number of components or cycles.
Suppose condition (c) is satisfied. If the length of \( Q \) is at most three, applying \( Q \) results in a single cycle and we are done. Otherwise applying \( Q \) creates two cycles. Additionally, the truncated inward alternating path until the first vertex within \( P \) becomes an inward alternating path of length less than 5 to an end vertex of a cycle. By the proof of condition (b), the number of components is reduced again and thus in total the number of components stays the same whereas the number of cycles increases.

In the following lemma, we have to handle paths of length one. The proof of the lemma uses the following observation.

**Observation 2.** If \( \{s,u\} \) is a component of \( M \) that is incident to a path in \( M \) of length at least 5, there is an improvement.

**Proof.** By symmetry, let us assume that there is an edge \( \{s,s_1\} \in E(G_x) \) and \( P_{s_1} \) is a path in \( M \) of length at least 5. Then in \( P_{s_1} \) there is a sub-path starting from \( s_1 \) of length at least three. Therefore, there is an improvement of type 5 (reducing the number of components of size two) by adding the edge \( \{s,s_1\} \) and removing one edge of \( P_{s_1} \).

**Lemma 6.** Let \( Q \) be an alternating path with respect to \( M \) of length less than 7 whose both ends are the same end vertex \( s \) of a path \( P \) in \( M \) and let \( u \) be the vertex adjacent to \( s \) in \( M \). There is an improved 2-matching \( M' \) if

- (a) \( u \) is adjacent to an end vertex \( t \) in another path of \( M \); or
- (b) \( \{s,u\} \) forms a component and there is an alternating path with respect to \( M \) of length less than 7 that has both its ends in \( u \).

**Proof.** Suppose condition (a) is satisfied. Let \( M'' \) be the 2-matching obtained from \( M \) by applying \( Q \). By Observation 1, \( t \) is an end vertex of a path in \( M'' \). Thus, applying \( Q \) and adding \( \{u,t\} \) gives a new 2-matching \( M' \). We now argue that \( M' \) is an improved 2-matching compared to \( M \).

The number of edges in \( M' \) is \( |E(M)| + 1 \), since applying \( Q \) did not change the number of edges and we introduced \( \{u,t\} \). Therefore, similar to the proof of Lemma 5, there is an improvement unless there are at least two cycles in \( M' \) that are not in \( M \). Again, each of the cycles has to contain connecting edges of \( Q \) and \( Q \) has at most three connecting edges. In particular, the two connecting edges of \( Q \) incident to \( s \) are in the same component of \( M' \) and therefore the only possibility to generate two cycles is to have one cycle with one connecting edge of \( Q \) and one cycle with the two connecting edges of \( Q \) incident to \( s \). But then there are four internal vertices of \( Q \) that are contained in one path of \( M \) and there is no possibility to close a cycle with only the two connecting edges of \( Q \) incident to \( s \).

Suppose condition (b) is satisfied. Let \( Q_s \) and \( Q_u \) be the two considered alternating paths starting from \( s \) and \( u \), let \( s_1, s_2, u_1, u_2 \) be the vertices adjacent to \( s \) and \( u \) in \( Q_s \) and \( Q_u \), and let \( P_{s_1} \) and \( P_{s_2} \) be the paths in \( M \) that contain \( s_1 \) and \( s_2 \).

Let us first assume that applying one of the two alternating paths, say \( Q_s \), increases the number of cycles. Let \( C \) be a new cycle that is not in \( M \) but in the resulting 2-matching. If \( C \) contains only one connecting edge of \( Q_s \), three out of at most five edges of \( Q \) only contain vertices of one path \( P' \neq P \) of \( M \). In particular, there are two non-consecutive matching edges of \( Q_s \) in \( P' \) and since \( s_1 \) and \( s_2 \) are no end vertices (unless there is a basic improvement), the length of \( P' \) is at least 5 and there is an improvement by Observation 2.

Otherwise, both edges of \( Q \) incident to \( s \) are in \( C \). Again, this can only be the case if both edges lead to the same path \( P'' \) of \( M \) since otherwise the matching edges of \( Q \) prevent the possibility to close the cycle with the remaining connecting edge (there are at most three connecting edges in \( Q \)). As in the previous case, the length of \( P'' \) is at least 5 and there is an improvement.

Thus, in the following we may assume that neither applying \( Q_s \) nor \( Q_u \) increases the number of cycles.
We show next that there is an improvement if $Q_s$ and $Q_u$ are not disjoint. We note that if in $P_1$, $s_1$ or $s_2$, say $s_1$, is adjacent to one of the vertices $u_1$ or $u_2$, say $u_1$, we can obtain an improvement by adding $\{u, u_1\}$ and $\{s, s_1\}$ and removing the edge $\{u_1, s_1\}$ from $P_1$. Due to our observations before, the only remaining possibility of $s_1$ and $u_1$ to be in the same path of $M$ without an improvement of type 5 is that $s_1$ is the same as $u_1$ (the remaining combinations are analogous). Then we obtain an improvement as follows. We apply our observations before, the only remaining possibility of $s$ and $t$ the only possibility of $s$ and $t$ to be in the same path of $M$ without an improvement of type 5 is that $s$ is the same as $t$ (the remaining combinations are analogous). Then we obtain an improvement as follows. We apply $Q_s$ to $M$, but instead of adding $\{s, u\}$ we do not remove $\{s, u\}$ and add $\{u, u_1\}$. Hence the number of edges is increased but we did not create a cycle.

Since there are only three connecting edges in $Q_s$ and $Q_u$, the only components of $M$ visited by $Q_s$ and $Q_u$ are $\{s, u\}, P_{s_1}, P_{s_2}, P_{u_1}$, and $P_{u_2}$. Neither applying $Q_s$ nor $Q_u$ separately creates a cycle and the component formed by $\{s, u\}$ is the only common component of $Q_s$ and $Q_u$ in $M$. But then, applying both $Q_s$ and $Q_u$ to $M$ does not introduce cycles and increases the number of edges by one. We conclude that the number of components has to decrease and thus we have found an improvement.

The following lemma is simple but useful.

**Lemma 7.** Let $M$ be a 2-matching in graph $G_x$, let $s$ be the end vertices of a path in $M$ and let $Q = \{s, u\}, \{u, v\}, \{v, s\}$ be a path from $s$ to $s$ of length three such that $\{u, v\}$ is in some path $P$ of $M$. Let $\{u', u\}$ and $\{v, v'\}$ be the two edges incident to $u$ and $v$ in $P$ (if they exist). Then there is an improvement if
(a) $u$ or $v$ is an end vertex of $P$; or
(b) there is an end vertex $t$ and an edge $e = \{t, w\}$ for $w \in \{u', u, v, v'\}$ such that $e \notin \{\{u', u\}, \{v, v'\}, \{s, u\}, \{s, v\}\}$.

**Proof.** If $u$ or $v$ is an end vertex, we simply add $\{s, u\}$ or $\{s, v\}$ to reduce the number of components.

If there is an edge $e$ as specified in the second condition, we add the edge $\{t, w\}$, remove $\{w, u\}$ or $\{w, v\}$ from $P$ (only one of them can be in $P$), and add $\{s, u\}$ or $\{s, v\}$ depending on the removed edge. Then either we reduced the number of components or increased the number of cycles without increasing the number of components. 

Note that all improvements considered in Lemma 7 are applications of alternating paths.

## 5 Computing a 2-Matching in the Support of SER

In this section we give an algorithm, Algorithm 1, that computes a 2-matching $M$ in the support of an optimal solution to SER ($G$) for instances $G$ of (1,2)-STSP, such that $M$ has some nice properties.

Let us analyze the algorithm. Each iteration of the while loop, except the last one, gives an improved 2-matching. Hence, Algorithm 1 computes a feasible solution to (1,2)-STSP. We can compute $x^*$ in polynomial time. Since the total number of improvements is bounded by a polynomial in $n$, the number of iterations of the while loop is also polynomial in $n$. Since Lemma 3 provides an efficient algorithm and we can list all alternating paths of a fixed constant length in polynomial time, also all steps in the while loop can be done in polynomial time. Thus, Algorithm 1 runs in polynomial time.

In some theorems we require that paths in the 2-matching $M'$ do not end with degree-2 vertices. A natural way to obtain that property would be to ensure that all 1-edges stay in $M'$. However, we only know how to prove the following.

**Lemma 8.** Every execution of Algorithm 1 which does not execute the line marked with * yields a 2-matching such that all end vertices of paths have degree least three in $G_{x^*}$.
An instance $G$ of $(1, 2)$-STSP such that $\text{SER}(n, G)$ has a solution.

Output: A 2-matching $M$.

Compute an optimal basic solution $x^*$ of $\text{SER}(n, G)$;

Let $M$ be the 2-matching in $G_{x^*}$ with $E(M) = \emptyset$ and let $M'$ be the 2-matching in $G_{x^*}$ that contains all 1-edges; \hfill // $E(M') \neq \emptyset$ (Boyd and Pulleyblank [7])

while $M \neq M'$ do

\hspace{1em} $M := M'$;

\hspace{1em} Apply Lemma 4 to $M'$; \hfill // Remove the singleton components.

\hspace{1em} If there is an alternating path $Q$ in $G_{x^*}$ that satisfies properties of Lemma 4

\hspace{1em} Lemma 6(a), or Lemma 7 obtain an improved $M'$;

\hspace{1em} * If there is an alternating path $Q$ in $G_{x^*}$ that satisfies properties of Lemma 6 (b), or

\hspace{1em} Observation 2 for $M'$, obtain an improved $M'$.

end

Algorithm 1: $(1,2)$-STSP algorithm for general graphs. The line marked with * is only used for Theorem 1.

Proof. We show the following more general statement.

Let $M$ be a 2-matching in some graph $G_x$ such that each end vertex of a path in $M$ has a degree of at least 3 in $G_x$. If there is an improved 2-matching $M''$ obtained from applying Lemma 4, Lemma 5, Lemma 6(a), or Lemma 7 then there is an improved 2-matching $M'$ such that each end vertex $s$ of a path in $M'$ has degree at least 3 in $G_x$.

Suppose $s \neq t$ and we obtained $M''$ by applying an alternating path $Q$ within one of the lemmas. Then, by Observation 1 in $M''$ both the degree of $s$ and $t$ is 2 and all other vertices in paths of $M$ have the same degree as in $M$. In particular, no degree-1 vertex of a path in $M$ can become an end vertex. The only intersection of $Q$ with cycles of $M$ may occur if $s$ or $t$ are in some cycle, as otherwise we would have considered a shorter alternating path. If both $s$ and $t$ are in the same cycle $C$, then we only considered $Q$ if $s$ and $t$ are path-forming in $C$ and the application of $Q$ did not create an end vertex of a path in $V(C)$. Otherwise, assume that $s$ belongs to a cycle $C$ with $t \notin V(C)$. Let $e$ be the edge incident to $s$ in $Q$. Since $x_e > 0$ and $x(\delta G_x(s)) = 2$, $s$ can have at most one incident 1-edge. When applying $Q$, we can choose to remove one of two edges in $C$ and in $M'$ we choose to not remove a 1-edge of $C$ (while in $M''$ a 1-edge may have been removed). We conclude that for all improvements due to Lemma 6 the claim of this lemma holds, since each improvement is obtained due two one or two applications of alternating paths. The same is true for Lemma 4 and Lemma 7.

Now let us assume that $Q$ is an alternating path that both starts and ends in the same vertex $s$. Applying $Q$ potentially creates a degree 2 vertex, but when using condition (a) of Lemma 4 this is excluded, since we apply another alternating path starting from the neighbor of $s$, which in turn obtains a degree of two.

6 Integrality Gap Upper Bounds for $(1,2)$-STSP

We now introduce our method to determine the quality of the solution computed by Algorithm 1. Let $G$ be an instance of $(1,2)$-STSP and let $x^*$ be an optimal solution to $\text{SER}(n, G)$. Furthermore, let $M$ be a 2-matching in $G_{x^*}$.

We map the LP values of the edges to the components such that the minimum sum of LP values is at least some value $\alpha$ unless we find an improvement. More precisely, let $C$ be the set of components of $M$. We either find an improvement or we find a solution to the following
linear program with variables $y_{C,e}$ for each pair of a component $C \in \mathcal{C}$ and edge $e \in E(G_x)$:

\[
\sum_{e \in E(G_x)} y_{C,e} \geq \alpha \quad \text{for all } C \in \mathcal{C},
\]

\[
\sum_{C \in \mathcal{C}} y_{C,e} \leq x^*_e \quad \text{for all } e \in E(G_x),
\]

\[
y_{C,e} \geq 0 \quad \text{for all } C \in \mathcal{C}, e \in E(G_x).
\]

We refer to the linear program as LP($x^*$). Within LP($x^*$), each $x^*_e$ is a fixed constant. Since cost($x^*$) = $n$, all edges in $M$ are of cost one and if there is a solution to LP($x^*$), we have to add at most $n/\alpha$ edges of cost at most two in order to obtain a tour. In other words, we obtain a tour of cost $n + n/\alpha$ and thus an $(\alpha + 1)/\alpha$ approximation for (1, 2)-STSP.

We start with the following known result.

**Lemma 9** (Wolsey [22]). Let $S$ be a set of vertices in $G$. Then $x^*(\delta_G(S) \cup \delta_G(S,S)) \geq |S| + 1$.

**Proof.** We have

\[
x^*(\delta_G(S) \cup \delta_G(S,S)) = (x^*(\delta_G(S)) + \sum_{v \in S} x^*(\delta_G(v)))/2
\]

\[= (x^*(\delta_G(S)) + 2|S|)/2 \geq |S| + 1,
\]

where we have to divide by two since the LP value of each edges is added twice. \qed

**Theorem 1.** There is a polynomial-time 5/4 approximation algorithm for (1, 2)-STSP with respect to Opt$_{\text{SER}}(n,G)$.

**Proof.** Let $M$ be the 2-matching computed by Algorithm 1 and let $x^*$ be the corresponding optimal solution to SER($n,G$). Instead of $\delta_G$ we simply write $\delta$.

To analyze the algorithm, we construct a solution to LP($x^*$) for $\alpha = 4$. In order to obtain meaningful alternating paths, for edges of $M$ we work with the complement of $x^*$. To this end we let $z$ be a vector of length $|x^*|$ with entries $0 \leq z_e \leq 1$ for all $e \in E(G)$. We set $z_e = x^*_e$ for all $e \in E(G) \setminus E(M)$ and $z_e = 1 - x^*_e$ for all $e \in E(M)$. We note two properties of $z$. Let $v$ be an internal vertex of a component in $M$. Then

\[
z(\delta(v) \setminus E(M)) = 2 - x^*(\delta(v) \cap E(M))
\]

\[= 2 - (2 - z(\delta(v) \cap E(M)))
\]

\[= z(\delta(v) \cap E(M)).
\]

Let $s$ be an end vertex of a path in $M$. Then

\[
z(\delta(s) \setminus E(M)) - 1 = 1 - x^*(\delta(s) \cap E(M))
\]

\[= 1 - (1 - z(\delta(s) \cap E(M)))
\]

\[= z(\delta(s) \cap E(M)).
\]

To simplify the discussion, we subdivide the edges as follows. Let $N \in \mathbb{N}$ be the smallest integer such that $x^*_e \cdot N$ is an integer for all $e \in E(G)$, where we used that each $x^*_e$ is a rational number. Note that this way, also $z_e \cdot N$ is an integer. For each $e$, we introduce $N \cdot x^*_e$ parallel edges $e_1, e_2, \ldots$. We extend the vectors $x^*$ and $z$ by setting $x^*_{ei} = 1/N$ for all $i \leq x^*_e \cdot N$ and $x^*_e = 0$ for all remaining $i$. If $e \notin E(M)$, we set $z_{ei} = x^*_{ei}$ for all $i$ and, if $e \in E(M)$, $z_{ei} = 1/N$ for all indices $x^*_e \cdot N < i \leq N$; for all remaining $i$ we set $z_{ei} = 0$.

The purpose of the subdivision is that we can assign integer values to the variables of LP($x^*$). For instance, for some component $C$, we write $y_{C,e_i} = 1$ and $y_{C,e_j} = 1$ instead of $y_{C,e} = 2/N$. We call $e_i$ a sub-edge of $e$. Each vertex has $2N$ incident sub-edges.
Note that for each internal vertex \( v \) of a path in \( M \), by \((9)\),
\[
|\{e_i : e \in \delta(v) \cap E(M), 1 \leq i \leq N, z_{e_i} > 0\}| = |\{e_i : e \in \delta(v) \setminus E(M), 1 \leq i \leq N, z_{e_i} > 0\}|, \tag{10}
\]
and for an end vertex \( s \) of a path in \( M \), by \((9)\),
\[
|\{e_i : e \in \delta(s) \cap E(M), 1 \leq i \leq N, z_{e_i} > 0\}| = |\{e_i : e \in \delta(s) \setminus E(M), 1 \leq i \leq N, z_{e_i} > 0\}| - N. \tag{11}
\]
Let \( I \) be the set of edges that lead from an end vertex of a path of \( M \) into the same path, that is,
\[
I := \{\{u,v\} \in E(G_x^+) \setminus E(M) : u \text{ is an end vertex of a path } C \text{ in } M \text{ and } v \text{ is in } C\}.
\]
If \( e \) is a sub-edges of an edges in \( I \), we say that \( e \) has high priority. In the following procedure, there are two types of marks of sub-edges: to be used as internal edge of an alternating path and to be used as at the end of an alternating path.

Let us now consider the following procedure to create alternating paths, where initially all sub-edges in \( E(G_z) \) are unmarked.

- Choose an end vertex \( s \) of a path in \( M \) with less than \( N \) sub-edges that are marked to be used ends of alternating paths, prefer high-priority sub-edges;
- Extend a path \( Q \) from \( s \) by choosing unmarked sub-edges that do not violate the inward property (property \((4)\) of the definition) and that alternate between \( E(G_z) \setminus E(M) \) and sub-edges within paths of \( M \):
  - Choose high priority sub-edges whenever possible.
  - If an end vertex \( t \) of a path in \( M \) is reached by a sub-edge not in \( M \) and there are less than \( N \) edges marked as ends of paths incident to \( t \), stop;
  - If there is no unmarked sub-edge that can be followed, stop;
- Mark the first and (if not truncated) the last sub-edge of \( Q \) to be an end of an alternating path and mark the remaining sub-edges of \( Q \) to be internal sub-edges.

It is not hard to check that any path \( Q \) created by the procedure is either an inward alternating path or a truncated inward alternating path. Let \( Q \) be the set of paths obtained by iteratively applying the procedure until each end vertex of each path in \( M \) is the end vertex of exactly \( N \) inward alternating paths or truncated inward alternating paths. Note that by \((1)\), such a set \( Q \) exists. We emphasize that we need \( Q \) only for the analysis.

Let us now construct a solution \( y \) to LP\((x^\ast)\). For each cycle \( C \) of \( M \) we set \( y_{C,e} = [x_e^\ast] \) for all sub-edges \( e \) incident to vertices of \( C \). For the remaining cases, let us fix a path \( Q \in \mathcal{Q} \).

- If \( Q \) is an inward alternating path of length less than \( 7 \) with both ends at one vertex \( s \), we set \( y_{C,e} = [x_e^\ast] \) for each sub-edge \( e \in \delta(s) \).
- If \( Q = e_1, e_2, \ldots, e_k \) is an inward alternating path of length at least \( 7 \) with both ends at one vertex \( s \), we set \( y_{C,e_1} = y_{C,e_3} = y_{C,e_5} = y_{C,e_{k-2}} = 1 \). By Lemma \((11)\), this case includes any inward alternating path with two distinct end vertices, since \( Q \) starts from the end vertex of a path.
- If \( Q = e_1, e_2, \ldots, e_k \) is a truncated inward alternating path of length at least three that starts from a path \( C \) of \( M \), we set \( y_{C,e_1} = y_{C,e_3} = 1 \). (There is no truncated inward alternating path starting from a cycle.)
• If \( Q = e_1, e_2 \) is a truncated inward alternating path of length two, by the properties of \( z \), the only reason that \( Q \) was forced to stop is that any prolongation violates the inward property. Let \( u \) and \( v \) be the vertices such that \( e_1 = \{s, u\} \) and \( e_2 = \{u, v\} \). Let \( P_u \) be the path of \( M \) containing \( s \) and \( P_v \) be the path of \( M \) that contains \( u \) (possibly \( P_u = P_v \)). Let \( P'_u \) be the sub-path of \( P_u \) without \( u \) that is left when removing \( v \) (see Fig. 4). If there is a sub-edge \( e \) of \( \{v, w\} \) for some \( w \) in \( P'_u \) such that \( y_{C', \{v, w\}} = 0 \) for all \( C' \) and \([x^*_{e'}] = 1\), we set \( y_{P_v, \{v, w\}} = 1 \). Otherwise we set \( y_{P_v, e} = [x^*_{e}] \) for all sub-edges \( e \in \delta(s) \).

Figure 4: Assignment for truncated inward alternating paths of length two.

We now show that the constraints (b) are satisfied. If we only consider cycles, alternating paths between two different vertices, alternating paths with one end vertex of length at least 7, and truncated alternating paths of length at least three, it is easy to verify that we assigned the LP value of each sub-edge to at most one component, and thus we did not violate the constraints. Note that we did not yet assign LP values of any edge in \( E(M) \).

For truncated alternating paths \( Q = e \) of length one, additionally to the edge of the truncated alternating path we assigned the LP value of one sub-edges \( e' \) of \( E(M) \). We did not assign the value of one of these edges for two different truncated alternating paths of length one, since this would imply the existence of an inward alternating path of length three which is excluded due to Lemma 5. The existence of \( e' \) with \( x^*_{e'} > 0 \) follows from the definition of \( z \) and that \( Q \) cannot be extended even though \( e \in I \).

In the remaining cases, we may have to set \( y_{C, e} = [x^*_{e}] \) for all sub-edges \( e \in \delta(s) \) where \( s \) is an end vertex of a path \( P \) in \( M \). These assignments do not interfere with assignments for cycles as otherwise there is a basic improvement. Note also that the sub-edges in \( E(M) \) used in the assignment for truncated alternating paths of length one are not in \( \delta(s) \).

Let \( s' \) be the vertex adjacent to \( s \) in \( P \). Suppose that \( P \) has at least three vertices and there is no end vertex \( t \in E(P) \) of a path in \( M \) such \( \{s', t\} \in E(G_z) \). Let us consider any alternating path \( Q \in Q \) that contains a sub-edge of \( \{s, s'\} \). We split \( Q \) into \( Q_s \) and \( Q_{s'} \) by removing the sub-edge of \( \{s, s'\} \) such that \( s \in V(Q_s) \). By our assumption and the definition of alternating paths, \( Q_{s'} \) has at least three edges. Note that due to the inward property, the vertex adjacent to \( s' \) in \( Q_{s'} \) is not the other end vertex of \( P \). By condition (b) of Lemma 5 we can also exclude that \( Q_s \) has less than 7 edges, unless \( Q_s \) is truncated or \( Q_s \) leads from \( s \) to \( s' \). Therefore, none of the LP values of sub-edges incident to \( s \) is assigned to another component than \( P \).

As a result, we may assume for the remaining cases that either \( P \) has exactly two vertices or \( s' \) is adjacent to an end vertex of another path than \( P \) of \( M \). We lead both situations to a contradiction.

By Lemma 5 there is no alternating path of length less than 7 from \( s \) to \( s' \) unless \( P \) has exactly two vertices and there is no alternating path shorter than 7 that starts and ends at the second vertex. Therefore, if \( Q \) is such an alternating path, assigning values of sub-edges in \( \delta(s) \) to \( P \) does not interfere with any of the previous cases.

We continue with truncated inward alternating paths of length two. If we were able to assign the third sub-edge consecutive to \( Q \), we are done. Otherwise, we have assigned the sub-edges
incident to \( s \). As before, let \( s, u, v \) be the vertices of the alternating path \( Q \) and let \( P_u \) be the path of \( M \) that contains \( u \) and its sub-path \( P_u' \). By the definition of inward alternating paths and the definition of \( z \), there is a vertex \( w \) in \( P_u' \) and a truncated alternating path \( Q \) such that it either is of length three and ends in \( v \) with a sub-edge of \( \{ w, v \} \) or \( Q \) is of length two, ends in \( w \), and a sub-edge of \( \{ w, v \} \) was assigned to the component where \( Q \) ends. Let \( \bar{Q} \) be the alternating path obtained by combining \( Q, \bar{Q} \), and (if the length of \( \bar{Q} \) is two) a sub-edge of \( \{ u, w \} \). The length of \( \bar{Q} \) is exactly five. Let \( w' \) and \( t \) be the two remaining vertices of \( \bar{Q} \) such that \( t \) is the end vertex. The only vertices of \( \bar{Q} \) that are possibly involved in the violation of the inward property are those adjacent to \( v \) and \( w \) in \( \bar{Q} \), i.e., \( u, v, w, w' \). In particular, the first and last edge of \( \bar{Q} \) are not involved in the violation of the inward property.

Therefore, by Lemma 5(c), \( \bar{Q} \) cannot start from a path other than \( P_u \). Since \( u, v, w, \) and \( w' \) are in the same component of \( M \) and none of them is an end vertex, \( P_u \) has a length of at least 5. Therefore, by our considerations above, and Observation 2 also \( s \neq t \) and, by Lemma 5(d) of Lemma 5, \( \bar{Q} \) contains internal vertices of \( P_u \).

Since \( u, v, w, \) and \( w' \) are in the same component of \( M \), we conclude that \( \bar{Q} \) does not leave \( P_u \). Since \( v \) and \( w \) are the only vertices involved in the violation of the inward property, we conclude that \( \bar{Q} \) has the type described in (c) of Lemma 5. Since \( P_u \) has more than two vertices, by our assumption there is an end vertex of some path \( P' \neq P_u \) adjacent to \( P_u \); this case was considered by condition (c).

Finally we show that the constraints (4) are satisfied for \( \alpha = 4 \). By Lemma 9 this is clearly true for cycles. For each end vertex of a path in \( M \), note that we assigned either 2N sub-edges to the component of \( s \) or all sub-edges in \( \delta(s) \), which also sums up to 2N. Therefore, we assigned 4N disjoint sub-edges to each path in \( M \), which amounts to a total value of 4.

**Theorem 2.** If we can obtain an optimal half-integral solution \( x^* \) to \( \text{SER}(n, G) \), there is a polynomial-time 7/6 approximation algorithm for \((1,2)\)-STSP with respect to \( \text{Opt}_{\text{SER}}(n, G) \).

**Proof.** Let \( M \) be the 2-matching obtained from Algorithm 1 without executing the line marked with *. Additionally, during the execution of the algorithm we also consider a very specific type of improvement, discussed in the end of this proof.

The analysis of the algorithm is similar to the previous one, but for the constraints (4) we show that we may choose \( \alpha = 6 \). In particular, we use the same \( z \) as in the proof of Theorem 1.

We use a modified set \( \bar{Q} \), where for each new alternating path we mark the edges as we did in the original construction of \( Q \). We start with the construction of \( \bar{Q} \) as in the proof of Theorem 1. Then, for each unmarked sub-edge leaving a cycle, we extend a new inward alternating path in the same manner as we did for end vertices of paths. As a result, additionally to the \( N \) sub-edges incident to each path of \( M \), all sub-edges leaving cycles are marked to be ends of alternating paths. We swap certain edges in order to prevent a problematic type of path as follows.

Let \( Q \in \bar{Q} \) be a truncated alternating path of length one starting from \( s \) such that its sub-edge \( e \) is part of a 1-edge \( \{ s, u \} \). Let \( e' \) be the second sub-edge of \( \{ s, u \} \). By the construction of \( \bar{Q} \), \( e' \) is in a (truncated) alternating path \( Q' \in \bar{Q} \) and \( Q' \) does not stop at \( u \).

By Lemma 5 there is a third sub-edge \( e'' \notin E(M) \) incident to \( s \) and \( e'' \) is not a sub-edge of a 1-edge. If \( e'' \) is not marked, we remove \( Q \) from \( \bar{Q} \) and extend a new inward alternating path from \( e'' \). Otherwise, we recombine the alternating paths incident to \( s \). Let \( Q'' \) be the alternating path containing \( e'' \). If \( Q'' \) ends in \( s \), \( Q' \) does not since there are exactly two alternating paths ending in \( s \). Then we cut \( Q' \) at \( s \). We extend the part of \( Q' \) not containing \( e' \) with \( e \) and \( Q \) becomes the remaining path of \( Q' \) starting with \( e' \). If \( Q'' \) does not end in \( s \), we cut \( Q'' \) at \( s \) and handle \( Q'' \) analogous to \( Q' \) in the previous case. We apply these modifications to all “bad” alternating paths in \( \bar{Q} \) and update the marking of edges.

Since we only consider edges in the support of \( x^* \), for each edge \( e \) such that a sub-edge of \( e \) is in \( Q \), \( z_e = x^*_e = 1/2 \).
Let us construct a solution \( y \) to \( \text{LP}(x^*) \). The types of assignments are performed in the order as listed, that is, each type of assignment is applied iteratively until there is no further occurrence of that type. As in Theorem 1, we argue via the extended version of \( \text{LP}(x^*) \) for sub-edges. For each cycle \( C \) of \( M \) we set \( y_{C,e} = \lceil x_e^* \rceil \) for all sub-edges \( e \) incident to vertices of \( C \). If \( |V(C)| \geq 5 \), by Lemma 9, we have assigned a sufficient amount to satisfy the constraints \( (5) \) of \( \text{LP}(x^*) \).

If \( |V(C)| = 4 \), there are at least two different (truncated) alternating paths \( Q, Q' \) starting from \( C \) with the edges \( e_1, e_2 \) resp. \( e_1', e_2' \), since there are at least 4 sub-edges leaving \( C \) and there are no truncated alternating paths of length one starting from cycles. Then both \( e_2 \) and \( e_2' \) are matching edges. We set \( y_{C,e_2} = y_{C,e_2'} = 1 \) and thus, by Lemma 9, we have assigned a sufficient amount to satisfy the constraints \( (5) \) of \( \text{LP}(x^*) \).

Similar to 4-cycles, if \( |V(C)| = 3 \), for each alternating path \( Q \) starting from \( C \) with \( e_1, e_2 \), we set \( y_{C,e_2} = y_{C,e_2'} = 1 \). However, the assignment does not provide a guarantee that sufficiently many sub-edges were assigned to \( C \). We will come back to cycles of length three after discussing paths of \( M \), since this way it is easier to exclude some of the possibilities.

For the remaining cases, let us fix an end vertex \( s \) of a path \( P_s \) in \( M \) and a path \( Q \in Q \) starting from \( s \). Let \( s' \) be the vertex adjacent to \( s \) in \( Q \). We aim to find a set of 6 sub-edges for \( s \) such that for each sub-edge \( e \) in the set, \( y_{P_s,e} = 1 \).

If \( Q \) is a path of length at least 3 starting with the sub-edges \( e_1, e_2, e_3 \), we set \( y_{P_s,e_1} = y_{P_s,e_2} = y_{P_s,e_3} = 1 \).

Suppose \( Q \) leads to the end vertex of a path. Then its length is at least 7, by Lemma 5(c) and 5(d), and we did not assign values twice (and therefore, in particular, we did not violate the constraints \( (3) \)).

Suppose \( Q \) leads to a cycle. Then its length is at least 5, by Lemma 5(b), and we did not assign values twice.

Suppose \( Q \) leads from \( s \) to \( s \) and its length is at least 7. Then again we did not assign values twice.

Suppose \( Q \) leads from \( s \) to \( s \) and its length is exactly 5. Then we assign \( y_{P_s,e} = 1 \) for all \( e \) in \( Q \) and we still have to find an additional sub-edge. Let us consider the edge \( \{s, s'\} \). If \( x_{\{s, s'\}} = 1 \), none of its sub-edges is in any path of \( Q \). We choose one of the two sub-edges, \( e \), such that \( y_{P_s,e} = 0 \) for all paths \( P \) in \( M \) and set \( y_{P_s,e} = 1 \). Note that we that we consider \( \{s, s'\} \) at most twice and therefore we can always find such a sub-edge \( e \).

Otherwise, if \( \{s, s'\} \) has only one sub-edge, there is a sub-edge \( e' \in \delta(v) \) that is neither a sub-edge of \( \{s, s'\} \) nor in \( Q \). Then we set \( y_{P_s,e} = 1 \). We have to exclude that \( e' \) was used for a previous assignment. If \( e' \) is in no alternating path of \( Q' \), a double use would directly imply the existence of a basic improvement. Otherwise, let \( Q' \in Q \) be the path containing \( e' \). If \( s \) is an end vertex of \( Q' \), there is nothing else to do since in this case, \( e' \) was assigned twice for \( s \). Otherwise let \( Q'_1 \) and \( Q'_2 \) be the two sub-paths of \( Q' \) that end at \( s \) such that \( e' \) is in \( Q'_2 \). Note that \( Q'_2 \) itself is a (truncated) inward alternating path. By Lemma 6(b), if \( Q'_1 \) leads to an end vertex, either its length is at least two and it leads to a cycle, or the length is at least three and it leads to the end of a path in \( M \). In all cases, we did not assign \( e' \) to any component yet.

Suppose \( Q \) leads from \( s \) to \( s \) and its length is at exactly 3. Let \( s, u, v \) be the three vertices of \( Q \) where \( \{u, v\} \) is an edge of a path \( P \) in \( M \). Lemma 7 implies that neither \( u \) nor \( v \) is an end vertex of \( Q \) and none of the sub-edges incident to \( u \) or \( v \) in \( P \) is used by any of the assignments considered until now. Let us consider the set \( S \) of sub-edges incident to \( u \) or \( v \) within \( Q \) or \( P \). Then \( |S| \geq 5 \) and for each \( e \in S \), we set \( y_{P_s,e} = 1 \). If \( |S| = 5 \), we assign one more sub-edge to \( P_s \) analogously to the previous case. Note that multiple assignments of this type cannot lead to multiple assignments of sub-edges, since otherwise there is an inward alternating path of length three.

Suppose \( Q = e \) is a truncated alternating path of length one. Then, due to its priority in the construction of \( Q \) and the transformation of \( Q \), \( Q \) is incident to a 1-edge \( e' \) of \( P_s \) in
direction towards $s$. Note that $e'$ is not at either end of $P_s$ and, since $Q$ is a single edge to an end vertex, $e'$ has not been used by any of the previous assignments (since we have Lemma 7). We set $y_{P_s,e''} = 1$ for each of the two sub-edges $e''$ of $e'$ and we set $y_{P_s,e} = 1$. Note that $e''$ is not assigned due to two different truncated alternating paths of length one, since otherwise there is an inward alternating path of length three considered in Lemma 5.

Suppose that $Q = (e,e')$ is an alternating path of length exactly two. We assign the values similar to the proof of Theorem 1. That is, we set $y_{P_s,e''} = y_{P_s,e'} = 1$. If we find a sub-edge $e''$ adjacent to $e'$ that was not yet assigned (see the proof of Theorem 1), we set $y_{P_s,e''} = 1$.

Suppose that we were not able to assign three sub-edges. Then, as we have seen in the proof of Theorem 1, we can find an alternating path $Q$ of length 5 from $s$ to some end vertex $t$ such that the first and last edge of $Q$ does not violate the inward property. If $s = t$, the situation is exactly the same as for inward alternating paths of length 5 from $s$ to $s$ considered above. Otherwise, as we have seen in the proof of Theorem 1, we may assume that all vertices of $Q$ are in $P_s$ and the neighbor of $s$ in $P_s$ is not adjacent to the end vertex of another path. We distinguish two possibilities. If $Q$ is the only truncated alternating path of length 2 starting from $s$, we assign the value of a third edge in the same way as we did for the last edge in the case of inward alternating paths from $s$ to $s$ of length 5. Otherwise, we conclude that there is a second alternating path $Q'$ of length 5 from $s$ to $t$ and for $t$, we already have assigned 6 sub-edges from $Q$ and $Q'$ to $P_s$. For each of the two alternating paths, we assign the value of a third edge in the same way as we did for the last edge in the case of inward alternating paths from $s$ to $s$ of length 5, one of them incident to $s$ and the other one incident to $t$.

To see that for paths in $M$ the constraints (5) are satisfied for $\alpha = 6$, let us check the cases above. For each end vertex $s$ of a path $P$, we either directly found 6 sub-edges whose values were assigned to $P$ or, since the degree of $s$ is at least three, we assigned two times the values of 3 sub-edges to $P$. Since $P$ has two end vertices, in total, we assigned the values of at least 12 sub-edges to $P$. Each assignment $y_{C,e'} = 1$ for a component $C$ and a sub-edge $e'$ of some edge $e$ increases $y_{C,e}$ by $1/2$, which sums up to at least 6 for each component $C$.

Finally, we continue the discussion where $C$ is a cycle of length three. By the subtour elimination constraints, there are at least 4 alternating paths starting from $C$. Therefore, if no sub-edge $e_2$ was used twice, we have assigned 4 an amount of at least two to $C$ additionally to the previously assigned value of at least 4 (by Lemma 9).

Otherwise, an edge $e_2$ was assigned twice, which implies that there is an alternating path from $C$ to $C$ of length exactly three. By Lemma 5(1), such an alternating path cannot have two different end vertices. We therefore name the vertices such that $V(C) = \{v_1,v_2,v_3\}$ and there is an alternating path $Q$ of length three from $v_1$ to $v_1$. Then $\deg_{G_s'}(v_1) = 4$ and therefore $x^*_{\{v_1,v_2\}} = x^*_{\{v_1,v_3\}} = 1/2$.

If also $x^*_{\{v_2,v_3\}} = 1/2$, then $\deg_{G_s'}(v_2) = \deg_{G_s'}(v_3) = 4$ and there are 6 alternating paths starting from $C$ with at most three edges assigned twice. Then the total amount of values assigned to $C$ is at least 6, which is sufficient.

The remaining case is that $x^*_{\{v_2,v_3\}} = 1$, in which case we have assigned 11 sub-edges resp. an amount of 5.5 to $C$ (see also Fig. 5). Thus, we aim to assign one more sub-edge to $C$. Let $u, u'$ be the two remaining vertices of $Q$ and let $P$ be the path of $M$ that contains $u$ and $u'$. By Lemma 5(1), there is no inward alternating path of length less than 7 from $v_1$ to another end vertex and therefore there are vertices $w, w' \in V(P)$ such that $\{w,u\}, \{u,u'\}, \{w',u'\} \in E(P)$. By the definition of $z$, $x^*_{\{u,u'\}} = 1/2$. If either $u$ or $u'$ is incident to a 1-edges $e'$, we choose one of its two sub-edges $e'_1$ such that $y_{C,e'_1} = 0$ for all components $C'$ and set $y_{C,e'_1} = 1$. Since by our previous discussion $e'$ is considered at most twice (once from each side), we can always find such a sub-edge $e'_1$. Therefore, we either have assigned the additional sub-edge or there are vertices $s, s'$ such that there are edges $\{u,s\}, \{u',s'\} \in E(G_s')$.

Let us recall the discussion of inward alternating paths of length three for paths. The only difference to paths is that a new cycle $C'$ without changing the number of components does not
necessarily provide an improvement, since $C$ becomes part of a path. There is an improvement if there is a newly created cycle that has a length of more than three and thus the number of vertices in cycles is increased without decreasing the number of cycles or increasing the number of components. Since alternating paths of length three from $v_1$ to another vertex than $v_1$ cannot introduce more than one cycle, we conclude that $s, s'$ are the end vertices of $P$ and $\{s, w\}, \{s', w'\} \in E(P)$. Note that any additional edge within $V(C) \cup V(P)$ allows us to reduce the number of components via an application of an alternating path. Furthermore, we can transform $C$ and $P$ into a path and a cycle $\{s, u, w\}$ or $\{s', u', w'\}$, and the remaining vertices form a path satisfying the conditions of Lemma 8. In the algorithm, we consider additional improvements that appear due to longer cycles as well as all improvements of Lemma 5, taking into account the three transformations. The total number of considered improvements for all appearances of the special situation only increases by a polynomial factor. Also, we can exclude that any edge in distance two from $P$ or $C$ has been assigned to any component except $P$ or $C$.

By Lemma 9, there are at least 20 sub-edges incident to the 9 vertices $V(P) \cup V(C)$, and we have additionally assigned at least two sub-edges due to $v_2$ and $v_3$ and two sub-edges due to $s$ and $s'$. The total number of 24 sub-edges is sufficient to assign 12 sub-edges to each of the two components $P$ and $C$.

![Figure 5: Special case of three cycles. The dashed edges have an LP-value of 1/2.](image_url)

### 7 Subcubic Fractionally Hamiltonian Supports

In this section we obtain tight integrality gaps, and an improved approximation guarantee, for $(1, 2)$-STSP instances $G$ which admit optimal basic solution with subcubic support of SER ($G$) exists.

**Theorem 3.** For the class of instances of $(1, 2)$-STSP admitting an optimal basic solution to SER($n, G$) with subcubic support, the tight upper bound on the integrality gap is $10/9$.

The argumentation to prove this theorem differs considerably from the previous two, because we do not assign the LP values to the components. Instead, we charge every vertex with a coin and redistribute these coins fractionally such that each component obtains at least 9 coins. The basic idea is that cycles in the 2-matching are well-behaved and that for each path we can collect two coins for each edge leaving an end vertex. However, the complications arise from the interferences if there are edges from the end of a path to the path itself.

Before we present the actual result of this subsection, let us observe some properties of cycles in a 2-matchings $M$ within the support graph.

For each component $C$ of $M$, let $\text{out}(C)$ be the set of edges in $\delta(V(C))$ incident to end vertices of $C$.

**Lemma 10.** Let $G_x$ be a subcubic support graph of an optimal basic solution $x$ of SER($n, G$), and let $C$ be a cycle of a 2-matching $M$ in $G_x$ with $|V(C)| \leq 6$. Then $|\text{out}(C)| \geq 3$ and there are two vertices $u, v \in V(C)$ such that $u$ and $v$ are path-forming and $|\text{out}(C) \cap \delta_{G_x}(\{u, v\})| = 2$.

**Proof.** Clearly $|\text{out}(C)| \geq 2$, since the constraints of SER imply that $x_e \leq 1$ for each edge $e \in E(G)$, unless $G$ consists of only 2 vertices (if $x_e > 1$, $x(\delta_{G_x}(V(e))) < 2$). For the sake
of contradiction, suppose that \(|\text{out}(C)| < 3\) and let \(s\) and \(t\) be the two vertices with incident edges leaving \(C\). Then the degree of \(s\) and \(t\) is 3 and the two edges leaving \(C\) are 1-edges since \(x(\delta_{G_x}(V(C))) \geq 2\). As a consequence, the edges incident to \(s\) and \(t\) within \(C\) cannot be 1-edges (due to the equality constraints). Since a vertex in \(G_x\) can only have degree 2 if it is incident to two 1-edges, all adjacent vertices of \(s\) and \(t\) have degree 3. However, this implies that there is a chord in \(C\) (an edge \(e \notin E(C)\) with both ends in \(V(C)\)). The vertices \(s\) and \(t\) cannot be neighbors, since otherwise \(x(\delta_{G_x}(V(C) \setminus \{s, t\})) < 2\). Therefore, \(|V(C)| \notin \{3, 5\}\). If \(|V| = 4\) there is one chord, and if \(|V| = 6\) there are two chords. It is not hard to check that in all possible arrangements, the solution is a convex combination of two paths from \(s\) to \(t\), contradicting that \(x\) is a basic solution.

The second claim follows immediately from our considerations since either there are two neighbors within \(C\) that both have neighbors in \(V(G_x) \setminus V(C)\) or \(C\) has chords. \(\square\)

For the proof of Theorem 3 let \(x\) be an optimal basic solution to \(\text{SER}(n, G)\) with subcubic support graph \(G_x\). We write \(\delta\) as shorthand for \(\delta_{G_x}\).

We start with running Algorithm 1 without the line marked by * and obtain a 2-matching \(M\) of the support graph \(G_x\) for which none of the considered improvements are possible. However, within the proof we use some types of improvements that are quite specific and therefore we did not include them into Lemma 5. We will argue for these improvements that we do not violate Lemma 8 if we consider them in Algorithm 1.

For each vertex \(v \in V(G_x)\), let \(\text{comp}_M(v)\) denote the component of \(M\) that \(v\) belongs to (that is, the component of \(M\) that has either one or two edges incident to \(v\)), and call each vertex \(v' \in V(G_x)\) that is adjacent to \(v\) in \(M\) a component neighbor of \(v\) (with respect to \(M\)).

We now analyze the number of components in \(M\). To this end, we put coins on each vertex, and redistribute these coins according to certain rules. Initially, we assign one coin to each vertex of \(G\). Then we redistribute the coins fractionally to the components of \(M\). We show how to assign at least nine coins to each component of \(M\).

### 7.1 Cycles

We first show how to assign at least nine coins to each component \(C\) of \(M\) that is a cycle. Let \(N_{\text{end}}(C) = \{w \in V(G) \setminus V(C) : \{v, w\} \in \delta(V(C))\}\) be the set of vertices that are neighbors of \(C\). We define a set \(S\) depending on the size of \(C\):

- If \(|V(C)| \geq 7\), set \(S := \{v\}\) for each vertex \(v \in V(S)\) that has some neighbor outside \(C\).
- If \(|V(C)| \in \{4, 5, 6\}\), set \(S := \{u, v\}\) for the two vertices considered in Lemma 10.
- If \(|V(C)| = 3\), set \(S := V(C)\).

Let \(S' := \{v \in N_{\text{end}}(C) : \{v, w\} \in \delta(S)\text{ for some }w \in V(G_x)\}\) be its set of neighbors not in \(C\).

Recall that each vertex \(v\) in the support graph is equipped with exactly one coin. Now we redistribute these coins and halves of them to all cycles \(C\), according to the following rules:

**Rule C1:** each vertex \(v \in V(C)\) assigns its whole coin to \(C\).

**Rule C2:** each component neighbor \(w\) of \(S'\) assigns 1/2 of its coin to \(C\).

Additionally, if \(|V(C)| = 4\), let \(s \in V(C) \setminus S\) be such that \(\{s, t\} \in E(G_x)\) for \(t \notin V(C)\). By Lemma 10 such a vertex \(s\) exists.

**Rule C3:** If \(t\) is not a component neighbor of some vertex in \(S'\), assign the coin of \(t\) to \(C\).

**Rule C4:** If \(t\) is a component neighbor of some vertex in \(S'\), assign half a coin of \(t\) to \(C\), and assign half a coin from \(t\)'s component neighbor \(w \notin S'\) to \(C\).
Next we show that all considered vertices exist, and that we did not assign more coins than available.

First, notice that all vertices in \( S' \) are internal vertices of paths, for otherwise there is a basic improvement because there is an alternating path of length one between two end vertices.

Second, fix a cycle \( C' \neq C \) in \( M \). Let \( u \in N_{\text{end}}(C) \) and \( u' \in N_{\text{end}}(C') \), and let \( v \in V(C), v' \in V(C') \) be such that \( \{v, w\} \in \delta(V(C)), \{v', w'\} \in \delta(V(C')) \). Notice that \( w \notin V(C') \) and, symmetrically, \( w' \notin V(C) \) as otherwise there is a basic improvement (an alternating path of length one between \( C \) and \( C' \)). Now, for any path \( P \) in \( M \) that contains both \( w, w' \), the two vertices cannot be consecutive in \( P \), for otherwise there is an inward alternating path of length 3, which is excluded due to Lemma 5(d).

Third, let \( w, w' \in N_{\text{end}}(C) \), and let \( v, v' \in V(C) \) be such that \( \{v, w\}, \{v', w'\} \in \delta(V(C)) \). Now, for any path \( P \) in \( M \) that contains both \( w, w' \), if \( v \) and \( v' \) are path-forming, these two vertices cannot be consecutive on \( P \). Otherwise, there is an inward alternating path of length 3, which is excluded due to Lemma 5(d).

Fourth, let \( C \) be a cycle with \( |V(C)| = 4 \) and let \( \{u, v, s\} \subset V(C) \) be the three vertices considered above. Let \( u', v', s' \) their neighbors in \( N_{\text{end}}(C) \) where \( S' = \{u', v'\} \). Then \( s' \) cannot be incident to both \( u' \) and \( v' \), since either \( s \) or \( u \) or \( s \) and \( v \) are path forming, a situation considered above.

With these four observations, clearly all assigned coins are available. Note that each cycle obtained at least 9 coins.

We conclude this subsection with an observation that follows immediate from the four observations above and that will be useful for assigning coins to paths.

**Observation 3.** Let \( C \) be a cycle in \( M \) and let \( u, v \in V(C) \) be two vertices that are not path-forming. Let \( u', v' \) be the neighbors of \( u \) and \( v \) not in \( V(C) \). Then

- the total number of coins assigned to \( C \) by \( u \) and \( v \) is at most 3, and
- half a coin of each component neighbor of either \( u' \) or of \( v' \) has not been assigned to any cycle, unless \( u' \) or \( v' \) has a neighbor \( w' \) such that the whole coin of \( w' \) was assigned to \( C \),

### 7.2 Paths

Now we are dealing with paths \( P, P', P'' \ldots \) that form components of the 2-matching \( M \). For a path \( P \), let \( v_1, u_1, v_2, \ldots, u_t, v_2 \) be an ordering of its such that \( \{v_1, u_1\}, \{u_t, v_2\} \) and \( \{u_i, u_{i+1}\} \) for \( i = 1, \ldots, t - 1 \) are exactly the edges of \( P \). Hence, \( v_1, v_2 \) are the two end vertices of \( P \), which form the set \( \text{end}(P) \). By \( \text{int}(P) = \{u_1, \ldots, u_t\} \) we denote the set of internal vertices of \( P \).

Our goal is to show that each path \( P \) of \( M \) can receive at least 9 coins, where \( P \) receives coins according to the following rules:

**Rule P1:** Every end vertex \( v \in \text{end}(P) \) sends 1/2 coin to the path \( P \).

**Rule P2:** Every internal vertex \( v \in \text{int}(P') \), for \( P' \neq P \), that is adjacent to an end vertex of \( P \), sends 1 coin to \( P \).

**Rule P3:** Every vertex \( v \in \text{int}(P') \cup \text{end}(P') \), for \( P' \neq P \), that is not adjacent to an end vertex of \( P \) but is adjacent to a vertex covered by Rule P2, sends 1/2 coin to \( P \).

**Rule P4:** Every other vertex \( v \) sends its coin to the path \( \text{comp}_M(v) \) that it belongs to.

Notice that these half coins have not been assigned to cycles or to other paths before since otherwise there is a basic improvement. They may, however have been assigned to the same path twice. We will ensure that we do not double-count them.

To analyze the redistribution of coins, we use the tool of alternating paths, which we denote by \( Q, Q', Q'', \ldots \). For each type of path \( P \) that can occur, we show that it can receive at least 9
coins from its internal vertices, its end vertices and vertices of paths $P' \neq P$ connected to $P$ by connecting edges, for otherwise we can find an improvement $M'$ of $M$ by exhibiting some alternating paths.

Similar to cycles, let $N_{\text{end}}(P) = \{w \in V \setminus V(P) \mid \{v, w\} \in \delta(\text{end}(P), V \setminus V(P))\}$ be the set of vertices in components of $M$ other than $P$ that are adjacent to end vertices of $P$. As was the case with cycles, $N_{\text{end}}(P)$ only contains internal vertices of paths. Recall that the degree of the vertices in $\text{end}(P)$ is exactly three, by Lemma 5(d). We distinguish five cases, for $|\text{out}(P)| \in \{0, 1, 2, 3, 4\}$. Due to the degree restriction, $|\text{out}(P)| \leq 4$.

**Case 1:** $|\text{out}(P)| = 4$.

**Case 1.1:** The four vertices in $N_{\text{end}}(P)$ form an independent set.

Then the path $P$ receives at least 9 coins, namely $2 \times 1/2 = 1$ coin from its two end vertices, and then $4 \times (1 + 1/2 + 1/2) = 8$ coins from each of the four vertices in $N_{\text{end}}(P)$ and their respective neighbors in components of $M$ other than $P$.

**Case 1.2:** There are two vertices $w_1, w_2 \in N_{\text{end}}(P)$ that are adjacent in $M$.

**Case 1.2.1:** The vertices $w_1, w_2$ are adjacent to distinct vertices in $N_{\text{end}}(P)$.

Then there is an inward alternating path of length three considered by Lemma 5(d).

**Case 1.2.2:** Vertices $w_1, w_2$ are in component $C$ and adjacent to the same end vertex $v_1 \in \text{end}(P)$.

Then we can add $v_1$ to $C$, which makes the neighbor $u \in \{u_1, u_t\}$ of $v_1$ in $P$ an end vertex. In particular, if there is an edge $e$ incident to $u$ and an end vertex such that $e$ is not in $P$, there is an improvement without creating new end vertices, a contradiction.

We assign to $P$ the two coins of $w_1, w_2$ the two half coins of the two component neighbors of $w_1, w_2$, $1/2$ coin of $v_1$, and half a coin of the component neighbor of $v_1$ to $P$. Thus, the two edges gain a total of 4 coins for $P$.

This finishes [Case 1.2] because in either subcase 1.2.1 and 1.2.2 we can assign at least 8 coins to $P$ additionally to the one coin that $P$ had already.

**Case 2:** $|\text{out}(P)| = 3$.

**Case 2.1:** The three vertices in $N_{\text{end}}(P)$ form an independent set.

Then $P$ collects $3 \times 2 = 6$ coins from vertices outside $P$ analogous to [Case 1] and $2 \times 1/2$ coins from its end vertices. Assume, without loss of generality, that $v_1 \in \text{end}(P)$ is the end vertex of $P$ for which there is an internal vertex $u_i$ in $P$ such that $\{v_1, u_i\}$ is an edge in $G_x$ for some $i \in \{2, \ldots, t\}$. Hence, $P$ collects $1/2$ coin from $u_i$. Thus, $P$ receives at least $7 \frac{1}{2}$ coins, and needs to collect $1 \frac{1}{2}$ more coins.

**Case 2.1.1:** There is a connecting edge $e \in E(G_x) \setminus E(M)$ with $e = \{z, w\}$ for $z \in \{u_{i-1}, u_{i+1}\}$ such that $w$ is an end vertex of a path $P' \neq P$.

In this case, $Q = (v_1, u_i, z, w)$ is an alternating path of length $3 < 5$, with end vertices $s = v_1$ and $t = w$ of different paths. Thus, by Lemma 5(d), there is an improvement of $M$, a contradiction.

**Case 2.1.2:** There is a connecting edge $e \in E(G_x) \setminus E(M)$ with $e = \{u_{i-1}, w\}$ such that $w$ is an end vertex of a cycle.

In this case, $Q = (v_1, u_i, u_{i-1}, w)$ is an inward alternating path of length $3 < 5$, with end vertices $s = v_1$ and $t = w$ of different paths. Thus, by Lemma 5(d), there is an improvement of $M$, a contradiction.

**Case 2.1.3:** There is a connecting edge $e \in E(G_x) \setminus E(M)$ with $e = \{u_{i+1}, w\}$ such that $w$ is an end vertex of a cycle and the previous cases do not apply.

**Case 2.1.3.1:** There is no connecting edge $e' \in E(G_x) \setminus E(M)$ with $e = \{u_j, w'\}$ for $j < i$ such that $w'$ is an end vertex of some component.

Then none of the coins between $v_1$ and $u_i$ has been assigned. We assign the remaining half coin of $v_1$, and one coin of $u_{i-1}$ to $P$. Therefore total number of coins assigned to $P$ is 9.

**Case 2.1.3.2:** There is a connecting edge $e' \in E(G_x) \setminus E(M)$ with $e = \{u_j, w'\}$ for some $j < i$ such that $w'$ is an end vertex of some component.
Case 2.1.3.2.1: For some choice of \( j \) and \( j' \), \( j < i < j' \), such that \( w' \) and \( w'' \) are end vertices adjacent to \( u_j \) and \( u_{i'} \), both \( w', w'' \) are not path-forming and belong to one cycle \( C \).

Then, by Observation 3, at most 3 coins have been sent to \( C \) due to \( w' \) and \( w'' \) and therefore the component neighbors of \( u_j \) and \( u_{j'} \) still have together at least two half coins that have not yet been assigned. We assign these two half coins to \( P \) and additionally half a coin of \( u_{i-1} \).

Again, the total number of coins assigned to \( P \) is 9.

Case 2.1.3.2.2: Otherwise, there is an improvement by including the edges \( \{v_1, u_1\}, \{u_j, w'\} \) and \( \{u_{i+1}, w\} \) and removing up to four edges.

Notice that for each removal we can choose an edge such that either we do not create a new end vertex or the removed edge is not a 1-edge. Therefore we can keep up the invariant of Lemma 8.

Case 2.1.4: No such connecting edge \( e \) exists.

Then \( P \) also receives \( 2 \times 1/2 = 1 \) coin from the two neighbors of \( u_i \) and the whole coin of \( u_{i-1} \), and hence receives 9 coins in total.

Case 2.2: There are two vertices \( w_1, w_2 \in N_{\text{end}}(P) \) that are adjacent in \( M \).

Case 2.2.1: Vertices \( w_1, w_2 \) are adjacent to distinct vertices in \( N_{\text{end}}(P) \).

Then there is an improvement by removing the edge \( \{w_1, w_2\} \) and adding the two edges from \( P \) to \( w_1 \) and \( w_2 \).

Case 2.2.2: Vertices \( w_1, w_2 \) are in component \( C \) and adjacent to the same end vertex \( v_i \in \text{end}(P) \).

Then we can add \( v_i \) to \( C \), which makes the neighbor \( u \in \{u_1, u_i\} \) of \( v_i \) in \( P \) an end vertex. In particular, if there is an edge \( e \) incident to \( u \) and an end vertex such that \( e \) is not in \( P \), there is an improvement, a contradiction.

We assign to \( P \) the two coins of \( w_1, w_2 \), the two half coins of the two component neighbors of \( w_1, w_2 \), 1/2 coin of \( v_i \), and half a coin of the component neighbor of \( v_i \) to \( P \). Thus, the two edges gain a total of 4 coins for \( P \). Additionally, \( P \) collects the remaining 5 coins analogously to Case 2.2 where a collected coin from \( v_i \) implies that there is an improvement. This finishes Case 2.2 because \( P \) can collect 9 coins.

Case 3: \( |\text{out}(P)| = 2 \).

Then there are two edges \( e_1, e_2 \in E(G_x) \setminus M \) that each have one endpoint in \( \text{end}(P) \) and its other endpoint in \( \text{int}(P) \), and their endpoints in \( \text{int}(P) \) are distinct. We analyze the possibilities how these edges may interfere with each other. Let \( e_1 = \{v, u_i\} \) and \( e_2 = \{v', u_j\} \), where \( v, v' \in \text{end}(P) \) with possibly \( v = v' \), and \( u_i, u_j \in \text{int}(P) \) with \( u_i \neq u_j \).

Analogous to Case 1 since \( |\text{out}(P)| = 2 \), the path \( P \) already collects \( 2 \times 2 = 4 \) coins from vertices outside \( P \) and \( 2 \times 1/2 = 1 \) coin from its two end vertices.

Case 3.1: Vertex \( v = v' \).

By renaming we may assume that \( v = v_1 \).

Case 3.1.1: Vertices \( u_i, u_j \) are adjacent in \( P \).

We say that a vertex \( w \) in \( P \) acts as an end vertex if there is a path \( P' \) in \( G_x \) with \( V(P) = V(P') \) such that \( w \) is an end vertex of \( P' \). We only consider simple transformations that can be done efficiently. Then the neighbor \( u_1 \) of \( v \) in \( P \) acts as an end vertex. Let \( v \) be closer to \( u_i \) than to \( u_j \) in \( P \) (that is, \( i < j \)). Then we assign to \( P \) the full coin of \( v \) and \( u_i \), half a coin of \( u_j \), half a coin of the neighbor \( u_1 \) of \( v \) in \( P \), and half a coin of \( u_{i-1} \). All of these coins can be assigned unless there is an inward alternating path of length three from \( v_1 \) via \( u_i \) and \( u_{i-1} \). There is one more coin that we have to assign. That is, if a coin that acts as an end vertex is adjacent to an end vertex of another component, there is an improvement without creating new end vertices.

Case 3.1.1.1: The subpath of \( P \) from \( v \) to \( u_j \) has at least five vertices.

Then the analysis is analogous to Case 2 that is, unless there is an improvement we either can assign sufficiently many coins of the sub-path from \( v_1 \) to \( u_i \) to \( P \) or we can assign half a coin of both \( u_j \) and \( u_{j+1} \) to \( P \).
Case 3.1.1.2: The subpath of $P$ from $v$ to $u_j$ has exactly four vertices.
Then all vertices of $P$ but $u_j$ act as end vertices, and by the LP constraints, there has to be at least one more edge leaving the subpath. If the destination of that edge is an end vertex, we have found an inward alternating path of length at most three, which is excluded by Lemma 5(d).
Otherwise, that vertex still has its coin an we transfer the coin to $P$.

Case 3.1.2: The vertices $u_i,u_j$ are not adjacent in $P$.
The analysis is analogous to Case 2 but the argument has to be applied for $u_i$ and $u_j$ separately.

This finishes Case 3.1 since $P$ was able to collect 4 coins additionally to the 5 initially collected coins.

Case 3.2: Vertex $v \neq v'$.
Let $u_i,u_j$ be internal vertices of $P$ such that $\{v_1,u_i\},\{v_2,u_j\}$ are edges of $E(G_x) \setminus E(M)$.

Case 3.2.1: $u_i$ and $u_j$ are not consecutive.
This case is analogous to Case 2 applied to $u_i$ and $u_j$ independently.

Case 3.2.2: $u_i$ and $u_j$ are consecutive.
If $i > j$, there is an inward alternating path of length three from $v_1$ to $v_2$, considered in Lemma 5(d). Therefore we may assume $i < j$.

Case 3.2.2.1: There is no $i' < i$ such that there is an edge $\{u_{i'},{w}\}$ for an end vertex $w$ of a component $C \neq P$. Then we assign all coins of the vertices $v_1,u_1,\ldots,u_j$ to $P$ and half a coin from $u_{j+1}$. Clearly, $j \geq 3$ and none of the assigned coins has been assigned previously. Together with the $\frac{3}{2}$ coins already assigned to $P$, the total number of coins adds up to at least 9.

Case 3.2.2.2: There is no $j' > j$ such that there is an edge $\{u_{j'},w\}$ for an end vertex $w$ of a component $C \neq P$.
This case is analogous to Case 3.2.2.1.

Case 3.2.2.3: There is an $i' < i$ and an edge $\{u_{i'},w\}$ for an end vertex $w$ of a component $C \neq P$, and there is a $j' > j$ and an edge $\{u_{j'},w'\}$ for an end vertex $w'$ of a component $C' \neq P$.
This case is analogous to Case 2.1.3.2.

Case 4: $|\text{out}(P)| = 1$.
Then $P$ collects $2 \times 1/2 = 1$ coin from its two end vertices, plus 2 coins from vertices outside $P$.
There are three edges $e_1 = \{v_1,u_i\}, e_2 = \{v_1,u_j\}, e_3 = \{v_2,u_p\} \in E(G_x) \setminus E(M)$ that each have one endpoint in $\text{end}(P)$ and their endpoints in $\text{int}(P)$ are distinct. Hence, $P$ collects $3 \times 1/2 = 3/2$ coins from $u_i,u_j,u_p$. Thus, $P$ needs to collect an additional $4\frac{1}{2}$ coins.

We analyze the possibilities how these edges may interfere with each other. Assume, without loss of generality, that $i < j$. Notice that Case 4 is very similar to Case 3. In particular, we use Case 3.1 in order to assign the coins related to $u_i$ and $u_j$. There are some complications that we address in the following cases.

Notice that any edge $\{u_{j'},v_2\}$ for $j' < j$ behaves the same way as an edge $\{u_{j'},w\}$ for an end vertex $w$ of a component $C \neq P$ with respect to $u_j$. The analogous statements are true for $u_i$ and $u_p$. Therefore the only cases that are not analogous to Case 1 or Case 2 are the following.

Case 4.1: $u_{j+1} = u_p$.
We claim that the coins of $u_j$ and $u_{j+1}$ have not been assigned twice. Notice that assigning the whole coin of $u_j$ to $P$ while considering $v_1,u_i,u_j$ implies that there is a $j' < j$ such that $j'$ is adjacent to an end vertex of another component $C$. Then the remaining argument is analogous to Case 2.1.3.2.

Case 4.2: $u_{i+1} = u_p$.
This case is analogous to Case 4.1.

Case 5: $\text{out}(P) = \emptyset$.
Then $P$ collects $2 \times 1/2 = 1$ coin from its two end vertices. Then there are four edges $e_1 = \{v_1,u_i\}, e_2 = \{v_1,u_j\}, e_3 = \{v_2,u_p\}, e_4 = \{v_2,u_q\} \in E(G_x) \setminus E(M)$ that each have one endpoint
in end(P) and their endpoints in int(P) are distinct. Hence, P collects 4 \times 1/2 = 2 coins from \( u_i, u_j, u_h, u_q \). Thus, P needs to collect an additional 6 coins.

We analyze the possibilities how these edges may interfere with each other. Assume, without loss of generality, that \( i < j \), and (by symmetry) that \( p > q \). Notice that Case 5 is very similar to Case 3. In particular, we use Case 3.1 in order to assign the coins related to \( u_i \) and \( u_j \), as well as to assign the coins related to \( u_p, u_q \). There are some complications that we address in the following cases.

Notice that any edge \( \{u_j', v_2\} \) for \( j' < j \) behaves the same way as an edge \( \{u_j', w\} \) for an end vertex \( w \) of a component \( C \neq P \) with respect to \( u_j \). The analogous statements are true for \( u_i, u_p, \) and \( u_q \). Therefore, the only cases that are not analogous to Case 1 or Case 2 are the following.

**Case 5.1:** \( u_{j+1} \in \{u_p, u_q\} \).

We claim that the coins of \( u_j \) and \( u_{j+1} \) have not been assigned twice. Notice that assigning the whole coin of \( u_j \) to \( P \) while considering \( v_1, u_i, u_j \) implies that there is a \( j' < j \) such that \( j' \) is adjacent to an end vertex of another component \( C \). The analogous statement is true for the coin of \( u_q \) and therefore there is a \( q' > q \) adjacent to an end vertex of a component \( C' \neq P \). Then the remaining argument is analogous to Case 2.1.3.2.

**Case 5.2:** \( u_{i+1} \in \{u_p, u_q\} \).

This case is analogous to Case 5.1.

\[ \square \]

### 8 Asymmetric \((1,2)\)-TSP

In this section, we consider linear programming-based approximations for \((1,2)\)-ATSP.

**Theorem 4.** There is a polynomial-time 3/2 approximation algorithm for \((1,2)\)-ATSP with respect to Opt\(_{\text{SER}}(n, G)\).

**Proof.** Let \( x^* \) be an optimal solution to SER\((n, G)\) and let \( z \) be as in the proof of Theorem \[1\]. Again, we subdivide the arcs into sub-arcs with respect to \( z \) just as we did with the edges in the proof of Theorem \[1\]. Let \( M \) be a directed 2-matching of \( G_{x^*} \) such that that all 1-arcs are in \( M \). We use directed alternating paths as defined in the proof of Lemma \[4\].

We use an algorithm similar to Algorithm \[4\]. However, instead of Lemma \[5\] and Lemma \[6\] we only use the simple observation that if there is an arc from some end vertex to some start vertex, there is an improvement of \( M \), unless both vertices are in one cycle. Note that in \( M \), there may be cycles of length two.

We construct a set of alternating paths \( Q_3 \). Write \( \delta \) short for \( \delta_{G,x^*} \). Initially, all sub-arcs in cycles are marked and all remaining sub-arcs are unmarked. Then we proceed as follows:

1. Choose an end vertex \( t \) with an unmarked sub-arc \( a \in \delta^+(t) \).

2. Extend \( a \) to a directed alternating path \( Q \) of length three, using two further unmarked sub-arcs.

3. Mark the three sub-arcs of \( Q \).

Similar to the proof of Theorem \[4\] we obtain \( Q_3 \) by applying the procedure iteratively until all sub-arcs that start from end vertices are marked.

Let us first verify that each of the alternating paths can be extended to length three. There are two reasons that could possibly prevent an extension: there is no sub-arc that is part of a feasible alternating path or all sub-arcs that are feasible candidates are marked already.

Let \( a = (t, u) \) be the first sub-arc of some alternating path. Since adding \( a \) to \( M \) does not lead to an improvement, \( u \) is not a start-vertex. We obtain

\[
z(\delta^-(u) \setminus E(M)) = x^*(\delta^-(u) \setminus E(M)) = 1 - x^*(\delta^-(u) \cap E(M)) = z(\delta^-(u) \cap E(M)) \ .
\]
In other words, from $u$ one can extend as many alternating path as there are sub-arcs leading to $u$. Let $d' = (v, u)$ be the subsequent arc in $Q$. Then

$$z(\delta^+ (v) \cap E(M)) = 1 - x^*(\delta^+ (v) \cap E(M)) = x^*(\delta^+ (v) \setminus E(M)) = z(\delta^+ (v) \setminus E(M)).$$

Thus there are sufficiently many sub-arc from $v$ to extend $Q$.

Note that in $Q_3$, each arcs leaving end vertices either is arcs within a cycle or it is the first arc of an alternating path of length three.

Let $\text{LP}(x^*)$ be analogous to Theorem 4. For any alternating path starting at some component (cycle or arc), let $a_1, a_2, a_3$ be the three sub-arcs that form $Q$. We set $y_{C, a_1} = y_{C, a_3} = 1$. Since all considered alternating paths are disjoint, clearly we did not violate any constraint from $(0)$.

Since for each end vertex $t$ of a path, $x^*(\delta^+(t)) = 1$ and for each cycle $C$, $x^*(\delta^+(V(C))) \geq 1$, we have assigned at least $2N$ sub-arcs to each component, which is sufficient to satisfy the constraints $(5)$ for $\alpha = 2$.

**Corollary 3.** If we can obtain a half-integral solution to $\text{SER}(n, G)$, there is a polynomial-time $4/3$ approximation algorithm for $(1, 2)$-$\text{ATSP}$ with respect to $\text{Opt}_{\text{SER}}(n, G)$.

*Proof.* We use the same proof as for Theorem 4 except the following. For any alternating path starting at some component (cycle or arc), let $a_1, a_2, a_3$ be the three sub-arcs that form $Q$. We set $y_{C, a_1} = y_{C, a_2} = y_{C, a_3} = 1$.

This is justified since as in the symmetric case, for half-integral instances $x^* = z$.  

\section{Intractability of Local Search for $(1, 2)$-STSP}

In this section we show that searching the $k$-edge change neighborhood is $W[1]$-hard for $(1, 2)$-STSP, which means that finding the best tour in the $k$-edge change neighborhood essentially requires complete search. Our proof follows the lines of the proof by Dénes Marx [15] that local search for metric TSP with three distances is $W[1]$-hard parameterized by the edge change distance. The difference are some small simplifications, and that we only have two distances, one and two. We choose to follow the lines of Marx so that the reader can easily make comparisons.

The local search problem for $(1, 2)$-STSP is, given a tour $T$, to find the best tour in the $k$-edge change neighborhood of $T$, which are those tours that can be reached from the current tour by replacing at most $k$ edges. On the one hand, if $k$ is part of the input, then this problem is NP-hard, as for $k = n$, the problem is equivalent to finding the best possible tour. On the other hand, the problem is polynomial-time solvable for every fixed value of $k$, in $n^{O(k)}$ time by complete search.

The main result of this section is that finding the best tour in the $k$-edge change neighborhood is $W[1]$-hard, which implies that the problem is not fixed-parameter tractable, unless $W[1] = \text{FPT}$ (which would imply subexponential time algorithms for many canonical NP-complete problems).

We consider a tour as a set of $n$ ordered pairs of cities. If $X$ and $Y$ are two tours on the same set of cities, then the distance of $X$ and $Y$ is $|X \setminus Y| = |Y \setminus X|$. Formally, we study the parameterized complexity of the following problem:

<table>
<thead>
<tr>
<th><strong>$k$-Edge Change $(1, 2)$-STSP</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input:</strong> A set $V$ of $n$ cities, a symmetric distance matrix over all city pairs with distances 1 and 2, a tour $C'$, and an integer $k \in \mathbb{N}$.</td>
</tr>
<tr>
<td><strong>Question:</strong> Is there a tour $C''$ with length less than that of $C'$, and with distance at most $k$ from $C'$?</td>
</tr>
</tbody>
</table>
Theorem 5. Given an instance of \((1, 2)\)-STSP and a Hamiltonian cycle \(C\) of it, it is \(\mathcal{W}[1]\)-hard to decide whether there is a Hamilton cycle \(C'\) such that the cost of \(C'\) is strictly less than the cost of \(C\) and the distance between \(C\) and \(C'\) is at most \(k\).

The proof is by reduction from \(t\)-CLIQUE: Given an undirected unweighted graph \(H\) where we have to find a clique of size \(t\), we construct an equivalent instance of \(k\)-Edge Change \((1, 2)\)-STSP on an undirected graph \(G\). Assume, without loss of generality, that \(t\) is odd, for otherwise we add a new vertex that is adjacent to all vertices of \(H\) and find a clique of size \(t + 1\).

9.1 The Switch Gadget

The graph \(G\) is built from several copies of the switch gadget shown in Fig. 6a. The gadget is connected to the rest of the graph at the vertices \(\alpha, \beta, \gamma, \delta\). It is easy to see that if a switch gadget is part of a larger graph and there is a Hamiltonian cycle in the larger graph, then this cycle traverses the gadget in one of the two ways presented in Fig. 6b and 6c. Either the cycle enters at \(\alpha\) and leaves at \(\beta\) in as 6b; or it enters at \(\gamma\) and leaves at \(\delta\) as in 6c, or vice-versa. We remark that the whole construction is undirected; the orientations only describe the order in which vertices are traversed by the tours. The parameters of the reduction will be set in such a way that a Hamiltonian cycle \(C'\) has to traverse a selection of switch gadget in way 6c in order to prevent that the total cost of the cycle will be too large. The gadget effectively acts as a switch: either it is used as an \(\alpha \rightarrow \beta\)-path, or as a \(\gamma \rightarrow \delta\)-path. In the first case, we say that the cycle uses the upper path of the gadget, in the second case we say that the cycle uses the lower path.

9.2 Vertex and Edge Segments

Let \(n\) be number of vertices in \(H\), and let \(m\) be its number of edges. For convenience, we identify the vertices with the integers \(\{1, \ldots, n\}\) and the edges with the integers \(\{1, \ldots, m\}\). We construct an undirected graph \(G\) that consists of \((n + m)t(t - 1)/2\) copies of the switch gadget and some additional vertices. The graph \(G\) represents an instance of \((1, 2)\)-STSP where all edges of \(G\) have cost 1 and all edges not in \(G\) have cost 2.

We have the following switch gadgets, where \([i]\) denotes the set \(\{1, 2, \ldots, i\}\) for an integer \(i\):

- **vertex gadget** \(V_{i,\{j_1,j_2\}}\) for each \(i \in [n]\) and \(\{j_1, j_2\} \in \binom{[i]}{2}\) and

- **edge gadget** \(S_{i,\{j_1,j_2\}}\) for each \(i \in [m]\) and \(\{j_1, j_2\} \in \binom{[i]}{2}\).

We form segments from one or more gadgets.

The vertex segment \(V_{i,j}\), for \(i \in [n]\) and \(j \in [t]\), consists of the entrance vertex \(a_{i,j}\), the exit vertex \(b_{i,j}\), and \((t - 1)/2\) gadgets \(V_{i,\{j,j'\}}\).

Consider the pairs \(\{j_1, j_2\} \in \binom{[i]}{2}\) and let \(P_1, \ldots, P_{(t-1)/2}\) be an ordering of these pairs such that the second element of a pair is the same as the first element of the next pair, that is, \(P_\ell = \{p_\ell, p_{\ell+1}\}\) for every \(\ell \in [t(t-1)/2]\). Such an ordering exists: consider for instance the
sequence of edges of an Eulerian tour in $K_t$, the complete graph of order $t$. (Since $t$ is odd, $K_t$ is an Eulerian graph.)

These pairs define a partition of all $t(t - 1)/2$ pairs into $t$ classes $R_1, R_2, \ldots, R_t$ of $(t - 1)/2$ vertices each as follows. Let $j \in [t]$ be some index. Then $R_j = \{\{p_j', p_{j'+1}'\} | p_j' = j\}$.

The vertex segment $V_{i,j}$ contains all vertex gadgets of pairs in $R_j$; see Fig. 7. To simplify the notation, let $W_1, \ldots, W_{(t-1)/2}$ be an arbitrary ordering of these gadgets. For every $\ell \in \{1, \ldots, (t - 1)/2 - 1\}$, there is an edge from vertex $\beta$ of $W_{\ell}$ to vertex $\alpha$ of $W_{\ell+1}$. Furthermore, there is an edge from $a_{i,j}$ to vertex $\alpha$ of gadget $W_1$, and there is an edge from vertex $\beta$ of gadget $W_{t-1}$ to $b_{i,j}$. Finally, there is a bypass edge from $a_{i,j}$ to $b_{i,j}$.

The edge segment $E_{i,j_1,j_2}$, for $i \in \{1, \ldots, m\}$ and $j_1, j_2 \in \{1, \ldots, t\}$ with $j_1 \neq j_2$ contains an entrance vertex $z_{i,j_1,j_2}$, an exit vertex $q_{i,j_1,j_2}$, and the gadget $S_{i,j_1,j_2}$, see Fig. 8. There is an edge from $z_{i,j_1,j_2}$ to vertex $\alpha$ of $S_{i,j_1,j_2}$ and an edge from vertex $\beta$ of $S_{i,j_1,j_2}$ to $q_{i,j_1,j_2}$. Additionally, there is a bypass edge from $z_{i,j_1,j_2}$ to $q_{i,j_1,j_2}$.

Consider an arbitrary ordering of the $nt + mt(t - 1)/2$ segments defined above. Add an edge from the exit of each segment to the entrance of the next segment. Note that in the $(1,2)$-STSP instance defined by $G$, there is an edge of cost 2 that goes from the exit of the last segment (denote it by $v_{\text{last}}$) to the entrance of the first segment (denote it by $v_{\text{first}}$). There will be some more edges in the graph $G$, but before completing the description of $G$, we first define the Hamiltonian cycle $C$. The cycle starts at $v_{\text{first}}$, goes through the upper path of the gadget(s) in the first segment, leaves the segment at the exit, enters the second segment at its entrance, etc. The cycle does not use the bypass edges, thus it visits every vertex of every gadget. Finally, when $C$ reaches the exit of the last segment ($v_{\text{last}}$), it goes back to the entrance of the first segment ($v_{\text{first}}$) using the edge of cost 2. The cycle traverses exactly 12 vertices in each switch gadget and additionally two vertices of each segment. Therefore, the total number of vertices in $G$ is $nt(12(t-1)/2 + 2) + (12 + 2)mt(t - 1)/2 = 6nt^2 - 4nt + 7mt(t - 1)$, and the total cost of $C$ is $6nt^2 - 4nt + 7mt(t - 1) + 1$.
9.3 Encoding the Graph

As discussed above, we will ensure that the cycle $C'$ traverses every gadget as either 6b or 6c in Fig. 6. In the latter case, we say that the gadget is active. We will show that the active gadgets describe a $t$-clique of graph $H$. If gadget $S_i,\{j_1,j_2\}$ in edge segment $E_r,\{j_1,j_2\}$ is active, then this means that the edge $i$ is the edge connecting the $j_1$-th and $j_2$-th vertices of the clique.

If gadget $V_i,\{j_1,j_2\}$ is active, then this means that vertex $i$ is incident to the edge between the $j_1$-th and $j_2$-th vertex of the clique. We connect the vertices of gadgets in a way that enforces that the active gadgets describe a clique.

For every edge gadget, we add edges as follows (see Fig. 9). If vertices $i, i'$ are endpoints of edge $r$, then there is an edge from vertex $\delta$ of $V_i,\{p_t,p_{t+1}\}$ to vertex $\gamma$ of $E_r,\{p_t,p_{t+1}\}$, and there is an edge from vertex $\delta$ of $E_{r,\{p_t,p_{t+1}\}}$ to vertex $\gamma$ of $V_{r,\{p_{t+1},p_{t+2}\}}$. Furthermore, for every $i \in \{1,\ldots,n\}$, there is an edge from $v_{last}$ to vertex $\gamma$ of $V_i,\{p_1,p_2\}$, and for each edge $r$ there is an edge from vertex $\delta$ of $S_{r,\{p_{t(t-1)/2-1},p_{t(t-1)}\}}$ to $v_{first}$. This completes the description of the graph $G$.

9.4 $k$-Edge Change (1,2)-STSP ⇒ $t$-Clique

We claim that if there is a Hamiltonian cycle $C'$ in $G$ with weight strictly less than $6nt^2 - 4nt + 7mt(t-1)+1$ that is at distance at most $k = 9t(t-1)+2(t+1)$ from $C$, then there is a $t$-clique in $H$. The cycle $C'$ has as many edges as vertices, hence the only way the total weight is at most $6nt^2 - 4nt + 7mt(t-1)$ is if $C'$ does not use the edge of weight 2 that goes from $v_{last}$ to $v_{first}$. Furthermore, every gadget has to be traversed either as 6b or 6c of Fig. 6.

Let us think about the cycle $C'$ as a path that starts from and returns to $v_{first}$. Similar to $C$, the cycle $C'$ has to go through the segments one by one. It is clear that if $C'$ enters a segment at its entrance, then it has to leave it via its exit as otherwise it has gadgets that cannot be collected entirely with edges of $G$. However, inside a segment, $C'$ can do two things: either it goes through the gadget(s) (similarly to $C$), or it skips the gadget(s) using the bypass edge. In the latter case, we say that the segment is active. If vertex segment $V_{i,j}$ is active, then we will take it as an indication that vertex $i$ should be the $j$-th vertex of the clique. If edge segment $E_{i,\{j_1,j_2\}}$ is active, then this will mean that the $j_1$-th and the $j_2$-th vertices of the clique are connected by edge $i$. By the time $C'$ reaches $v_{last}$, every gadget is completely traversed, or not visited at all. The cycle has to return to $v_{first}$ by visiting all the skipped gadgets. As a result, the only possibility to continue from $v_{last}$ without an edge of cost 2 is to go to the $\gamma$-vertex of some gadget $V_{i,\{p_1,p_2\}}$ of an active vertex segment $V_{i,p_1}$. The tour has to continue from $\delta$ of $V_{i,\{p_1,p_2\}}$ to $\gamma$ of some edge gadget $S_{r,\{p_1,p_2\}}$ where $r$ is an edge incident to $i$ and $E_{r,\{p_1,p_2\}}$ is an active edge segment.

We argue that unless the tour traverses exactly $t(t-1)$ switch gadgets from $\gamma$ to $\delta$, there
Theorem 6. There is no function $f(k) \cdot n^{O(1)}$ time algorithm for $k$-Edge Change (1, 2)-STSP with $n$ cities, unless the Exponential Time Hypothesis fails.
Proof. The proof in Theorem 5 takes an instance \((H, t)\) of \(t\)-CLIQUE, and turns it into an equivalent instance of \(k\)-EDGE CHANGE \((1, 2)\)-STSP with \(k = O(t^3)\). Therefore, an \(f(k) \cdot n^{o(\sqrt{k})}\) time algorithm for \(k\)-EDGE CHANGE \((1, 2)\)-STSP would be able to solve any instance of \(t\)-CLIQUE in time \(f'(t) \cdot n^{o(t)}\). As shown by Chen et al. \[8\], this would imply that \(n\)-variable 3-SAT can be solved in time \(2^{o(n)}\).

10 Discussion

Our main results in this paper are improved integrality upper bounds for undirected and directed variants of TSP with distances one and two that arise from showing structural properties of the subtour elimination LP support graph. For undirected fractionally Hamiltonian instances with subcubic support, we were able to show a tight integrality gap of 10/9. Our results extend recent work on this problem by Qian et al. \[18\] and open up new directions for improved approximation algorithms for TSP with distances one and two.

A number of interesting research questions arises.

- The analysis of our algorithms yields approximation factors that are slightly worse than the best known approximation factors. However, it could very well be that our algorithms provide better approximation guarantees but we are unable to prove this. It would thus be interesting to know whether stronger algorithms, or simply a better analysis the presented algorithms, are needed to improve the best known approximation factors due to Berman and Karpinski \[2\] respectively Bläser \[3\].

- Can we guarantee that there are consecutive 1-edges in the support of an optimal solution to the subtour elimination relaxation for \((1, 2)\)-STSP? This would directly imply that due to Lemma 2 for \((1, 2)\)-STSP the assumption of Corollary 1 always holds and thus we obtain integrality gap upper bounds without assuming the instance to be fractionally Hamiltonian.

- Is there a 10/9 approximation for \((1, 2)\)-STSP, and a 7/6 approximation for \((1, 2)\)-ATSP, obtainable in polynomial time?

- Can we implement our algorithm to have a small polynomial running time, say cubic in the input size?

- Does a \(k\)-edge change local search without computing a solution to SER still provide a good integrality gap upper bound? That is: is computing an optimal solution to SER essential for the algorithm or only for the analysis?
References


