Strongly Connected Spanning Subgraph for Almost Symmetric Networks

A. Karim Abu-Affash* Paz Carmi† Anat Parush Tzur‡

Abstract

In the *strongly connected spanning subgraph* ($SCSS$) problem, the goal is to find a minimum weight spanning subgraph of a strongly connected directed graph that maintains the strong connectivity. In this paper, we consider the $SCSS$ problem for two families of geometric directed graphs; *t-spanners* and *symmetric disk graphs*. Given a constant $t \geq 1$, a directed graph G is a t-spanner of a set of points V if, for every two points u and v in V, there exists a directed path from u to v in G of length at most $t \cdot |uv|$, where $|uv|$ is the Euclidean distance between u and v. Given a set V of points in the plane such that each point $u \in V$ has a radius r_u, the symmetric disk graph of V is a directed graph $G = (V, E)$, such that $E = \{(u, v) : |uv| \leq r_u \text{ and } |uv| \leq r_v\}$. Thus, if there exists a directed edge (u, v), then (v, u) exists as well.

We present $\frac{3}{4}(t+1)$ and $\frac{3}{2}$ approximation algorithms for the $SCSS$ problem for t-spanners and for symmetric disk graphs, respectively. Actually, our approach achieves a $\frac{3}{4}(t+1)$-approximation algorithm for all directed graphs satisfying the property that, for every two nodes u and v, the ratio between the shortest paths, from u to v and from v to u in the graph, is at most t.

1 Introduction

A directed graph is said to be *strongly connected* if it contains a directed path from every node to any other node. Given a directed graph $\overrightarrow{G}$, a *spanning subgraph* of $\overrightarrow{G}$ is a subgraph of $\overrightarrow{G}$ that contains all nodes of $\overrightarrow{G}$. In the *strongly connected spanning subgraph* ($SCSS$) problem, one has to find a minimum weight spanning subgraph of a strongly connected directed graph that maintains the strong connectivity. The $SCSS$ problem is a basic network design problem [6] and is known to be NP-hard [4,8]. The NP-hardness can be shown by a simple reduction from the Hamiltonian cycle problem.

For unweighted directed graphs (i.e., all edges have weight 1), Khuller et al. [10,11] proposed a polynomial-time 1.61-approximation algorithm for the $SCSS$ problem. Later, Vetta [16] presented a polynomial-time approximation algorithm achieving an approximation ratio of 3/2. Zhao et al. [17] gave a linear-time 5/3-approximation algorithm. For weighted directed graphs, Frederickson and JáJá [6] studied the $SCSS$ problem and presented a linear-time algorithm achieving an approximation ratio of 2.

Given a set V of points in the plane such that each point $u \in V$ has a radius r_u, the *symmetric disk graph* of V is a directed graph $\overrightarrow{G}= (V, \overrightarrow{E})$, such that $\overrightarrow{E}= \{(u, v) : |uv| \leq r_u \text{ and } |uv| \leq r_v\}$, where $|uv|$ is the Euclidean distance between u and v. The weight of an edge $(u, v) \in\overrightarrow{E}$ (denoted by $wt(u, v)$) is some polynomial function on $|uv|$. This weight function is typically used in wireless networks, where $wt(u, v) = |uv|^\alpha$, for $1 \leq \alpha \leq 5$.

Given a set V of points in the plane and a constant $t \geq 1$, a directed graph $\overrightarrow{G}$ is a (geometric) *t-spanner* of V if, for every two points u and v in V, there exists a directed path from u to v in $\overrightarrow{G}$ of length at most $t \cdot |uv|$.

In this paper, we focus on the $SCSS$ problem for symmetric disk graphs and t-spanners. We present a $\frac{3}{2}$-approximation algorithm for the $SCSS$ problem for symmetric disk graphs. Then, we extend this algorithm to obtain a $\frac{3}{4}(t+1)$-approximation algorithm for the $SCSS$ problem for t-spanners. Our approximation algorithms are based on Christofides' algorithm for the traveling salesman (TSP) problem.

Actually, our approach provides a $\frac{3}{4}(t + 1)$-approximation algorithm for the $SCSS$ problem for an extended family of directed graphs, which is called *t-symmetric*. For a weighted directed graph $\overrightarrow{G}$, let $\delta_{\overrightarrow{G}}(u, v)$ denote a minimum weight path from u to v in $\overrightarrow{G}$. A weighted directed graph $\overrightarrow{G}$ is called a *t-symmetric* directed graph, for a given constant $t \geq 1$, if, for each pair of nodes u and v in $\overrightarrow{G}$, the weight of $\delta_{\overrightarrow{G}}(u, v)$ is at most t times the weight of $\delta_{\overrightarrow{G}}(v, u)$. Given a t-symmetric directed graph $\overrightarrow{G}$ that is strongly connected, the goal is to find a minimum weight strongly connected spanning subgraph of $\overrightarrow{G}$.

The TSP is defined as follows. Given a weighted complete graph on n nodes, the goal is to find a tour, i.e., a

*Software Engineering Department, Shamoon College of Engineering, Beer-Sheva 84100, Israel, abuaa1@sce.ac.il.

†Department of Computer Science, Ben-Gurion University, Beer-Sheva 84105, Israel, carmip@cs.bgu.ac.il. The research is partially supported by the Lynn and William Frankel Center for Computer Science and by grant 680/11 from the Israel Science Foundation (ISF).

‡Department of Computer Science, Ben-Gurion University, Beer-Sheva 84105, Israel, parusha@cs.bgu.ac.il.

simple cycle spanning all the nodes, of minimum weight. Shani and Gonzalez [14] proved that the TSP problem is NP-Complete. In the metric TSP, the weight function of the edge set forms a metric, i.e., the weight function satisfies the triangle inequality; despite this restriction the problem remains NP-hard. A 2-approximation algorithm based on utilizing a minimum spanning tree was proposed in [13]. Christofides [1] improved the algorithm by also utilizing a minimum weight perfect matching, and achieved a 3/2-approximation algorithm.

A connected graph $G = (V, E)$ is called *k-edge-connected* if, for each subset $E' \subseteq E$ of size at most $k-1$, the graph $G' = (V, E \setminus E')$ is also connected. In the *k-edge-connectivity* problem, the goal is to find a minimum weight spanning subgraph of G that is k-edge-connected. The k-edge-connectivity problem has applications in network reliability (besides its theoretical interest), since it ensures that even when $k-1$ links fail , the network remains connected.

The 2-edge-connectivity problem is known to be MAX-SNP-hard [2, 5], as is the unweighted version in which the objective is to minimize the number of edges of the subset. For unweighted graphs, Vempala and Vetta [15] presented a 4/3-approximation algorithm for the 2-edge-connectivity problem. Jothi et al. [9] improved this result by describing a 5/4-approximation algorithm for the 2-edge-connectivity problem. The 3-approximation algorithm for the 2-edge-connectivity problem in weighted graphs that follows from the approximation algorithm of Frederickson and JáJá [6] for the bridge augmenting connectivity problem, was afterwards improved to 2 by Khuller and Vishkin [12]. For weighted complete graphs whose cost function satisfies the triangle inequality, Frederickson and JáJá [7] presented 3/2-approximation algorithm for the 2-edge-connectivity problem. For complete Euclidean graphs in $\mathbb{R}^d$ this problem admits a PTAS [3].

At first glance, the $SCSS$ problem in symmetric disk graphs looks equivalent to the 2-edge-connectivity problem in undirected graphs, since any solution for the 2-edge-connectivity problem is also a solution for the $SCSS$ problem. However, the weight of an optimal solution for the 2-edge-connectivity problem can be $\Omega(n^{\alpha-1})$ times the weight of an optimal solution for the $SCSS$ problem, where the weight of an edge (u, v) is $|uv|^\alpha$ and $\alpha \geq 1.$, as illustrated in Figure 1.

The rest of this paper is organized as follows. In Section 2, we give a $\frac{3}{2}$-approximation algorithm for the $SCSS$ problem in symmetric disk graphs. Then, in Section 3, we extend this algorithm to obtain a $\frac{3}{4}(t+1)$-approximation algorithm for the $SCSS$ problem in t-spanners.

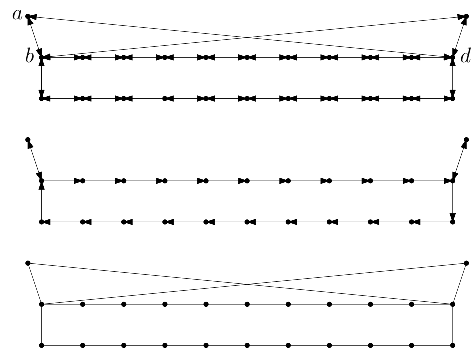

Figure 1: Top, a symmetric disk graph H of n nodes, where all nodes have radius 1 except nodes a, b, c and d that have radius $(n-2)/2$. Middle, an optimal solution for the $SCSS$ problem in H of weight $n+2$. Bottom, the unique solution to the 2-edge-connectivity problem for the undirected version of H, of weight $n + 2(\frac{n-2}{2})^\alpha$, where $wt(u, v) = |uv|^\alpha$.

2 The $SCSS$ problem in symmetric disk graphs

Given a strongly connected symmetric disk graph $\overrightarrow{G}= (V, \overrightarrow{E})$, in the $SCSS$ problem, the goal is to find a minimum weight set $R^* \subseteq \overrightarrow{E}$, such that $G_{R^*} = (V, R^*)$ is strongly connected. Let OPT denote the weight of R^*, i.e., the total weight of the edges in R^*. In this section, we present an algorithm that computes a set $R \subseteq \overrightarrow{E}$, such that the graph $G_R = (V, R)$ is strongly connected and the weight of R is at most $\frac{3}{2} \cdot OPT$.

A pair of nodes u and v in a strongly connected graph $\overrightarrow{G}$ is called a *cut pair* if the edges (u, v) and (v, u) are in G and their removal separates $\overrightarrow{G}$ into two subgraphs. Thus, if $\overrightarrow{G}$ contains a cut pair, then this pair separates the $SCSS$ problem into two independent $SCSS$ subproblems that can be approximated by the proposed algorithm. Moreover, cut pairs must be in any feasible solution for the $SCSS$ problem, and, in particular, in any optimal solution. Therefore, from now on, we assume that no cut pairs exist in $\overrightarrow{G}$.

Let $\delta_{\overrightarrow{G}}(u, v)$ denote a minimum weight path from u to v in $\overrightarrow{G}$, and let $wt(\delta_{\overrightarrow{G}}(u, v))$ denote the weight of $\delta_{\overrightarrow{G}}(u, v)$. The Shortest Paths Graph of $\overrightarrow{G}$ (denoted by $SPG(\overrightarrow{G})$), is an undirected complete graph over V, where the weight of an edge $\{u, v\}$ is $wt(\delta_{\overrightarrow{G}}(u, v))$. Notice that, since $\overrightarrow{G}$ is a symmetric disk graph, $wt(\delta_{\overrightarrow{G}}(u, v)) = wt(\delta_{\overrightarrow{G}}(v, u))$, and therefore, the weight function of the $SPG(\overrightarrow{G})$ is well defined, and it forms a metric.

Our algorithm applies the well known Christofides' algorithm (for the TSP problem) on $SPG(\overrightarrow{G})$.

Christofides' algorithm finds two edge sets, a minimum spanning tree of $SPG(\overrightarrow{G})$ and a minimum weight perfect matching in the complete graph over the nodes of odd degree in the minimum spanning tree. The graph that consists of these two edge sets is connected and all its nodes are of even degree, therefore, it contains an Eulerian cycle. Due to the triangle inequality, the Eulerian cycle can be relaxed into a Hamiltonian cycle (by "shortcutting" whenever a node is revisited) without increasing its weight. It has been shown that the approximation ratio of this algorithm is 3/2 [1].

Given a strongly connected symmetric disk graph $\overrightarrow{G}=(V,\overrightarrow{E})$, in Algorithm 1, we describe how to compute a set $R \subseteq \overrightarrow{E}$, such that $G_R=(V,R)$ is strongly connected. Then, in Section 2.1 we bound the weight of R with respect to OPT.

Algorithm 1

construct $SPG(\overrightarrow{G})$
compute a solution T for the TSP in $SPG(\overrightarrow{G})$ using Christofides' algorithm
direct T arbitrarily and denote this directed tour by $\overrightarrow{T}$
$R \leftarrow \emptyset$
for each edge $(u,v) \in \overrightarrow{T}$ **do**
    $R \leftarrow R \cup \delta_{\overrightarrow{G}}(u,v)$
return R

It is not hard to see that the running time of Algorithm 1 is polynomial ($O(n^3)$), and the resulting graph $G_R=(V,R)$ is strongly connected.

2.1 Approximation ratio

Let R^* be an optimal solution for the $SCSS$ problem in $\overrightarrow{G}=(V,\overrightarrow{E})$, let OPT denote the weight of R^*, and let R be the set obtained by Algorithm 1. In this section, we prove that the weight of R (i.e., $wt(R)$) is at most $\frac{3}{2}\cdot OPT$. Let $\overline{G}_{R^*}$ be the undirected graph of $G_{R^*}=(V,R^*)$, that is, $\overline{G}_{R^*}$ contains an undirected edge between nodes u and v if either $(u,v)\in R^*$ or $(v,u)\in R^*$.

Lemma 1 *If all the nodes in $\overline{G}_{R^*}$ are of even degree, then $wt(R)\le \frac{3}{2}\cdot OPT$.*

Proof. Each edge (u,v) in $\overrightarrow{T}$ (the directed tour that is constructed during Algorithm 1) contributes to R a set $\delta_{\overrightarrow{G}}(u,v)$ of edges that compose a minimum weight path from u to v in $\overrightarrow{G}$. The weight of $\delta_{\overrightarrow{G}}(u,v)$ is equal to the weight of the edge (u,v) in $\overrightarrow{T}$. Notice that we might add to R edges that are already in R. As a result, $wt(R)\le wt(\overrightarrow{T})$. Let T^* denote an optimal solution for the TSP in $SPG(\overrightarrow{G})$. Then, by the bound of Christofides' algorithm, $wt(\overrightarrow{T})\le \frac{3}{2}\cdot wt(T^*)$. Finally, $\overline{G}_{R^*}$ contains an Eulerian cycle C (since all nodes are of even degree) that yields a solution for the TSP in $SPG(\overrightarrow{G})$. Therefore, OPT is an upper bound on the weight of the edge set of T^*, i.e., $wt(T^*)\le OPT$. Therefore, we have

$$wt(R)\le wt(\overrightarrow{T})\le \tfrac{3}{2}\cdot wt(T^*)\le \tfrac{3}{2}\cdot OPT\,. \qquad \square$$

In general, the inequality $wt(T^*)\le OPT$ does not hold without the restriction of even degree on the nodes in $\overline{G}_{R^*}$. To see this, consider the example in Figure 2. The weight of any optimal solution T^* for the TSP in $SPG(\overrightarrow{G})$ is of weight $(4\cdot OPT-10)/3$. Thus, $wt(T^*)\ge(4/3-\epsilon)OPT$, for any $\epsilon>0$.

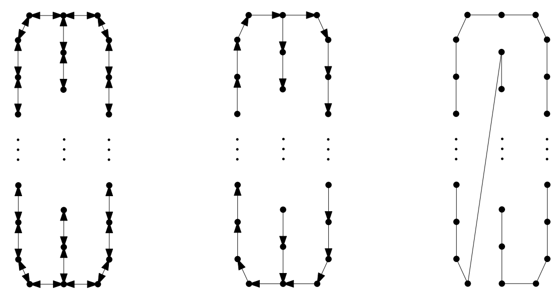

Figure 2: Left, a symmetric disk graph $\overrightarrow{G}$ on n nodes, in which the weight of each edge is 1. Middle, an optimal solution for the $SCSS$ problem in $\overrightarrow{G}$ of weight $n+1$. Right, an optimal solution T^* for the TSP in $SPG(\overrightarrow{G})$ of weight $\frac{4}{3}n-2$.

Lemma 2 *Let $G_{\Delta\le 3}=(V,E)$ be a 2-edge-connected undirected graph whose maximum degree is 3. Then, $G_{\Delta\le3}$ contains a path composed of edges $E_p=\{(v_1,v_2),(v_2,v_3),\dots,(v_{k-1},v_k)\}$, such that $v_1\ne v_k$, v_1 and v_k are of degree 3, each node in $V_p=\{v_2,\dots,v_{k-1}\}$ is of degree 2, and $(V\setminus V_p, E\setminus E_p)$ is 2-edge-connected. We call such a path a chord.*

Proof. We show the existence of such a chord using a constructive method. In each iteration i, we maintain a 2-edge-connected component C_i and extend C_i via an unexplored node $v^*\in C_i$ of degree 3. Initially, $i=0$, C_i is a cycle, and $v^*\in C_i$ is a node of degree 3. Let P_i be a path connecting v^* to a node $u\in C_i$ that is edge disjoint from C_i (such a path exists since otherwise $G_{\Delta\le3}$ is not 2-edge-connected). If the inner nodes of P_i are of degree 2 then P_i is a chord, and we are done. Otherwise, P_i contains a node w of degree 3. Let $C_{i+1}=C_i\cup P_i$ and set v^* to be w. Repeat this procedure until a chord is found. This procedure halts, since in each iteration a new node v^* of degree 3 is explored. □

Lemma 3 *Let P be a simple path composed of vertices $V_p = \{v_1, v_2, \ldots, v_k\}$ and edges $E_p = \{(v_1, v_2), (v_2, v_3), \ldots, (v_{k-1}, v_k)\}$. There exists a perfect matching M_p in P of the nodes in V_p except for at most the two end-vertices v_1 and v_k, i.e., $V_p \setminus W$, where $W \subseteq \{v_1, v_k\}$, such that the weight of M_p is at most half of the weight of P.*

Proof. The correctness follows from the pigeonhole principle for both cases of the parity of k.

- If k is odd, then one of the two matchings $\{(v_1, v_2), (v_3, v_4), \ldots, (v_{k-2}, v_{k-1})\}$ or $\{(v_2, v_3), (v_4, v_5), \ldots, (v_{k-1}, v_k)\}$ is at most half of the weight of P.
- If k is even, then one of the two matchings $\{(v_1, v_2), (v_3, v_4), \ldots, (v_{k-1}, v_k)\}$ or $\{(v_2, v_3), (v_4, v_5), \ldots, (v_{k-2}, v_{k-1})\}$ is at most half of the weight of P. □

Lemma 3 yields the following corollary.

Corollary 4 *Let P, E_p, and V_p be as in Lemma 3, and let $V'_p \subseteq V_p$. There exists a perfect matching M'_p of the nodes in $V'_p \cup \{v_1, v_k\} \setminus W$, where $W \subseteq \{v_1, v_k\}$, such that the weight of M'_p is at most half of the weight of P, where the weight of an edge $\{v_i, v_j\}$ in M'_p is the weight of the subpath between v_i and v_j in P.*

Let $\mathcal{T}$ be the minimum spanning tree of $SPG(\overrightarrow{G})$ that is found during Christofides' algorithm. Let V_{odd} be the set of nodes of odd degree in $\mathcal{T}$, let $G_{odd} = (V_{odd}, E_{odd})$ be the (complete) subgraph of $SPG(\overrightarrow{G})$ induced by V_{odd} (E_{odd} is the set of all edges of $SPG(\overrightarrow{G})$ having both end-vertices in V_{odd}), and let $\mathcal{M}$ denote a minimum weight perfect matching of G_{odd}. Recall that R^* is an optimal solution for the $SCSS$ problem in $\overrightarrow{G}$ of weight OPT. In the following, we bound the weights of $\mathcal{T}$ and $\mathcal{M}$ with respect to OPT.

Lemma 5 $wt(\mathcal{T}) \leq OPT$.

Proof. Since the graph $G_{R^*} = (V, R^*)$ is a spanning subgraph of $\overrightarrow{G}$ that is strongly connected, the undirected graph $\overline{G}_{R^*}$ of G_{R^*} contains a spanning tree T of weight at most OPT. Let $\{u, v\}$ be an edge in T such that, w.l.o.g., it is the undirected edge of (u, v) in G_{R^*}. Since $SPG(\overrightarrow{G})$ is a complete graph over V, it also contains T, and the weight of the edge $\{u, v\}$ in $SPG(\overrightarrow{G})$ is equal to the weight of a minimum weight path from u to v in $\overrightarrow{G}$. Thus, the weight of $\{u, v\}$ in $SPG(\overrightarrow{G})$ is equal to the weight of (u, v) in G_{R^*}. Therefore,

$$wt(\mathcal{T}) \leq wt(T) \leq OPT.$$ □

Lemma 6 $wt(\mathcal{M}) \leq OPT/2$.

Proof. Let $G' = (V', R')$, where $V' \subseteq V$ and $R' \subseteq R^*$, be the minimum weight subgraph of $G_{R^*} = (V, R^*)$, in which the nodes of V_{odd} are strongly connected. Clearly, $wt(R') \leq OPT$. We first show that G' can be converted to a graph $G'_{\Delta \leq 3}$ (i.e., a 2-edge connected undirected graph with degree at most 3) whose weight is equal to $wt(G')$, and thus, $wt(G'_{\Delta \leq 3}) \leq OPT$. Then, we show that there exists a perfect matching M' of V_{odd} in $G'_{\Delta \leq 3}$, such that $wt(M') \leq \frac{1}{2} \cdot wt(G'_{\Delta \leq 3})$.

Let $\overline{G'}$ be the undirected graph of G', such that $\overline{G'}$ contains an undirected edge $\{u, v\}$ between nodes u and v if either $(u, v) \in R'$ or $(v, u) \in R'$, and $wt(\{u, v\}) = wt(u, v)$. If both (u, v) and (v, u) are in R', then $\overline{G'}$ contains two undirected edges $\{u, v\}$ and $\{u, v\}'$ between u and v, each of weight $wt(u, v)$. Notice that $\overline{G'}$ is a 2-edge connected undirected graph with the same weight as G', and the minimum degree of each node in $\overline{G'}$ is 2. Moreover, if $\overline{G'}$ contains two edges $\{u, v\}$ and $\{u, v\}'$, then the nodes u and v are a cut pair in G'. We show how to convert $\overline{G'}$ to $G'_{\Delta \leq 3}$. First, while there exists a node u of degree greater than 3 in $\overline{G'}$ that is incident to two edges $\{u, v\}, \{u, v\}'$, select an adjacent node $w \neq v$ of u in $\overline{G'}$. Add the edge $\{v, w\}$ of weight $wt(\{u, v\}) + wt(\{u, w\})$ to $\overline{G'}$, and remove the edge $\{u, w\}$ and $\{u, v\}'$ from $\overline{G'}$. At this stage, $\overline{G'}$ does not contain any cut pairs, i.e., if the number of vertices in $\overline{G'}$ is greater than two, then there is no node u in $\overline{G'}$ that is incident to two edges $\{u, v\}, \{u, v\}'$.

Next, while there exists a node u of degree greater than 3 in $\overline{G'}$, select an adjacent node w of u in $\overline{G'}$. Since $\overline{G'}$ is 2-edge connected undirected graph, there is a path $P_{wu} = (w, \ldots, w', u)$ in $\overline{G'}$ from w to u which is different from the edge $\{w, u\}$. Notice that w' is the last node before u in this path P_{wu}, and let $v \notin \{w', w\}$ be a node that is adjacent to u in $\overline{G'}$. Add the edge $\{v, w\}$ of weight $wt(\{u, v\}) + wt(\{u, w\})$ to $\overline{G'}$, and remove the edges $\{u, v\}$, $\{u, w\}$ from $\overline{G'}$.

The obtained graph is 2-edge connected undirected graph. Since, in each iteration, the degree of one node is reduced (by two), and the degree of the other nodes remains the same, this routine ends. Moreover, for each edge that is added to the graph, two edges with equal total weight are removed and, thus, the weight of the graph $\overline{G'}$ is preserved. At the end of this routine, set $G'_{\Delta \leq 3}$ to be $\overline{G'}$.

We now show that there exists a perfect matching M' of V_{odd} in $G'_{\Delta \leq 3}$, such that (i) each edge in M' corresponds to a path in $G'_{\Delta \leq 3}$; (ii) the weight of each edge $e \in M'$ is equal to the weight of the corresponding path of e in $G'_{\Delta \leq 3}$; and (iii) $wt(M') \leq \frac{1}{2} \cdot wt(G'_{\Delta \leq 3})$.

The existence of such a matching M' is shown in Procedure 2. In each iteration (Lines 2–19), the number of nodes of degree 3 in G_{temp} is reduced by 2, thus, this while loop ends, and at Line 20 the resulting graph is

Procedure 2 Constructing a matching M'

1: $M' \leftarrow \emptyset$, $G_{temp} \leftarrow G'_{\Delta \leq 3}$
2: **while** there is a node v in G_{temp} of degree 3 **do**
3: let $P = (V_P, E_P)$ be a chord in G_{temp}
/* *Such P exists by Lemma 2* */
4: let U be the set of the two endvertices of P
5: let $V'_P \leftarrow V_{odd} \cap V_P$
6: let M_{chord} be a perfect matching in P of the nodes in $V'_P \cup U \setminus W$, where $W \subseteq U$, such that $wt(M_{chord}) \leq \frac{1}{2} wt(P)$
/* *Such M_{chord} exists by Corollary 4* */
7: $M' \leftarrow M' \cup \{\{v_i, v_j\} | v_i, v_j \in V'_P\}$
8: $G_{temp} \leftarrow G_{temp} \setminus (P \setminus U)$
/* *Remove all inner nodes of P and their incident edges from G_{temp}* */
9: **for** each $v \in U$ such that $\{v_i, v\} \in M_{chord}$ **do**
10: let p and q be the two nodes adjacent to v that are not in P
11: **if** $v \in V'_P$ **then**
12: add the edge $\{p, q\}$ to G_{temp}
13: set $wt(\{p, q\})$ to be $wt(\{p, v\}) + wt(\{v, q\})$
14: remove v and its incident edges from G_{temp}
15: **else**
16: let $P_{v_i, v}$ be the path from v_i to v in P that corresponds to the edge $\{v_i, v\}$
17: replace v by v_i in G_{temp},
18: set $wt(\{p, v_i\})$ to be $wt(\{p, v\})$
19: set $wt(\{q, v_i\})$ to be $wt(\{q, v\})$
20: let M_c be a perfect matching in G_{temp} of the nodes in V'_P, such that $wt(M_c) \leq \frac{1}{2} wt(G_{temp})$
/* *At this stage, G_{temp} is a cycle* */
21: $M' \leftarrow M' \cup M_c$
22: **return** M'

a 2-edge-connected graph with nodes of degree 2, i.e., a cycle C. The number of nodes in V_{odd} is even, and while removing a chord from G_{temp}, an even number of nodes from V_{odd} are removed. Therefore, C contains an even number of nodes from V_{odd}.

In the following we bound the weight of M' obtained by Procedure 2. The weight of the matching found at Line 6 is at most $\frac{1}{2} \cdot wt(P)$. Thus, at Line 7, we add to M' at most half of the weight of the path P. Then, at Line 8, the edges of P are removed from G_{temp}, and these edges are not charged again. Clearly, the same bound holds for the matching that is found at Line 20. Thus, the weight of M' is bounded by half of the weight of the edge set of $G'_{\Delta \leq 3}$, i.e., $wt(M') \leq \frac{1}{2} \cdot wt(G'_{\Delta \leq 3})$.

Consider a node $v \notin V_{odd}$ such that $\{v_i, v\} \in M_{chord}$ in some iteration j of the while loop. Notice that the weight $wt(\{v_i, v\})$ is charged in this iteration, even though the edge $\{v_i, v\}$ is not added to M'. This is done to compensate that later, in some iteration $j' > j$, the node v_i is matched to some node $v_l \in V_{odd}$, and the weight $wt(\{v_i, v_l\})$ corresponds to the weight $wt(\{v, v_l\})$ (see Lines 18 or 19). Therefore, the weight $wt(\{v_i, v_l\})$ might not include the weight $wt(\{v_i, v\})$. However, as mentioned, this does not affect the bound on the weight of the matching M', since the weight $wt(\{v_i, v\})$ has already been charged in the iteration j.

In order to prove the lemma, we generate a perfect matching M^* in G_{odd} based on M'. For each edge $\{v_i, v_j\} \in M'$, we add to M^* the edge $\{v_i, v_j\}$ of G_{odd}. Each edge $\{v_i, v_j\}$ has a corresponding path from v_i to v_j in $G_{\Delta \leq 3}$, i.e., an equivalent (in weight) path from v_i to v_j in $\overrightarrow{G}$, and, therefore, the weight of the edge $\{v_i, v_j\}$ in M' is an upper bound on the weight of the edge $\{v_i, v_j\}$ in G_{odd}, so, $wt(M^*) \leq wt(M')$. Recall that $wt(G'_{\Delta \leq 3}) \leq OPT$. To sum up, we found a perfect matching in G_{odd} of weight at most half of the weight of R^*. Clearly, the weight of the perfect matching found is an upper bound on the weight of a minimum one, $\mathcal{M}$. Thus, we have $wt(\mathcal{M}) \leq wt(M^*) \leq wt(M') \leq \frac{1}{2} \cdot wt(G'_{\Delta \leq 3}) \leq \frac{1}{2} \cdot OPT$. $\square$

Theorem 7 *Algorithm 1 is a $\frac{3}{2}$-approximation algorithm for the SCSS problem in symmetric disk graphs.*

Proof. $wt(R) \leq wt(\overrightarrow{T}) \leq wt(\mathcal{T}) + wt(\mathcal{M}) \leq \frac{3}{2} \cdot OPT$, where the first inequality is already noted in the proof of Lemma 1, the second inequality follows immediately from the description of Christofides' algorithm, and the last inequality holds due to Lemma 5 and Lemma 6. $\square$

3 The $SCSS$ problem in t-spanners

Given a set V of points in the plane and a constant $t \geq 1$, a directed graph $\overrightarrow{G}$ is a *t-spanner* of V if, for every two points u and v in V, there exists a directed path from u to v in $\overrightarrow{G}$ of length at most $t \cdot |uv|$. In this section, we generalize Theorem 7 for t-spanners. The Shortest Paths Graph of a t-spanner $\overrightarrow{G}$ of V (denoted by $SPG(\overrightarrow{G})$), is an undirected complete graph over V, in which the weight of an edge $\{u, v\}$ equals to $\min\{wt(\delta_{\overrightarrow{G}}(u, v)), wt(\delta_{\overrightarrow{G}}(v, u))\}$, where $\delta_{\overrightarrow{G}}(u, v)$ is a minimum weight path from u to v in $\overrightarrow{G}$.

Theorem 8 *Algorithm 3 is a $\frac{3}{4} \cdot (t+1)$-approximation algorithm for the SCSS problem in t-spanners.*

Proof. Let E_t be the tour computed during Algorithm 3. Consider an edge $\{u, v\} \in E_t$ of weight $\min\{wt(\delta_{\overrightarrow{G}}(u, v)), wt(\delta_{\overrightarrow{G}}(v, u))\}$, and assume, w.l.o.g., that $wt(\{u, v\}) = \delta_{\overrightarrow{G}}(u, v)$. Since the graph $\overrightarrow{G}$ is t-spanner,

$$
\begin{aligned}
wt(\delta_{\overrightarrow{G}}(u, v)) + wt(\delta_{\overrightarrow{G}}(v, u)) &\leq wt(\delta_{\overrightarrow{G}}(u, v)) + t \cdot |uv| \\
&\leq wt(\delta_{\overrightarrow{G}}(u, v)) + t \cdot wt(\delta_{\overrightarrow{G}}(u, v)) \\
&= (t+1) \cdot wt(\delta_{\overrightarrow{G}}(u, v)) \\
&= (t+1) \cdot wt(\{u, v\}).
\end{aligned}
$$

Algorithm 3

construct $SPG(\overrightarrow{G})$ of $\overrightarrow{G}$
compute an Eulerian tour E_t using Christofides' algorithm (the tour before the shortcuts)
let $\overrightarrow{E_t}$ be a directed tour obtained by traversing the Eulerian tour E_t arbitrary
let $\overleftarrow{E_t}$ denote the opposite directed tour of $\overrightarrow{E_t}$
$\overrightarrow{R} \leftarrow \emptyset$, $\overleftarrow{R} \leftarrow \emptyset$
traverse the edges of $\overrightarrow{E_t}$, (resp. $\overleftarrow{E_t}$) and, for each edge (u, v) visited during the traversal, add the set of directed edges $\delta_{\overrightarrow{G}}(u, v)$ to $\overrightarrow{R}$ (resp. $\overleftarrow{R}$)
if $wt(\overrightarrow{R}) \leq wt(\overleftarrow{R})$ **then**
    return $\overrightarrow{R}$
else
    return $\overleftarrow{R}$

We now bound the the output of Algorithm 3.

$$\begin{aligned}
\min\{wt(\overrightarrow{R}),wt(\overleftarrow{R})\} & \\
&\leq \frac{1}{2} \cdot \left(wt(\overrightarrow{R}) + wt(\overleftarrow{R}) \right) \\
&\leq \frac{1}{2} \cdot \sum_{\{u,v\} \in E_t} \left(wt(\delta_{\overrightarrow{G}}(u, v)) + wt(\delta_{\overrightarrow{G}}(v, u)) \right) \\
&\leq \frac{1}{2} \cdot \sum_{\{u,v\} \in E_t} (t+1) \cdot wt(\{u, v\}) \\
&= \frac{1}{2} \cdot (t+1) \cdot wt(E_t) \\
&\leq \frac{3}{4} \cdot (t+1) \cdot OPT,
\end{aligned}$$

where the later inequality follows from Theorem 7. □

Corollary 9 *Algorithm 3 is a $\frac{3}{4} \cdot (t+1)$-approximation algorithm for the SCSS problem in any graph $\overrightarrow{G}$, where the weight of $\delta_{\overrightarrow{G}}(u, v)$ is at most t times the weight of $\delta_{\overrightarrow{G}}(v, u)$, for each pair of nodes u and v in $\overrightarrow{G}$.*

References

[1] N. Christofides. Worst-case analysis of a new heuristic for the traveling salesman problem. *Technical Report 388, Graduate School of Industrial Administration, Carnegie-Mellon University*, 1976.

[2] B. Csaba, M. Karpinski, and P. Krysta. Approximability of dense and sparse instances of minimum 2-connectivity, TSP and path problems. In *SODA*, pages 74–83, 2002.

[3] A. Czumaj and A. Lingas. On approximability of the minimum-cost k-connected spanning subgraph problem. In *SODA*, pages 281–290, 1999.

[4] K. P. Eswaran and R. E. Tarjan. Augmentation problems. *SIAM J. Comput.*, 5(4):653–665, 1976.

[5] C. G. Fernandes. A better approximation ratio for the minimum size k-edge-connected spanning subgraph problem. *J. Algorithms*, 28(1):105–124, 1998.

[6] G. N. Frederickson and J. JáJá. Approximation algorithms for several graph augmentation problems. *SIAM J. Comput.*, 10(2):270–283, 1981.

[7] G. N. Frederickson and J. JáJá. On the relationship between the biconnectivity augmentation and travelling salesman problems. *Theoretical Computer Science*, 19(2):189 – 201, 1982.

[8] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman & Co., New York, NY, USA, 1990.

[9] R. Jothi, B. Raghavachari, and S. Varadarajan. A 5/4-approximation algorithm for minimum 2-edge-connectivity. In *SODA*, pages 725–734, 2003.

[10] S. Khuller, B. Raghavachari, and N. Young. Approximating the minimum equivalent directed graph. *SIAM J. Comput.*, 24(4):859–872, 1995.

[11] S. Khuller, B. Raghavachari, and N. E. Young. On strongly connected directed graphs with bounded cycle length. *Discrete Applied Mathematics*, 69(3):281–289, 1996.

[12] S. Khuller and U. Vishkin. Biconnectivity approximations and graph carvings. *J. ACM*, 41(2):214–235, 1994.

[13] D. J. Rosenkrantz, R. E. Stearns, and P. M. Lewis II. An analysis of several heuristics for the traveling salesman problem. *SIAM J. Comput.*, 6(3):563–581, 1977.

[14] S. Sahni and T. Gonzalez. P-complete approximation problems. *J. ACM*, 23(3):555–565, 1976.

[15] S. Vempala and A. Vetta. Factor 4/3 approximations for minimum 2-connected subgraphs. In *APPROX*, pages 262–273, 2000.

[16] A. Vetta. Approximating the minimum strongly connected subgraph via a matching lower bound. In *SODA*, pages 417–426, 2001.

[17] L. Zhao, H. Nagamochi, and T. Ibaraki. A linear time 5/3-approximation for the minimum strongly-connected spanning subgraph problem. *Inf. Process. Lett.*, 86(2):63–70, 2003.