On the Integrality Gap of the Subtour LP for the 1,2-TSP

Jiawei Qian[1⋆], Frans Schalekamp, David P. Williamson[1⋆⋆], and Anke van Zuylen[2]

[1] School of Operations Research and Information Engineering,
Cornell University, Ithaca, NY 14853, USA.
jq35@cornell.edu, dpw@cs.cornell.edu
[2] Max-Planck-Institut für Informatik, Department 1: Algorithms and Complexity,
Campus E1 4, Room 311c, 66123 Saarbrücken, Germany.
anke@mpi-inf.mpg.de

Abstract. In this paper, we study the integrality gap of the subtour LP relaxation for the traveling salesman problem in the special case when all edge costs are either 1 or 2. For the general case of symmetric costs that obey triangle inequality, a famous conjecture is that the integrality gap is 4/3. Little progress towards resolving this conjecture has been made in thirty years. We conjecture that when all edge costs $c_{ij} \in \{1, 2\}$, the integrality gap is 10/9. We show that this conjecture is true when the optimal subtour LP solution has a certain structure. Under a weaker assumption, which is an analog of a recent conjecture by Schalekamp, Williamson and van Zuylen, we show that the integrality gap is at most 7/6. When we do not make any assumptions on the structure of the optimal subtour LP solution, we can show that inegrality gap is at most $19/15 \approx 1.267 < 4/3$; this is the first bound on the integrality gap of the subtour LP strictly less than 4/3 known for an interesting special case of the TSP.

1 Introduction

The Traveling Salesman Problem (TSP) is one of the most well studied problems in combinatorial optimization. Given a set of cities $\{1, 2, \ldots, n\}$, and distances $c(i, j)$ for traveling from city i to j, the goal is to find a tour of minimum length that visits each city exactly once. An important special case of the TSP is the case when the distance forms a metric, i.e., $c(i, j) \leq c(i, k) + c(k, j)$ for all i, j, k, and all distances are symmetric, i.e., $c(i, j) = c(j, i)$ for all i, j. The symmetric TSP is known to be APX-hard, even if $c(i, j) \in \{1, 2\}$ for all i, j [15]; note that such instances trivially obey the triangle inequality.

⋆ Supported in part by NSF grant CCF-1115256.

⋆⋆ This work was carried out while the author was on sabbatical at TU Berlin. Supported in part by the Berlin Mathematical School, the Alexander von Humboldt Foundation, and NSF grant CCF-1115256.

The symmetric TSP can be approximated to within a factor of $\frac{3}{2}$ using an algorithm by Christofides [7] from 1976. The algorithm combines a minimum spanning tree with a matching on the odd-degree nodes to get an Eulerian graph that can be shortcut to a tour; the analysis shows that the minimum spanning tree and the matching cost no more than the optimal tour and half the optimal tour respectively. Better results are known for several special cases, but, surprisingly, no progress has been made on approximating the general symmetric TSP in more than thirty years. A natural direction for trying to obtain better approximation algorithms is to use linear programming. The following linear programming relaxation of the traveling salesman problem was used by Dantzig, Fulkerson, and Johnson [9] in 1954. For simplicity of notation, we let $G = (V, E)$ be a complete undirected graph on n nodes. In the LP relaxation, we have a variable $x(e)$ for all $e = (i, j)$ that denotes whether we travel directly between cities i and j on our tour. Let $c(e) = c(i, j)$, and let $\delta(S)$ denote the set of all edges with exactly one endpoint in $S \subseteq V$. Then the relaxation is

$$\begin{array}{lll}
 & \text{Min} \sum_{e \in E} c(e)x(e) & \\
(SUBT) \quad \text{subject to:} & \sum_{e \in \delta(i)} x(e) = 2, \qquad \forall i \in V, & (1) \\
 & \sum_{e \in \delta(S)} x(e) \geq 2, \qquad \forall S \subset V,\ 3 \leq |S| \leq |V| - 3, & (2) \\
 & 0 \leq x(e) \leq 1, \qquad \forall e \in E. & (3)
\end{array}$$

The first set of constraints (1) are called the *degree constraints*. The second set of constraints (2) are sometimes called *subtour elimination constraints* or sometimes just *subtour constraints*, since they prevent solutions in which there is a subtour of just the nodes in S. As a result, the linear program is sometimes called the *subtour LP*. It has been shown by Wolsey [19] (and later Shmoys and Williamson [17]) that Christofides' algorithm finds a tour of length at most $\frac{3}{2}$ times the optimal value of the subtour LP; these proofs show that the minimum spanning tree and the matching on odd-degree nodes can be bounded above by the value of the subtour LP, and half the value of the subtour LP, respectively. This implies that the integrality gap, the worst case ratio of the length of an optimal tour divided by the optimal value of the LP, is at most $\frac{3}{2}$. However, no examples are known that show that the integrality gap can be as large as $\frac{3}{2}$; in fact, no examples are known for which the integrality gap is greater than $\frac{4}{3}$. A well known conjecture states that the integrality gap is indeed $\frac{4}{3}$; see (for example) Goemans [10].

Recently, progress has been made in several directions, both in improving the best approximation guarantee and in determining the exact integrality gap of the subtour LP for certain special cases of the symmetric TSP. In the *graph*-TSP, the costs $c(i, j)$ are equal to the shortest path distance in an underlying unweighted graph. Oveis Gharan, Saberi, and Singh [14] show that the graph-TSP can be approximated to within $\frac{3}{2} - \epsilon$ for a small constant $\epsilon > 0$. Boyd, Sitters, van der Ster and Stougie [6], and Aggarwal, Garg and Gupta [1] independently, give a $\frac{4}{3}$-approximation algorithm if the underlying graph is cubic. Mömke and Svensson [12] improve these results by giving a 1.461-approximation for the graph-TSP and an $\frac{4}{3}$-approximation algorithm if the underlying graph is subcubic. Their results also imply upper bounds on the integrality gap of 1.461 and $\frac{4}{3}$ in these cases. Mucha [13] improves the analysis of this algorithm for graph-TSP to a bound of $\frac{13}{9}$.

In Schalekamp, Williamson and van Zuylen [16], three of the authors of this paper resolve a related conjecture. A 2-matching of a graph is a set of edges such that no edge appears twice and each node has degree two, i.e., it is an integer solution to the LP ($SUBT$) with only constraints (1) and (3). Note that a minimum-cost 2-matching thus provides a lower bound on the length of the optimal TSP tour. A minimum-cost 2-matching can be found in polynomial time using a reduction to a certain minimum-cost matching problem. Boyd and Carr [5] conjecture that the worst case ratio of the cost of a minimum-cost 2-matching and the optimal value of the subtour LP is at most $\frac{10}{9}$. This conjecture was proved to be true by Schalekamp et al. and examples are known that show this result is tight.

Unlike the techniques used to obtain better results for the graph-TSP, the techniques of Schalekamp et al. work on general weighted instances that are symmetric and obey the triangle inequality. However, their results only apply to 2-matchings and it is not clear how to enforce global connectivity on the solution obtained by their method. A potential direction for progress on resolving the integrality gap for the subtour LP is a conjecture by Schalekamp et al. that the worst-case integrality gap is attained for instances for which the optimal subtour LP solution is a basic solution to the linear program obtained by dropping the subtour elimination constraints.

In this paper, we turn our attention to the 1,2-TSP, where $c(i,j) \in \{1,2\}$ for all i,j. Papadimitriou and Yannakakis [15] show how to approximate 1,2-TSP within a factor of $\frac{11}{9}$ starting with a minimum-cost 2-matching. In addition, they show a ratio of $\frac{7}{6}$ with respect to the the minimum-cost 2-matching that has no cycles of length 3. Bläser and Ram [4] improve this ratio and the best known approximation factor of $\frac{8}{7}$ is given by Berman and Karpinski [3].

We do not know a tight bound on the integrality gap of the subtour LP even in the case of the 1,2-TSP. As an upper bound, we appear to know only that the gap is at most $\frac{3}{2}$ via Wolsey's result. There is an easy 9 city example showing that the gap must be at least $\frac{10}{9}$; see Figure 1. This example has been extended to a class of instances on $9k$ nodes for any positive integer k by Williamson [18]. The contribution of this paper is to begin a study of the integrality gap of the 1,2-TSP, and to improve our state of knowledge for the subtour LP in this case. We are able to give the first bound that is strictly less than $\frac{4}{3}$ for these instances. This is the first bound on the integrality gap for the subtour LP with value less than $\frac{4}{3}$ for a natural class of TSP instances. Under an analog of a conjecture of Schalekamp et al. [16], we show that the integrality gap is at most $\frac{7}{6}$, and with

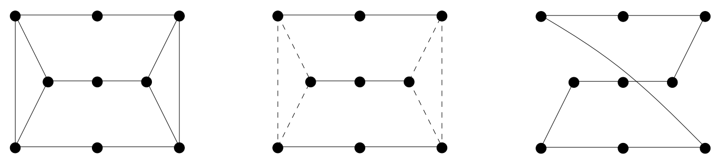

Fig. 1: Illustration of the worst example known for the integrality gap for the 1,2-TSP. The figure on the left shows all edges of cost 1. The figure in the center gives the subtour LP solution, in which the dotted edges have value $\frac{1}{2}$, and the solid edges have value 1; this is also an optimal fractional 2-matching. The figure on the left gives the optimal tour and the optimal 2-matching.

an additional assumption on the structure of the solution, we can improve this bound to $\frac{10}{9}$. We describe these results in more detail below.

We start by giving a bound on the subtour LP in the general case of 1,2-TSP. All the known approximation algorithms since the initial work of Papadimitriou and Yannakakis [15] on the problem start by computing a minimum-cost 2-matching. However, the example of Figure 1 shows that an optimal 2-matching can be as much as $\frac{10}{9}$ times the value of the subtour LP for the 1,2-TSP, so we cannot directly replace the bound on the optimal solution in these approximation algorithms with the subtour LP in the same way that Wolsey did with Christofides' algorithm in the general case. Using the result of Schalekamp, Williamson, and van Zuylen [16] and some additional work, we are able to show that an algorithm of Papadimitriou and Yannakakis [15] obtains a bound on the subtour LP for the 1,2-TSP of $\frac{7}{9} \cdot \frac{10}{9} + \frac{4}{9} = \frac{106}{81} \approx 1.3086$.

Next, we show stronger results in some cases. A fractional 2-matching is a basic optimal solution to the LP ($SUBT$) with only constraints (1) and (3). Schalekamp et al. [16] have conjectured that the worst-case integrality gap for the subtour LP is obtained when the optimal solution to the subtour LP is an extreme point of the fractional 2-matching polytope. We show that if this is the case for 1,2-TSP then we can find a tour of cost at most $\frac{7}{6}$ the cost of the fractional 2-matching, implying that the integrality gap is at most $\frac{7}{6}$ in these cases. We then show that if this optimal solution to the fractional 2-matching problem has a certain structure, then we can find a tour of cost at most $\frac{10}{9}$ times the cost of the fractional 2-matching, implying that the integrality gap is at most $\frac{10}{9}$ in these cases. Figure 1 shows that this result is tight. The results above all lead to polynomial-time algorithms, though we do not state the exact running times.

Finally, we prove that there exists a worst-case integrality gap instance for which the optimal value of the subtour LP is less than $n+1$, where n denotes the number of nodes. For such instances, we show that our previous arguments imply that one can construct a tour of cost at most $\frac{19}{15}$ times the subtour LP value. We anticipate that substantially stronger bounds on the integrality gap can be shown. In particular, we conjecture that the integrality gap is in fact exactly $\frac{10}{9}$.

We perform computational experiments that show that this conjecture is true for $n \leq 12$.

The remainder of this paper is structured as follows. Section 2 contains preliminaries and a first general bound on the integrality gap for the 1,2-TSP. We show how to obtain stronger bounds if the optimal subtour LP solution is a fractional 2-matching in Section 3. In Section 4, we combine the arguments from the previous sections and show that the integrality gap without any assumptions on the structure of the subtour LP solution is at most $\frac{19}{15}$. We describe our computational experiments in Section 5. Finally, we close with a conjecture on the integrality gap of the subtour LP for the 1,2-TSP in Section 6. Some proofs are omitted due to space reasons and can be found in the full version. of the paper.

2 Preliminaries and a first bound on the integrality gap

We will work extensively with 2-matchings and fractional 2-matchings; that is, extreme points x of the LP ($SUBT$) with only constraints (1) and (3), where in the first case the solutions are required to be integer. For convenience we will abbreviate "fractional 2-matching" by F2M and "2-matching" by 2M. F2Ms have the following well-known structure (attributed to Balinski [2]). Each connected component of the support graph (that is, the edges e for which $x(e) > 0$) is either a cycle on at least three nodes with $x(e) = 1$ for all edges e in the cycle, or consists of odd-sized cycles with $x(e) = \frac{1}{2}$ for all edges e in the cycle connected by paths of edges e with $x(e) = 1$ for each edge e in the path (the center figure in Figure 1 is an example). We call the former components *integer components* and the latter *fractional components*. In a fractional component, we call a path of edges e with $x(e) = 1$ a *1-path*. The edges e with $x(e) = \frac{1}{2}$ in cycles are called *cycle edges*. An F2M with a single component is called *connected*, and we call a component *2-connected* if the sum of the x-values on the edges crossing any cut is at least 2. We let n denote the number of nodes in an instance.

As mentioned in the introduction, Schalekamp, Williamson, and van Zuylen [16] have recently shown the following.

Theorem 1 (Schalekamp et al. [16]). *If edge costs obey the triangle inequality, then the cost of an optimal 2-matching is at most $\frac{10}{9}$ times the value of the subtour LP.*

It is not hard to show that this immediately implies an upper bound of $\frac{4}{3} \times \frac{10}{9}$ on the integrality gap of the subtour LP for the 1,2-TSP: we can just compute a minimum cost 2-matching, and remove the most expensive edge from each cycle, which gives a collection of node disjoint paths, which can be combined into a tour of cost at most $\frac{4}{3} \times \frac{10}{9}$ times the value of the subtour LP. The following theorem states that applying an algorithm by Papadimitriou and Yannakakis [15] to this 2-matching will produce a tour of cost at most $\frac{106}{81}$ times the value of the subtour LP.

Theorem 2. *The integrality gap of the subtour LP is at most $\frac{106}{81}$ for 1,2-TSP.*

Proof. Papadimitriou and Yannakakis [15] observe that we can assume without loss of generality that the optimal 2M solution consists of a number of cycles with only edges of cost 1 ("pure" cycles) and at most one cycle which has one or more edges of cost 2 (the "non-pure" cycle). Moreover, if i is a node in the non-pure cycle which is incident on an edge of cost 2 in the cycle, then there can be no edge of cost 1 connecting i to a node in a pure cycle (since otherwise, we can merge the non-pure cycle with a pure cycle without increasing the cost).

The Papadimitriou-Yannakakis algorithm solves the following bipartite matching problem: On one side we have a node for every pure cycle, and on the other side, we have a node for every node in the instance. There is an edge from pure cycle C to node i, if $i \notin C$ and there is an edge of cost 1 from i to some node in C. Let r be the number of pure cycles that are unmatched in the maximum cardinality bipartite matching. Papadimitriou and Yannakakis show how to "patch together" the matched cycles, and finally how to combine the resulting cycles into a tour of cost at most $\frac{7}{9}OPT(2M) + \frac{4}{9}n + \frac{1}{3}r$, where $OPT(2M)$ is the cost of an optimal 2M. We claim that

$$OPT(SUBT) \geq n + r,$$

where $OPT(SUBT)$ denotes the cost of the optimal subtour LP solution. Note that, combined with the analysis of Papadimitriou and Yannakakis and Theorem 1, this implies that the cost of the tour is then at most $\frac{7}{9} \cdot \frac{10}{9}OPT(SUBT) + \frac{4}{9}OPT(SUBT)$.

To prove the claim, we note that for a bipartite matching instance, the size of the minimum cardinality vertex cover is equal to the size of the maximum matching. We use this fact to construct a feasible dual solution to the subtour LP that has value $n+r$. Let $\mathcal{C}_M, V_M$ be the pure cycles and nodes (in the original graph), for which the corresponding nodes in the bipartite matching instance are in the minimum cardinality vertex cover. The dual of the subtour LP ($SUBT$) is

$$\begin{array}{lll}
 & \text{Max } 2\sum_{S\subset V} y(S) + 2\sum_{i\in V} y(i) - \sum_{e\in E} z(e) & \\
(D) \text{ subject to:} & \sum_{S\subset V: e\in\delta(S)} y(S) + y(i) + y(j) - z(e) \leq c(e), & \forall e=(i,j), \\
 & y(S) \geq 0, & \forall S\subset V,\ 3\leq |S| \leq n-3, \\
 & z(e) \geq 0, & \forall e \in E.
\end{array}$$

We set $z(e) = 0$ for each $e \in E$, and we set $y(i) = \frac{1}{2}$ for each $i \in V \backslash V_M$. For a pure cycle on a set of nodes C, we set $y(C) = \frac{1}{2}$, if the cycle is not in $\mathcal{C}_M$. Note that the dual objective for this solution is exactly $n+r$. It remains to show that the dual constructed is feasible. The details are deferred to the full version. □

We note that the bound obtained on the integrality gap seems rather weak, as the best known lower bound on the integrality gap is only $\frac{10}{9}$. Schalekamp, Williamson, and van Zuylen [16] have conjectured that the integrality gap (or worst-case ratio) of the subtour LP occurs when the solution to the subtour LP is a fractional 2-matching. That is, the worst-case ratio for the subtour LP occurs for costs c such that an optimal subtour LP solution for c is the same as an optimal fractional 2-matching for c. Schalekamp et al. call such costs c *fractional 2-matching costs* for the subtour LP.

Conjecture 1 (Schalekamp et al. [16]) *The integrality gap for the subtour LP is attained for a fractional 2-matching cost for the subtour LP.*

In the next section, we show that we can obtain better bounds on the integrality gap of the subtour LP in the case that the optimal solution is a fractional 2-matching. In Section 4, we show how to combine the proof of Theorem 2 with the bounds in the next section, to obtain a better bound on the integrality gap.

3 Better bounds if the optimal solution is an F2M

If the optimal solution to the subtour LP is a fractional 2-matching, then a natural approach to obtaining a good tour is to start with the edges with cost 1 and x-value 1, and add as many edges of cost 1 and x-value $\frac{1}{2}$ as possible, without creating a cycle on a subset of the nodes. In other words, we will propose an algorithm that creates an acyclic spanning subgraph (V, T) where all nodes have degree one or two. We will call an acyclic spanning subgraph in which all nodes have degree 1 or 2 a partial tour. A partial tour can be extended to a tour by adding $d/2$ edges of cost 2, where d is the number of degree 1 nodes. The cost of the tour is $c(T) + d$, where $c(T) = \sum_{e \in T} c(e)$.

Lemma 1. *Let $G = (V, T)$ be a partial tour. Let A be a set of edges not in T that form an odd cycle or a path on $V' \subset V$, where the nodes in V' have degree one in T. We can find $A' \subset A$ such that $(V, T \cup A')$ is a partial tour, and*

- $|A'| \geq \frac{1}{3}|A|$ *if A is a cycle,*
- $|A'| \geq \frac{1}{3}(|A| - 1)$ *if A is a path,*

We will now use the lemma above to show a bound of $\frac{7}{6}$ on the integrality gap if the optimal subtour LP solution is a fractional 2-matching.

Theorem 3. *There exists a tour of cost at most $\frac{7}{6}$ times the cost of a connected F2M solution if $c(i, j) \in \{1, 2\}$ for all i, j.*

Proof. Let $P = \{e \in E : x(e) = 1\}$ (the edges in the 1-paths of x). We will start the algorithm with $T = P$. Let $R = \{e \in E : x(e) = \frac{1}{2} \text{ and } c(e) = 1\}$ (the edges of cost 1 in the cycles of x). Note that the connected components of the graph (V, R) consist of paths and odd cycles. The main idea is that we consider these components one by one, and use Lemma 1 to show that we can add a large number of the edges of each path and cycle, where we keep as an invariant that T satisfies the conditions of the lemma. Note that by Lemma 1, the number of edges added from each path or cycle A is at least $|A|/3$, except for the paths for which $|A| \equiv 1 \pmod 3$. Let $\mathcal{P}_1$ be this set of paths. We would like to claim that we add a third of the edges from *each* component, and we therefore preprocess the paths in $\mathcal{P}_1$, where we add one edge (either the first or last edge from each path in $\mathcal{P}_1$) to T if this is possible without creating a cycle in T, and if so, we remove this edge and its neighboring edge in R (if any) from R. After the preprocessing, we use Lemma 1 to process each of the components in (V, R).

We call a path A in $\mathcal{P}_1$ "eared" if the 1-paths that are incident on the first and last node of the path are such that they go between two neighboring nodes of A. It is not hard to show that we can add an edge from at least half of the paths in $\mathcal{P}_1$ that are not eared.

Note that for a path A in $\mathcal{P}_1$ that is not eared, and for which we had already added an edge to T in the preprocessing step, will have added a total of at least $1 + (|A| - 2 - 1)/3 = |A|/3$ edges. For a path in $\mathcal{P}_1$ for which we did not add an edge to T in the preprocessing stage, we have added at least $(|A| - 1)/3$ edges. Now, recall that a path A in $\mathcal{P}_1$ has $|A| \equiv 1 \pmod 3$, and that the number of edges added is an integer, so in the first case, the number of edges added is at least $|A|/3 + \frac{2}{3}$ and in the second case it is $|A|/3 - \frac{1}{3}$.

Let z be the number of eared paths in $\mathcal{P}_1$. Then, the total number of edges from R added can be lower bounded by $\frac{1}{3}|R| - \frac{1}{3}z$. We now give an upper bound on the number of nodes of degree one in T.

Let k be the number of cycle nodes in x, i.e. $k = \#\{i \in V : x(i,j) = \frac{1}{2}$ for some $j \in V\}$, and let p be the number of cycle edges of cost 2 in x, i.e. $p = \#\{e \in E : x(e) = \frac{1}{2}$ and $c(e) = 2\}$. Note that $p \geq z$, since R contains p paths on the k cycle nodes. Initially, when T contains only the edges in the 1-paths, all k nodes have degree one, and there are $k-p$ edges in R. We argued that we added at least $\frac{1}{3}|R| - \frac{1}{3}z = \frac{1}{3}k - \frac{1}{3}p - \frac{1}{3}z$ edges to T. Each edge reduces the number of nodes of degree one by two, and hence, the number of nodes of degree one at the end of the algorithm is at most $k - 2(\frac{1}{3}k - \frac{1}{3}p - \frac{1}{3}z) = \frac{1}{3}k + \frac{2}{3}p + \frac{2}{3}z$. Recall that $c(P)$ denotes the cost of the 1-paths, and the total cost of T at the end of the algorithm is equal to $c(P) + \frac{1}{3}k - \frac{1}{3}p - \frac{1}{3}z$. Since at most $\frac{1}{3}k + \frac{2}{3}p + \frac{2}{3}z$ nodes have degree one in T, we can extend T into a tour of cost at most $c(P) + \frac{2}{3}k + \frac{1}{3}p + \frac{1}{3}z$.

The cost of the solution x can be expressed as $c(P) + \frac{1}{2}k + \frac{1}{2}p$. Note that each 1-path connects two cycle nodes, hence $c(P) \geq \frac{1}{2}k$. Moreover, an eared path A is incident to one (if $|A| = 1$) or two (if $|A| > 1$) 1-paths of length two, since the support graph of x is simple. Therefore we can lower bound $c(P)$ by $\frac{1}{2}k + z$. Therefore, $\frac{7}{6}\left(c(P) + \frac{1}{2}k + \frac{1}{2}p\right) \geq c(P) + \frac{1}{12}k + \frac{1}{6}z + \frac{7}{12}k + \frac{7}{12}p \geq c(P) + \frac{2}{3}k + \frac{1}{3}z + \frac{1}{3}p$, where $p \geq z$ is used in the last inequality. □

We remark that the ratio of $\frac{7}{6}$ in Theorem 3 is achieved if every 1-path contains just one edge of cost 1, and all cycle edges have cost 1. However, in such a case, we could find another optimal F2M solution by removing the 1-path with endpoints in two different odd cycles of edges with $x(e) = \frac{1}{2}$, increasing the x-value on the four cycle edges incident on its endpoints to 1, and then alternating between setting the x-value to 0 and 1 around the cycles. Now, since the cycles are odd, the degree constraints are again satisfied. The objective value does not increase because we only change the x-value on edges of cost 1.

We will call an F2M solution *canonical*, if there exists no 1-path of cost 1, for which the endpoints are in different odd cycles, and the four cycle edges incident on the endpoints all have cost 1.

If the F2M solution is canonical *and* 2-connected, we can improve the analysis in Theorem 3 to show the following.

Theorem 4. *There exists a tour of cost at most $\frac{10}{9}$ times the cost of a 2-connected canonical F2M solution if $c(i,j) \in \{1,2\}$ for all i,j.*

Proof. We adapt the final paragraph of the proof of Theorem 3. As before, the cost of the tour is at most $c(P)+\frac{2}{3}k+\frac{1}{3}p+\frac{1}{3}z$. However, since the F2M solution x is 2-connected, $z=0$: if there is a 1-path connecting two nodes connected by a cycle edge $\{i,j\}$, then $\{i,j\}$ defines a cut in x with only two cycle edges crossing the cut, and hence x is not 2-connected.

The cost of the F2M solution is $c(P)+\frac{1}{2}k+\frac{1}{2}p$, and by the fact that x is canonical, we have that $c(P) \geq k-2p$. The proof is concluded by noting that then $\frac{10}{9}\left(c(P)+\frac{1}{2}k+\frac{1}{2}p\right) \geq c(P)+\frac{1}{9}k-\frac{2}{9}p+\frac{5}{9}k+\frac{5}{9}p=\frac{2}{3}k+\frac{1}{3}p+c(P)$. □

4 A better upper bound on the integrality gap

We now show how to use the results in the previous two sections, to obtain a general upper bound that is better than the bound given in Section 2.

Note that in order to bound the integrality gap of the solution obtained by the Papadimitriou-Yannakakis algorithm, we need to (i) bound the difference between the cost of the 2M solution and the subtour LP, and (ii) bound the difference between the 2M solution and the tour constructed from it by the Papadimitriou-Yannakakis algorithm.

As in the proof of Theorem 2, we call a cycle in a 2M solution a "pure" cycle if all its edges have cost 1, and a non-pure cycle otherwise. The idea behind this section is to show that the difference of (i) can be "charged" to the nodes in the non-pure cycles only, and that the difference of (ii) can be "charged" mainly to the nodes in the pure cycles.

We first state the following lemma, which formalizes the second statement.

Lemma 2. *If $OPT(SUBT) < n+1$, then the difference between the 2M solution and the tour constructed by the Papadimitriou-Yannakakis algorithm is at most $\frac{4}{15}n_{\text{pure}}+\frac{1}{10}n_{\text{non-pure}}$, where n_{pure} is the number of nodes in pure cycles in the 2M solution, and $n_{\text{non-pure}}$ is the number of nodes in non-pure cycles.*

Note that the proof of Theorem 2 and the assumption that $OPT(SUBT) < n+1$ imply that the Papadimitriou-Yannakakis algorithm finds a bipartite matching that matches all the pure cycles. A careful look at the analysis of Papadimitriou and Yannakakis [15] shows that their algorithm finds a tour which satisfies the lemma. We now show that we can indeed restrict our attention to instances with $OPT(SUBT) < n+1$, the requirement of Lemma 2.

Lemma 3. *The worst-case integrality gap is attained on an instance with subtour LP value less than $n+1$, where n is the number of nodes in the instance.*

The idea behind the proof is that, if $\lfloor OPT(SUBT) \rfloor = n + k$, then the total x-value on edges with cost 2 is at least k. We can add k nodes and for each new node, add edges of cost 1 to each existing node. We obtain a feasible subtour solution for the new instance with the same cost as the solution for the original instance, by rerouting one unit of flow from edges with cost 2 to go through each new node. Also, the cost of the optimal tour on the new instance is at least the cost of the optimal tour on the original instance, and hence, the integrality gap of the new instance is at least the integrality gap of the original instance.

Theorem 5. *The integrality gap of the subtour LP is at most* $\frac{19}{15}$ *for the 1,2-TSP.*

Proof. By Lemma 3, we can assume without loss of generality that $OPT(SUBT) < n + 1$. We first find an optimal F2M solution, and use Theorem 3 to convert each component of the F2M solution into a cycle on the nodes in the component. Note that the F2M problem is a relaxation of the subtour LP, and, since the F2M solution is half-integral, its objective value is either $n + \frac{1}{2}$ or n. Let n_{pure} and $n_{\text{non-pure}}$ be defined as in Lemma 2, where the cycles are those returned by applying Theorem 3 on each connected fractional component of the F2M solution. By Theorem 3, the total cost of the resulting 2-matching is at most $n_{\text{pure}} + \frac{7}{6} n_{\text{non-pure}} + \frac{7}{12}$ if the F2M solution has cost $n + \frac{1}{2}$ and at most $n_{\text{pure}} + \frac{7}{6} n_{\text{non-pure}}$ if the F2M solution has cost n. Combining this with Lemma 2, there exists a tour of cost at most $n + \frac{4}{15} n_{\text{pure}} + (\frac{1}{6} + \frac{1}{10}) n_{\text{non-pure}} + \frac{7}{12}$, if $OPT(SUBT) \geq n + \frac{1}{2}$, and at most $n + \frac{4}{15} n_{\text{pure}} + (\frac{1}{6} + \frac{1}{10}) n_{\text{non-pure}}$, if $OPT(SUBT) \geq n$. In either case, the cost of the tour is at most $\frac{19}{15} OPT(SUBT)$. □

5 Computational results

In the case of the 1,2-TSP, for a fixed n we can generate all instances as follows. For each value of n, we first generate all nonisomorphic graphs on n nodes using the software package NAUTY [11]. We let the cost of edges be one for all edges in G and let the cost of all other edges be two. Then each of the generated graph G gives us an instance of 1,2-TSP problem with n nodes, and this covers all instances of the 1,2-TSP for size n up to isomorphism.

In fact, we can do slightly better by only generating biconnected graphs. We say that a graph $G = (V, E)$ is *biconnected* if it is connected and there is no vertex $v \in V$ such that removing v disconnects the graph; such a vertex v is a *cut vertex*. It is easy to see that the subtour LP value is at least $n+1$ if G is not biconnected, hence, by Lemma 3 it suffices to consider biconnected graphs.

For each instance of size n, we solve the subtour LP and the corresponding integer program using CPLEX 12.1 [8] and a Macintosh laptop computer with dual core 2GHz processor and 1GB of memory. It is known that the integrality gap is 1 for $n \leq 5$, so we only consider problems of size $n \geq 6$. The results are summarized in Table 1. For $n = 11$, the number of nonisomorphic biconnected graphs is nearly a billion and thus too large to consider, so we turn to another approach. For $n = 11$ and $n = 12$, we use the fact that we know a lower bound

n	6	7	8	9	10	11	12
Subtour IP/LP ratio	8/7.5	8/7.5	9/8.5	10/9	11/10	12/11	13/12
# graphs	56	468	7,123	194,066	9,743,542	900,969,091	–

Table 1: The subtour LP integrality gap for 1,2-TSP for $6 \le n \le 12$, along with the number of nonisomorphic biconnected graphs for $6 \le n \le 11$.

on the integrality gap of $\alpha_n = \frac{n+1}{n}$, namely for the instances we obtain by adding two or three additional nodes to one of the 1-paths in the example in Figure 1. We then check whether this is the worst integrality gap for each vertex of subtour LP. A list of non-isomorphic vertices of the subtour LP is available for $n = 6$ to 12 at Sylvia Boyd's website `http://www.site.uottawa.ca/~sylvia/subtourvertices`. In order to check whether the lower bound on the integrality gap is tight, we solve the following integer programming problem for each vertex x of the polytope for $n = 11$ and $n = 12$, where now the costs $c(e)$ are the decision variables, and x is fixed:

$$\max\{z - \alpha_n \sum_{e \in E} c(e)x(e) : \sum_{e \in T} c(e) \ge z \ \forall \text{ tours } T; c(e) \in \{1,2\} \ \forall e \in E.\}$$

Note that α_n is the lower bound on the integrality gap for instances of n nodes. If the objective is nonpositive for all of the vertices of the subtour LP, then we know that α_n is the integrality gap for a particular value of n.

Since the number of non-isomorphic tours of n nodes is $(n-1)!/2$, the number of constraints is too large for CPLEX for $n = 11$ or 12. We overcome this difficulty by first solving the problem with only tours that have at least $n-1$ edges in the support graph of the vertex x, and repeatedly adding additional violated tours. We find that the worst case integrality gap for $n = 11$ is $\frac{12}{11}$ and for $n = 12$ is $\frac{13}{12}$.

6 Conjectures and conclusions

As stated in the introduction, we conjecture the following.

Conjecture 2 *The integrality gap of the subtour LP for the 1,2-TSP is $\frac{10}{9}$.*

Schalekamp, Williamson, and van Zuylen [16] have conjectured that the integrality gap (or worst-case ratio) of the subtour LP occurs when the solution to the subtour LP is a fractional 2-matching. We have shown in Theorem 3 that if an analogous conjecture is true for 1,2-TSP, then the integrality gap for 1,2-TSP is at most $\frac{7}{6}$; it would be nice to show that if the analogous conjecture is true for 1,2-TSP then the integrality gap is at most $\frac{10}{9}$.

Acknowledgements

We thank Sylvia Boyd for useful and encouraging discussions.

References

1. Aggarwal, N., Garg, N., Gupta, S.: A 4/3-approximation for TSP on cubic 3-edge-connected graphs (2011), available at `http://arxiv.org/abs/1101.5586`
2. Balinski, M.L.: Integer programming: Methods, uses, computation. Management Science 12, 253–313 (1965)
3. Berman, P., Karpinski, M.: 8/7-approximation algorithm for (1,2)-TSP. In: Proceedings of the 17th ACM-SIAM Symposium on Discrete Algorithms. pp. 641–648 (2006)
4. Bläser, M., Shankar Ram, L.: An improved approximation algorithm for TSP with distances one and two. In: Liskiewicz, M., Reischuk, R. (eds.) Fundamentals of Computation Theory, 15th International Symposium, FCT 2005, Lecture Notes in Computer Science, vol. 3623, pp. 504–515. Springer (2005)
5. Boyd, S., Carr, R.: Finding low cost TSP and 2-matching solutions using certain half-integer subtour vertices. Discrete Optimization 8(4), 525–539 (2011), prior version available at `http://www.site.uottawa.ca/~sylvia/recentpapers/halftri.pdf`. Accessed June 27, 2011
6. Boyd, S., Sitters, R., van der Ster, S., Stougie, L.: TSP on cubic and subcubic graphs. In: Günlük, O., Woeginger, G.J. (eds.) Integer Programming and Combinatorial Optimization, 15th International Conference, IPCO 2011, pp. 65–77. No. 6655 in Lecture Notes in Computer Science, Springer, Berlin, Germany (2011)
7. Christofides, N.: Worst case analysis of a new heuristic for the traveling salesman problem. Report 388, Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, PA (1976)
8. IBM ILOG CPLEX 12.1 (2009)
9. Dantzig, G., Fulkerson, R., Johnson, S.: Solution of a large-scale traveling-salesman problem. Operations Research 2, 393–410 (1954)
10. Goemans, M.X.: Worst-case comparison of valid inequalities for the TSP. Mathematical Programming 69, 335–349 (1995)
11. McKay, B.D.: Practical graph isomorphism. Congressus Numerantium 30, 45–97 (1981)
12. Mömke, T., Svensson, O.: Approximating graphic TSP by matchings. In: Proceedings of the 52th Annual Symposium on Foundations of Computer Science (2011)
13. Mucha, M.: 13/9-approximation for graphic TSP. In: 29th International Symposium on Theoretical Aspects of Computer Science (2012)
14. Oveis Gharan, S., Saberi, A., Singh, M.: A randomized rounding approach to the traveling salesman problem. In: Proceedings of the 52th Annual Symposium on Foundations of Computer Science (2011)
15. Papadimitriou, C.H., Yannakakis, M.: The traveling salesman problem with distances one and two. Mathematics of Operations Research 18, 1–11 (1993)
16. Schalekamp, F., Williamson, D.P., van Zuylen, A.: A proof of the Boyd-Carr conjecture. In: Proceedings of the 23nd ACM-SIAM Symposium on Discrete Algorithms (2012)
17. Shmoys, D.B., Williamson, D.P.: Analyzing the Held-Karp TSP bound: A monotonicity property with application. Information Processing Letters 35, 281–285 (1990)
18. Williamson, D.P.: Analysis of the Held-Karp Heuristic for the Traveling Salesman Problem. Master's thesis, MIT, Cambridge, MA (Jun 1990), also appears as Tech Report MIT/LCS/TR-479
19. Wolsey, L.A.: Heuristic analysis, linear programming and branch and bound. Mathematical Programming Study 13, 121–134 (1980)