An Improved Analysis of the Mömke-Svensson Algorithm for Graph-TSP on Subquartic Graphs*

Alantha Newman

CNRS-Université Grenoble Alpes and G-SCOP, F-38000 Grenoble, France
firstname.lastname@grenoble-inp.fr

Abstract. Recently, Mömke and Svensson presented a beautiful new approach for the traveling salesman problem on a graph metric (graph-TSP), which yielded a $\frac{4}{3}$-approximation guarantee on subcubic graphs as well as a substantial improvement over the $\frac{3}{2}$-approximation guarantee of Christofides' algorithm on general graphs. The crux of their approach is to compute an upper bound on the minimum cost of a circulation in a particular network, $C(G,T)$, where G is the input graph and T is a carefully chosen spanning tree. The cost of this circulation is directly related to the number of edges in a tour output by their algorithm. Mucha subsequently improved the analysis of the circulation cost, proving that Mömke and Svensson's algorithm for graph-TSP has an approximation ratio of at most $\frac{13}{9}$ on general graphs.

This analysis of the circulation is local, and vertices with degree four and five can contribute the most to its cost. Thus, hypothetically, there could exist a subquartic graph (a graph with degree at most four at each vertex) for which Mucha's analysis of the Mömke-Svensson algorithm is tight. In this paper, we show that this is not the case and that Mömke and Svensson's algorithm for graph-TSP has an approximation guarantee of at most $\frac{46}{33}$ on subquartic graphs. To prove this, we present a different method to upper bound the minimum cost of a circulation on the network $C(G,T)$. Our approximation guarantee actually holds for all graphs that have an optimal solution to a standard linear programming relaxation of graph-TSP with subquartic support.

1 Introduction

The *metric* traveling salesman problem (TSP) is one of the most well-known problems in the field of combinatorial optimization and approximation algorithms. Given a complete graph, $G = (V,E)$, with non-negative edge weights that satisfy the triangle inequality, the goal is to compute a minimum cost tour of G that visits each vertex exactly once. Christofides' algorithm, dating from almost four decades ago, yields a tour with cost no more than $3/2$ times that of an optimal tour [Chr76]. It remains a major open problem to improve upon this approximation factor.

* Supported in part by LabEx PERSYVAL-Lab (ANR–11-LABX-0025).

A. Schulz and D. Wagner (Eds.): ESA 2014, LNCS 8737, pp. 737–749, 2014.

Recently, there have been many exciting developments relating to *graph*-TSP. In this setting, we are given an unweighted graph $G = (V, E)$ and the goal is to find the shortest tour that visits each vertex *at least* once. This problem is equivalent to the special case of metric TSP where the shortest path distances in G define the metric. It is also equivalent to the problem of finding a connected, Eulerian multigraph in G with the minimum number of edges.

A promising approach to improving upon the factor of 3/2 for metric TSP is to round a linear programming relaxation known as the Held-Karp relaxation [HK70]. A lower bound of 4/3 on its integrality gap can be demonstrated using a family of graph-TSP instances. Even in this special case of metric TSP, graph-TSP had also long resisted significant progress before the recent spate of results.

1.1 Recent Progress on Graph-TSP

In 2005, Gamarnik et al. presented an algorithm for graph-TSP on cubic 3-edge connected graphs with an approximation factor of $3/2 - 5/389$ [GLS05], thus proving that Christofides' approximation factor of 3/2 is not optimal for this class of graphs. Their approach is based on finding a cycle cover for which they can upper bound the number of components. This general approach was also taken by Boyd et al. who combined it with polyhedral ideas to obtain approximation guarantees of 4/3 for cubic graphs and 7/5 for subcubic graphs, i.e. graphs with degree at most three at each vertex [BSvdSS11]. Shortly afterwards, Oveis Gharan et al. proved that a subtle modification of Christofides' algorithm has an approximation guarantee of $3/2 - \epsilon_0$ for graph-TSP on general graphs, where ϵ_0 is a fixed constant with value approximately 10^{-12} [GSS11].

Mömke and Svensson then presented a beautiful new approach for graph-TSP, which resulted in a substantial improvement over the 3/2-approximation guarantee of Christofides [MS11]. Their approach also lead to a surprisingly simple algorithm with an 4/3-approximation guarantee for subcubic graphs. We will discuss their algorithm in more detail in Section 1.2, since our paper is directly based on their approach. Ultimately, they were able to prove an approximation guarantee of 1.461 for graph-TSP. Mucha subsequently gave an improved analysis, thereby proving that Mömke and Svensson's algorithm for graph-TSP actually has an approximation ratio of at most 13/9 [Muc12]. Sebő and Vygen introduced an approach for graph-TSP based on ear decompositions and matroid intersection, which incorporated the techniques of Mömke and Svensson, and improved the approximation ratio to 7/5, where it currently stands [SV12]. For the special case of k-regular graphs, Vishnoi gave an algorithm for graph-TSP with an approximation guarantee that approaches 1 as k increases [Vis12].

Some of the new techniques for graph-TSP have also lead to progress on the metric s-t-path TSP, in which the goal is to find a path between two fixed vertices that visits every vertex at least once. Recent results improved upon the previously best-known bound of 5/3 for the s-t-path TSP due to Hoogeveen [Hoo91] in the special case of s-t-path graph-TSP [MS11, Muc12, SV12, Gao13] as well as in the case of general metrics [AKS12, Seb13].

1.2 Mömke-Svensson's Approach to Graph-TSP

Christofides' algorithm for graph-TSP finds a spanning tree of the graph and adds to it a J-join, where J is the set of vertices that have odd degree in the spanning tree. Since the spanning tree is connected, the resulting subgraph is clearly connected, and since the J-join corrects the parity of the spanning tree, the resulting subgraph is Eulerian. In contrast, the recent approach of Mömke and Svensson is based on removing an odd-join of the graph, which yields a possibly disconnected Eulerian subgraph. Thus, to maintain connectivity, one must double, rather than remove, some of the edges in the odd-join. The key step in proving the approximation guarantee of the algorithm is to show that many edges will actually be removed and relatively few edges will be doubled, resulting in a connected, Eulerian subgraph with few edges. Using techniques of Naddef and Pulleyblank [NP81], Mömke and Svensson show how to sample an odd-join of size $|E|/3$, where E is the subset of edges in the support of the linear programming relaxation for graph-TSP (see section 2.1). The number of edges that are doubled to guarantee connectivity is directly related to the minimum cost circulation of particular network, referred to as $C(G, T)$, which Mömke and Svensson construct based on the input graph G, an optimal solution to a linear programming relaxation for graph-TSP, and a carefully chosen spanning tree T. Lemma 4.1 from [MS11] relates the size of the solution for their algorithm to the minimum cost circulation of this network.

Lemma 1. **[MS11]** *Given a 2-vertex connected graph G and a depth first search tree T of G, let C^* be a minimum cost circulation for $C(G, T)$ of cost $c(C^*)$. Then there is a spanning Eulerian multigraph in G with at most $\frac{4}{3}n + \frac{2}{3}c(C^*)$ edges.*

We defer a precise description of the circulation network $C(G, T)$ to Section 2, where we formulate it using different notation from that in [MS11]. For the moment, we emphasize that if one can prove a better upper bound on the value of $c(C^*)$, then this directly implies improved upper bounds on the number of edges in a tour output by Mömke and Svensson's algorithm.

1.3 Our Contribution

We consider the graph-TSP problem for *subquartic* graphs, i.e. graphs in which each vertex has degree at most four. As pointed out in Lemma 2.1 of [MS11], we can assume that these graphs are 2-vertex connected. The best-known approximation guarantee for these graphs is inherited from the general case, even when the graph is 4-regular, and is therefore 7/5 due to Sebő and Vygen. For subquartic graphs, we give an improved upper bound on the minimum cost of a circulation for $C(G, T)$. Using Lemma 1, this leads to an improved approximation guarantee of 46/33 for graph-TSP on these graphs. Before we give an overview of our approach, we first explain our motivation for studying graph-TSP on this restricted class of graphs.

As mentioned in Section 1.1, graph-TSP is now known to be approximable to within 4/3 for subcubic graphs. So, on the one hand, trying to prove the same guarantee for subquartic graphs is arguably a natural next step. Additionally, it is a well-motivated problem to study the graph-TSP on sparse graphs, because the support of an optimal solution to the standard linear programming relaxation (reviewed in Section 2.1) has at most $2n-1$ edges (see Theorem 4.9 in [CFN85]). Thus, any graph that corresponds to the support of an optimal solution to the standard linear program has average degree less than four.

However, our actual motivation for studying graphs with degree at most four has more to do with understanding the Mömke-Svensson algorithm than with an abstract interest in subquartic graphs. The basic approach to computing an upper bound on the minimum cost circulation in $C(G,T)$ used in both [MS11] and [Muc12] is to specify flow values on the edges of $C(G,T)$ that are functions of an optimal solution to the linear programming relaxation for graph-TSP on the graph G. The cost of the circulation obtained using these values can be analyzed in a local, vertex by vertex manner. Mucha showed that vertices with degree four or five potentially increase the cost of the circulation the most [Muc12]. In fact, one could hypothetically construct a tight example for Mucha's analysis of the Mömke-Svensson algorithm on a graph where each vertex has degree at most four (or where each vertex has degree at most five). Thus it seems worthwhile to determine if the cost of the circulation can be improved on subquartic graphs. Our results actually hold for a slightly more general class of graphs than subquartic graphs: they hold for any graph that has an optimal solution to the standard linear programming relaxation of graph-TSP with subquartic support.

1.4 Organization

In Section 2.1, we discuss the standard linear programming relaxation for graph-TSP, and in Section 2.2, we present notation and definitions necessary for defining the circulation network $C(G,T)$. In Section 3, we show that if, for a subquartic graph, the optimal solution to the linear program has value equal to the number of vertices in G, then the network $C(G,T)$ has a circulation of cost zero, implying that the Mömke-Svensson algorithm has an approximation ratio of 4/3. This observation provides us with some intuition as to how one may attempt to design a better circulation for general subquartic graphs.

In Section 4, we describe two different methods to obtain feasible circulations. In Section 4.1, we detail the method used by Mömke-Svensson and Mucha, which becomes somewhat simpler in the special case of subquartic graphs. This method directly uses values from the optimal solution to the linear program to obtain flow values on edges in the network. In Section 4.2, we present a new method, which "rounds" the values from the optimal solution to the linear program. The latter circulation alone leads to an improved analysis over 13/9 for subquartic graphs, but it does not improve on the best-known guarantee of 7/5. However, as we finally show in Section 5, if we take the best of the two circulations, we can show that at least one of the circulations will lead to an approximation guarantee of at most 46/33.

We remark that our notation differs from that in [MS11] or [Muc12], even though we are using exactly the same circulation network and we use their approach for obtaining the feasible circulation described in Section 4.1. This different notation allows us to more easily analyze the tradeoff between the two circulations. Due to space constraints, this extended abstract is missing many proofs, which can be found in the full version.

2 Preliminaries: Notation and Definitions

Let $G = (V, E)$ be an undirected graph with maximum degree four, a property often referred to as *subquartic*. Throughout this paper, we make use of the following well-studied linear programming relaxation for graph-TSP.

2.1 Linear Program for Graph-TSP

For a graph $G = (V, E)$, the following linear program is a relaxation of graph-TSP. We refer to Section 2 of [MS11] for a discussion of its derivation and history.

$$\min \sum_{e \in E} y_e$$
$$\begin{aligned} y(\delta(S)) &\geq 2 \quad \text{for } \emptyset \neq S \subset V, \\ y &\geq 0. \end{aligned}$$

We denote this linear program by $LP(G)$ and we denote the value of an optimal solution for $LP(G)$ by $OPT_{LP}(G)$. Let n be the number of vertices in V. We can assume that G has the following two properties: (i) $|E| \leq 2n - 1$, and (ii) G is 2-vertex connected. Assumption (i) is based on the fact that any extreme point of $LP(G)$ has at most $2n-1$ edges (see Theorem 4.9 in [CFN85]), and restricting the graph to the edges in the support of an extreme point with optimal value does not increase the optimal value $OPT_{LP}(G)$. Assumption (ii) is based on Lemma 2.1 from [MS11]. We note that the two theorems we just cited may have to be applied multiple times to guarantee that G has the desired properties (i) and (ii).

Lemma 2. *Let $G = (V, E)$ be a 2-edge connected graph. Then there exists $x \in LP(G)$, $x \leq 1$ minimizing the sum of coordinates of a vector in $LP(G)$.*

From Lemma 2, we define $x \in \mathbb{R}^{|E|}$ to be an optimal solution for $LP(G)$ with the following properties: (i) the support of x contains at most $2n - 1$ edges, (ii) the support of x is 2-vertex connected, and (iii) $x \leq 1$. We will refer to the set of values $\{x_e\}$ for $e \in E$ as x-values. Let $\sum_{e \in E} x_e = OPT_{LP}(G) = (1 + \epsilon)n$ for some ϵ, where $0 \leq \epsilon \leq 1$. We will eventually make use of the following definitions.

Definition 1. *The* excess *x-value $\epsilon(v)$ at a vertex v is the amount by which the total value on the adjacent edges exceeds 2, i.e $\epsilon(v) = x(\delta(v)) - 2$.*

Definition 2. *A vertex $v \in V$ is called* heavy *if $x(\delta(v)) > 2$.*

The following fact will be useful in our analysis. If $OPT_{LP}(G) = (1+\epsilon)n$, then,

$$\sum_{v \in V} x(\delta(v)) = \sum_{v \in V} (2 + \epsilon(v)) \quad = \quad 2(1+\epsilon)n.$$

This implies, $\sum_{v \in V} \epsilon(v) = 2\epsilon n$.

2.2 Spanning Trees and Circulations

Let us recall some useful definitions from the approach of Mömke and Svensson [MS11] that we use throughout this paper.

Definition 3. *A* greedy DFS tree *is a spanning tree formed via a depth-first search of G. If there is a choice as to which edge to traverse next, the edge with the highest x-value is chosen.*

For a given graph G and an optimal solution to $LP(G)$, let T denote a greedy DFS tree. Let $B(T) \subset E$ denote the set of back edges with respect to the tree T. Each edge in T will be directed away from the root of the tree T and each edge in $B(T)$ will be directed towards the root of T. We use the notation (i,j) to denote an edge directed from i to j. Note that once we have fixed a tree T, all edges in E can be viewed as directed edges. When we wish to refer to an undirected edge in E, we use the notation $ij \in E$. With respect to the greedy DFS tree T, we have the following definitions.

Definition 4. *An* internal *node in T is a vertex that is neither the root of T nor a leaf in T. We use T_{int} to denote this subset of vertices.*

Definition 5. *An* expensive *vertex is a vertex in T_{int} with two incoming edges that belong to $B(T)$. We use T_{exp} to denote this subset of vertices.*

As we will see in Lemma 4, expensive vertices are the vertices that can contribute to the cost of $C(G,T)$. The root can also contribute a negligible value of either one or two to the cost of $C(G,T)$. For the sake of simplicity, we ignore the contribution of the root in most of our calculations.
Fact. The number of expensive vertices is bounded as follows: $|T_{exp}| \leq n/2$.

Definition 6. *A* branch *vertex in T is a vertex with at least two outgoing tree edges.*

Lemma 3. *A branch vertex is not expensive.*

Definition 7. *A* tree cut *is the partition of the vertices of the tree T induced when we remove an edge $(u,v) \in T$.*

For each edge $(i,j) \in B(T)$, let $b(i,j) \leq 1$ be a non-negative value.

Definition 8. *Consider a tree cut corresponding to edge $(u,v) \in T$ and remove all back edges $(w,u) \in B(T)$, where w belongs to the subtree of v in T. We say that the remaining back edges that cross this tree cut* cover *the cut. If the total b-value of the edges that cover the cut is at least 1, then we say that this tree cut is* satisfied by *b.*

We extend this definition to the vertices of T.

Definition 9. *A vertex v in T is* satisfied by b *if for each adjacent outgoing edge in T, the corresponding tree cut is satisfied by b. On the other hand, if there is at least one adjacent outgoing edge whose corresponding tree cut is not satisfied by b, then the vertex v is* unsatisfied by b.

Mömke and Svensson define a circulation network, $C(G,T)$ (see Section 4 of [MS11]), and use the cost of a feasible circulation to upper bound the length of a TSP tour in G. (See Lemma 1.)

Lemma 4. *Let $b : B(T) \rightarrow [0,1]$. If each internal vertex in T is satisfied by b, then there is a feasible circulation of $C(G,T)$ whose cost is upper bounded by the following function:*

$$\sum_{j \in T_{exp}} \max \left\{ 0, \left(\sum_{i:(i,j) \in B(T)} b(i,j) \right) - 1 \right\}. \tag{1}$$

Although finding b-values for the back edges that satisfy all the vertices is equivalent to finding a feasible circulation of $C(G,T)$, and we could have stuck to the notation presented in [MS11], we believe our notation results in a clearer presentation of our main theorems.

3 Subquartic Graphs: $OPT_{LP}(G) = n$

We now show that in the special case when $OPT_{LP}(G) = n$ (i.e. $\epsilon = 0$), there is a circulation with cost zero. Note that if $|E| = n$, then each edge in E must have x-value 1. Thus, G is a Hamiltonian cycle. If $|E| > n$, then we can show that we can find a greedy DFS tree T for G such that each edge $ij \in E$ with x-value $x_{ij} = 1$ (a "1-edge") belongs to T.

Lemma 5. *When $OPT_{LP}(G) = n$ and $|E| > n$, there is a greedy DFS tree T such that all 1-edges are in T.*

For the rest of Section 3, let T denote a greedy DFS tree in which all 1-edges are tree edges.

Lemma 6. *If $OPT_{LP}(G) = n$ and each back edge $(i,j) \in B(T)$ is assigned value $f(i,j) = 1/2$, then each vertex in T_{int} is satisfied by f.*

Lemma 7. *If $OPT_{LP}(G) = n$, setting $f(i,j) = 1/2$ for each edge $(i,j) \in B(T)$ yields a circulation with cost zero.*

Theorem 1. *If $OPT_{LP}(G) = n$ and G is a subquartic graph, then G has a TSP tour of length at most $4n/3$.*

4 Subquartic Graphs: General Case

In this section, we consider the general case of subquartic graphs. For a graph $G = (V, E)$, suppose $OPT_{LP}(G) = (1 + \epsilon)n$ for some $\epsilon > 0$. There is a fixed greedy DFS tree T as defined in Section 2.2. If we assign values to the edges in $B(T)$, then the only vertices that can add to the cost function are the expensive vertices, as we have defined them, since the maximum value allowed on an edge is one. Let $x(i, j) = x_{ij}$ for all back edges in $B(T)$. Recall that the $\{x_{ij}\}$ values are obtained from the solution to $LP(G)$ in Section 2.1.

Lemma 8. *A vertex v in T_{int} has at most one outgoing tree edge whose corresponding tree cut is not satisfied by x.*

Definition 10. *A vertex $v \in T_{int}$ that is satisfied by x is called* LP-satisfied.

Definition 11. *A vertex $v \in T_{int}$ that is not satisfied by x is called* LP-unsatisfied.

Lemma 9. *An expensive vertex is LP-satisfied.*

Lemma 10. *An LP-unsatisfied vertex is heavy.*

The reason we emphasize that an LP-unsatisfied vertex is heavy is that we can use the excess x-value of this vertex to increase an edge that covers the unsatisfied tree cut corresponding to one of its adjacent outgoing edges so that this tree cut becomes satisfied. We also wish to use the excess x-value of an expensive vertex to pay for some of its contribution to the cost function incurred by the back edges coming into the vertex. For each vertex v, we want to use the quantity $\epsilon(v)$ at most once. This will be guaranteed by the fact that LP-unsatisfied vertices and expensive vertices are disjoint sets.

4.1 The x-Circulation

In this section, we use the x-values to obtain an upper bound on the cost of a circulation, essentially following the arguments of Mömke and Svensson [MS11] and Mucha [Muc12]. We present the analysis here, since we refer to it in Section 5 when we analyze the cost of taking the best of two circulations. Also, the arguments can be somewhat simplified due to the subquartic structure of the graph, which is useful for our analysis.

For each back edge in $B(T)$, set $x(i, j) = x_{ij}$, where $x \in \mathbb{R}^{|E|}$ is an optimal solution to $LP(G)$. (For a vertex $j \notin T_{exp}$, we can actually set $x(i, j) = 1$, since there is at most one incoming back edge to vertex j, but this does not change the worst-case analysis.)

Definition 12. *For each vertex $j \in T_{exp}$, let $x_{min}(j) \le x_{max}(j)$ denote the x-values of the two incoming back edges to vertex j. Let $c_x(j) = x_{min}(j) + x_{max}(j) - 1 - \epsilon(j)$.*

Lemma 11. *For an expensive vertex $j \in T_{exp}$, the following holds:*

$$2 \cdot x_{max}(j) + x_{min}(j) \leq 2 + \epsilon(j).$$

We will show that there is a function $x' : B(T) \to [0, 1]$ such that each vertex in T is satisfied by x' and the cost of the circulation can be bounded by:

$$\sum_{j \in T_{exp}} \max \left\{ 0, \left(\sum_{i:(i,j) \in B(T)} x'(i,j) \right) - 1 \right\} \leq \sum_{j \in T_{exp}} \max\{0, c_x(j)\} + \sum_{j \in T} \epsilon(j) \quad (2)$$

Lemma 12. *The value $c_x(j)$ can be upper bounded as follows:*

$$c_x(j) \leq \frac{x_{min}(j)}{2} - \frac{\epsilon(j)}{2} \; \leq \; 1 - x_{min}(j).$$

Lemma 13. *For a vertex $j \in T_{exp}$, $c_x(j) \leq 1/3$.*

To make the circulation feasible, we need to increase the x-values of some of the back edges in $B(T)$ so that all of the LP-unsatisfied vertices become satisfied. By Lemma 10, these vertices are heavy. Thus, we will use $\epsilon(v)$ for an LP-unsatisfied vertex v to "pay" for increasing the x-value on an appropriate back edge. For ease of notation, we now set $x'(u,v) = x(u,v)$ for all $(u,v) \in B(T)$. We will update the $x'(u,v)$ values so that each LP-unsatisfied vertex is satisfied by x'.

Consider an LP-unsatisfied, non-branch vertex $j \in T$, and consider the tree cut corresponding to the single edge (j, t_2) outgoing from j in T. Let $S \subseteq B(T)$ denote the edges that cover this tree cut. Let $(i,j), (j,k) \in B(T)$ represent the adjacent back edges, and let $(t_1, j) \in T$ denote the incoming tree edge. Recall that in this tree cut, both edges (j, t_2) and (i,j) are removed and the remaining edges in $B(T)$ that cross this cut *cover* it. We have:

$$x(j,t_2) + x(j,k) + x(t_1,j) + x(i,j) = 2 + \epsilon(j).$$

Since,

$$x(S) + x(j,t_2) + x(i,j) \; \geq \; 2, \quad x(S) + x(j,k) + x(t_1,j) \; \geq \; 2,$$

it follows that

$$2 \cdot x(S) \; \geq \; 2 - \epsilon(j) \quad \Rightarrow \quad x(S) \; \geq \; 1 - \epsilon(j)/2.$$

Let $(u,v) \in S$ be an arbitrary edge in S. We will update the value of $x'(u,v)$ as follows:

$$x'(u,v) := \min\{1, \; x'(u,v) + \epsilon(j)/2\}.$$

We use this notation, because a back edge's value can be increased multiple times in the process of satisfying all LP-unsatisfied vertices.

If j is an LP-unsatisfied branch vertex, then it must have two outgoing edges in T (call them (j, t_2) and (j, t_3)) and one incoming back edge $(i,j) \in B(T)$.

Let (t_1, j) denote the incoming tree edge. Suppose that vertex i is in the subtree hanging from t_2 in T. Then consider the tree cut corresponding to edge (j, t_2), i.e. remove edges (j, t_2) and (i, j). Let $S \subset B(T)$ denote the back edges that cover this tree cut. Then we have,

$$x(S) + x(j, t_2) + x(i, j) \;\geq\; 2, \quad x(S) + x(j, t_3) + x(t_1, j) \;\geq\; 2.$$

We can conclude that $x(S) \geq 1 - \epsilon(j)/2$. Thus, as we did previously, we can increase the x'-value of some edge in S by the quantity $\epsilon(j)/2$. The following Lemma follows by the construction of the x' values.

Lemma 14. *The cost of satisfying all of the LP-unsatisfied vertices is at most* $\sum_{j \in T \setminus T_{exp}} \epsilon(j)/2$. *In other words:*

$$\sum_{(u,v) \in B(T)} (x'(u, v) - x(u, v)) \leq \sum_{j \in T \setminus T_{exp}} \frac{\epsilon(j)}{2}.$$

Since all vertices in T are now satisfied by x', the x'-values can be used to compute an upper bound on the cost of a feasible circulation of $C(G, T)$.

Theorem 2. *The function* $x' : B(T) \to [0, 1]$ *corresponds to a feasible circulation of* $C(G, T)$ *with cost at most:*

$$\sum_{j \in T_{exp}} \max\{0,\; c_x(j)\} + \sum_{j \in T} \epsilon(j).$$

Theorem 3. *When* $OPT_{LP}(G) = (1 + \epsilon)n$*, there is a feasible circulation for* $C(G, T)$ *with cost at most* $n/6 + 2\epsilon n$.

4.2 The f-Circulation

Now we describe a new method to obtain a feasible circulation, i.e. how to obtain values $f'(i, j)$ for each edge $(i, j) \in B(T)$ such that each vertex in T is satisfied by f'. The values will be used to demonstrate an improved upper bound on the cost of a circulation of $C(G, T)$ when G is a subquartic graph. In this section, we will prove the following theorem, which implies that the Mömke-Svensson algorithm has an approximation guarantee of 17/12 for graph-TSP on subquartic graphs.

Theorem 4. *When* $OPT_{LP}(G) = (1 + \epsilon)n$*, there is a feasible circulation for* $C(G, T)$ *with cost at most* $n/8 + 2\epsilon n$.

Consider a vertex $v \in T_{exp}$. If both incoming back edges had f-value 1/2, then this vertex would not contribute anything to the cost of the circulation. Thus, on a high level, our goal is to find f-values that are as close to half as possible, while at the same time not creating any additional unsatisfied vertices. The f-value therefore corresponds to a decreased x-value if the x-value is high, and an increased x-value if the x-value is low. A set of f-values corresponding to decreased x-values may pose a problem if they correspond to the set of back

$x_{ij} > 3/4$	$\Rightarrow$	$f(i,j) = 2x_{ij} - 1,$
$x_{ij} < 1/4$	$\Rightarrow$	$f(i,j) = 2x_{ij},$
$1/4 \leq x_{ij} \leq 3/4$	$\Rightarrow$	$f(i,j) = 1/2.$

Fig. 1. Rules for creating the f-values from the x-values

edges that cover an LP-unsatisfied vertex. However, in Section 4.1, we only used $\epsilon(j)/2$ to satisfy an LP-unsatisfied vertex j. We can actually use at least $\epsilon(j)$. This observation allows us to decrease the x-values. We use the rules shown in Figure 1 to determine the values $f : B(T) \to [0,1]$.

Lemma 15. *If a vertex v is LP-satisfied, then it is satisfied by f.*

Definition 13. *For each vertex $j \in T_{exp}$, let $c_f(j) = \sum_{i:(i,j)\in B(T)} f(i,j) - 1 - \epsilon(j)$.*

For ease of notation, set $f'(u,v) = f(u,v)$ for all $(u,v) \in B(T)$.

Lemma 16. *For an LP-unsatisfied vertex $v \in T_{int}$, if we increase by the amount $\epsilon(v)$ the f'-value of an edge that covers its unsatisfied tree cut, then vertex v will be satisfied by f'.*

Lemma 17. *For $j \in T_{exp}$, if $x_{min}(j), x_{max}(j) \geq 1/2$ or if $x_{min}(j), x_{max}(j) \leq 3/4$, then $c_f(j) \leq 0$.*

Lemma 18. *If $x_{max}(j) \geq 3/4$ and $0 < x_{min}(j) \leq 1/2$, then $c_f(j) \leq \min\{x_{min}(j),\ 1/2 - x_{min}(j)\}$.*

Similar to the approach taken in Section 4.1, any LP-unsatisfied vertex can have the value of the edges covering the unsatisfied tree cut by adding $\epsilon(v)$ to one of the edges covering the cut. We now have the following theorem.

Theorem 5. *When $OPT_{LP}(G) = (1+\epsilon)n$, there is a feasible circulation for $C(G,T)$ with cost at most $n/8 + 2\epsilon n$.*

5 Combining the x- and the f-circulations

We can classify each vertex in T_{exp} according to the value of $x_{min}(j)$. Intuitively, if many vertices contribute a lot, say 1/3 to the x-circulation, then they will not contribute a lot of the f-circulation, and vice versa.

$x_{min}(j)$	$c_x(j)$	$c_f(j)$
$[0, 1/4]$	$x_{min}(j)/2$	$x_{min}(j)$
$[1/4, 1/2]$	$x_{min}(j)/2$	$1/2 - x_{min}(j)$
$[1/2, 2/3]$	$x_{min}(j)/2$	0
$[2/3, 1]$	$1 - x_{min}(j)$	0

Theorem 6. *When* $OPT_{LP}(G) = (1+\epsilon)n$*, there is a feasible circulation for* $C(G,T)$ *with cost at most* $n/11 + 2\epsilon n$*.*

Theorem 7. *The approximation guarantee of the Mömke-Svensson algorithm on subquartic graphs is at most* $46/33$*.*

6 Final Remarks

We note that we can obtain a slightly better approximation ratio by allowing an larger coefficient in front of the amount ϵn in Theorem 6. However, the improvement we obtain from this is extremely small (approximately 1.393) and not worth the technical equations.

Acknowledgements. We wish to thank Sylvia Boyd, Satoru Iwata, R. Ravi, András Sebő and Ola Svensson for helpful discussions and comments. This work was done in part while the author was a member of the THL2 group at EPFL.

References

[AKS12] An, H.-C., Kleinberg, R., Shmoys, D.B.: Improving Christofides' algorithm for the s-t-path TSP. In: Proceedings of the 44th Symposium on Theory of Computing, pp. 875–886. ACM (2012)

[BSvdSS11] Boyd, S., Sitters, R., van der Ster, S., Stougie, L.: TSP on cubic and subcubic graphs. In: Günlük, O., Woeginger, G.J. (eds.) IPCO 2011. LNCS, vol. 6655, pp. 65–77. Springer, Heidelberg (2011)

[CFN85] Cornuéjols, G., Fonlupt, J., Naddef, D.: The traveling salesman problem on a graph and some related integer polyhedra. Mathematical Programming 33(1), 1–27 (1985)

[Chr76] Christofides, N.: Worst-case analysis of a new heuristic for the travelling salesman problem. Technical report, DTIC Document (1976)

[Gao13] Gao, Z.: An LP-based-approximation algorithm for the s-t path graph traveling salesman problem. Operations Research Letters 41(6), 615–617 (2013)

[GLS05] Gamarnik, D., Lewenstein, M., Sviridenko, M.: An improved upper bound for the TSP in cubic 3-edge-connected graphs. Operations Research Letters 33(5), 467–474 (2005)

[GSS11] Gharan, S.O., Saberi, A., Singh, M.: A randomized rounding approach to the traveling salesman problem. In: 2011 IEEE 52nd Annual Symposium on Foundations of Computer Science (FOCS), pp. 550–559. IEEE (2011)

[HK70] Held, M., Karp, R.M.: The traveling-salesman problem and minimum spanning trees. Operations Research 18(6), 1138–1162 (1970)

[Hoo91] Hoogeveen, J.A.: Analysis of Christofides' heuristic: some paths are more difficult than cycles. Operations Research Letters 10(5), 291–295 (1991)

[MS11] Mömke, T., Svensson, O.: Approximating graphic TSP by matchings. In: IEEE 52nd Annual Symposium on Foundations of Computer Science, pp. 560–569 (2011)

[Muc12] Mucha, M.: 13/9-approximation for graphic TSP. In: Dürr, C., Wilke, T. (eds.) STACS. LIPIcs, vol. 14, pp. 30–41. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2012)

[NP81] Naddef, D., Pulleyblank, W.R.: Matchings in regular graphs. Discrete Mathematics 34(3), 283–291 (1981)

[Seb13] Sebő, A.: Eight-fifth approximation for the path TSP. In: Goemans, M., Correa, J. (eds.) IPCO 2013. LNCS, vol. 7801, pp. 362–374. Springer, Heidelberg (2013)

[SV12] Sebő, A., Vygen, J.: Shorter tours by nicer ears: 7/5-approximation for graphic TSP, 3/2 for the path version, and 4/3 for two-edge-connected subgraphs. arXiv:1201.1870 (2012)

[Vis12] Vishnoi, N.K.: A permanent approach to the traveling salesman problem. In: 2012 IEEE 53rd Annual Symposium on Foundations of Computer Science, pp. 76–80. IEEE (2012)