Article

Approximating the Minimum Tour Cover of a Digraph

Viet Hung Nguyen

LIP6, Université Pierre et Marie Curie Paris 6, 4 place Jussieu, Paris,
France; E-Mail: hung.nguyen@lip6.fr

Abstract: Given a directed graph G with non-negative cost on the arcs, a directed tour cover T of G is a cycle (not necessarily simple) in G such that either head or tail (or both of them) of every arc in G is touched by T. The minimum directed tour cover problem (DToCP), which is to find a directed tour cover of minimum cost, is NP-hard. It is thus interesting to design approximation algorithms with performance guarantee to solve this problem. Although its undirected counterpart (ToCP) has been studied in recent years, in our knowledge, the DToCP remains widely open. In this paper, we give a $2\log_2(n)$-approximation algorithm for the DToCP.

Keywords: approximation algorithm; graph algorithm; packing and covering; tour cover; tree cover

1. Introduction

Let $G = (V, A)$ be a directed graph with a (non-negative) cost function $c : A \Rightarrow \mathbb{Q}_+$ defined on the arcs. A *directed tour cover* T is a subgraph $T = (U, F)$ such that

1. For every $e \in A$, F contains an arc f intersecting e, *i.e.*, f and e have at least one end-node in common.

2. T is a closed directed walk (or branching in the case of tree cover).

We consider in this paper *the minimum directed tour cover (DToCP)* problem which is to find a directed tour cover of minimum cost. As in the previous works [1,2] on the undirected tour cover, we allow the directed tour cover to be a closed directed walk, *i.e.*, it may visit a node more than once, to

avoid restricting the cost function c to be a metric. However, if the cost function c satisfies the triangle inequality, this directed walk can be short-cut into a simple directed cycle which covers every arc in A without increasing the cost. The undirected version of DToCP is the minimum tour cover (ToCP) problem which has been studied in recent years. The ToCP was introduced in a paper by Arkin *et al.* [1]. The authors proved that ToCP is NP-hard and gave a purely combinatorial 5.5-approximation algorithm. Later, Konemann *et al.* [2] proposed an integer formulation for ToCP and used it to improve the approximation ratio to 3. In this paper, the DToCP was listed as an open problem.

We can prove that DToCP is NP-hard by a reduction from the metric Asymmetric Traveling Salesman Problem (ATSP). The proof is very similar to the one for ToCP described in [1]. Given any instance $G = (V, A)$ of ATSP, for every node $v \in V$, we create a new node v' and an arc (v, v') of cost $+\infty$ to obtain a new graph G'. Let us consider the DToCP on G', we can see that any directed tour cover on G' must cover V. As the costs are metric, given any directed tour cover, we can find a traveling salesman tour on G of no greater cost. Hence, there is an one-to-one correspondence between the optimal solutions of DToCP on G' and the optimal traveling salesman tours on G. Moreover, this reduction is approximation-preserving, *i.e.*, if we have an α-approximation algorithm for DToCP then we can approximate the metric ATSP with the same ratio α. In our knowledge, the best approximation ratio for ATSP is so far $0.824 \log_2(n)$ achieved by Kaplan *et al.* [3]. Therefore, it is rather difficult to design an algorithm of a constant approximation ratio for DToCP. In this paper, we will present a $2\log_2(n)$-approximation algorithm for the DToCP. This ratio is more than 2 times the ratio for ATSP, since our algorithm is inspired from the algorithm given by Konemann *et al.* [2] for the undirected case which is based on the bound of the Held-Karp relaxation for metric TSP. It is proved that the metric TSP costs no more than $\frac{3}{2}$ the Held-Karp bound for the undirected case and the metric ATSP costs no more than $\log_2(n)$ the Held-Karp bound for the directed case [4,5]. Hence, our ratio $2\log_2(n)$ for DToCP follows logically the best ratio 3 for ToCP. Our work for DToCP is motivated by the asymmetric versions of the applications enumerated by Arkin *et al.* for the undirected case, especially a special case of the Symmetric Traveling Purchaser Problem [1]. We give here a concrete example of applications of DToCP, which is the following simplified version of the waste collection problem. A company is in charge of waste collection in an urban area. The area is supposed to be small enough so that one truck is sufficient to collect all of its daily wastes. The gather is done by the truck in early morning of each day. For each street in the area, waste should be regrouped at one of the end-points of the street in the night of the day before. When the truck goes along a street, this induces some cost to the company. Hence, the aim of the company is to find a cheapest tour from its depot for the truck such that the waste collection is accomplished for every street of the area. Note that streets could be one-way or sloping, making the costs asymmetric. This problem suggests that the depot should belong to the tour. We shall see that our algorithm for DToCP will also work for the case when the tour should contain a fixed vertex.

The paper is organized as follows: In Section 2, we present an integer formulation for DToCP. We discuss in Section 3 about the parsimonious property of Eulerian directed graphs, in particular in Section 3.1. We prove the equivalent directed version of Goemans and Bertsimas's theorem on splitting operations. In Section 3.2, we derive an equivalent linear program to the Held-Karp relaxation for metric ATSP. This linear program will be useful in the analysis of our algorithm's performance guarantee in Section 5. Before that, we state our approximation algorithm for DToCP in Section 4.

Let us introduce the notations that will be used in the paper. Let $G = (V, A)$ be a digraph with vertex set V and arc set A. If $x \in \mathbb{Q}^{|A|}$ is a vector indexed by the arc set A and $F \subseteq A$ is a subset of arcs, we use $x(F)$ to denote the sum of values of x on the arcs in F, $x(F) = \sum_{e \in F} x_e$. Similarly, for a vector $y \in \mathbb{Q}^{|V|}$ indexed by the vertices and $S \subseteq V$ is a subset of vertices, $y(S)$ denotes the sum of values of y on the vertices in the set S. For a subset of vertices $S \subseteq V$, let $A(S)$ denote the set of the arcs having both end-nodes in S. Let $\delta^+(S)$ and $\delta^-(S)$ denote the set of the arcs having only the tail and head in S, respectively. Let $\delta(S) = \delta^+(S) \cup \delta^-(S)$. We will call $\delta^+(S)$ the *outgoing cut* associated to S, $\delta^-(S)$ the *ingoing cut* associated to S and $\delta(S)$ simply the cut associated to S. For each vertex $v \in V$, let $d^+(v)$ be the *outgoing degree* of v which is equal to the number of arcs in G having v as tail and let $d^-(v)$ be the *ingoing degree* of v which is equal to the number of arcs in G having v as head. For two subset $U, W \subset V$ such that $U \cap W = \emptyset$, let $(U : W)$ be the set of the arcs having the tail in U and the head in W. For $u \in V$, we specify v as an *outneighbor* of u if $(u, v) \in A$ and as an *inneighbor* of u if $(v, u) \in A$. For any $u, v \in V$, let $p(u, v)$ be the number of arc-disjoint paths from u to v. For the sake of simplicity, in clear contexts, the singleton $\{u\}$ will be denoted simply by u. When we work on more than one graph, we specify the graph in the index of the notation, e.g., $\delta_G^+(S)$ will denote $\delta^+(S)$ in the graph G.

2. Integer Formulation

Let
$$\mathcal{F} = \{S \subseteq V \mid A(S) \neq \emptyset, A(V \setminus S) \neq \emptyset\}$$

Given any tour cover TC of G, by definition, for any $S \in \mathcal{F}$ we have $TC \cap \delta^+(S) \geq 1$ and $TC \cap \delta^-(S) \geq 1$. Note that the condition $TC \cap \delta^-(S) \geq 1$ is equivalent to $TC \cap \delta^+(V \setminus S) \geq 1$, and by definition of $\mathcal{F}$, a vertex subset S belongs to $\mathcal{F}$ if and only if $V \setminus S$ belongs to $\mathcal{F}$. This observation motivates our integer formulation for DToCP. For any $e \in A$, let x_e indicate the number of copies of e included in the tour cover. We minimize the total weight of arcs included, under the condition that every outgoing cut associated to some $S \in \mathcal{F}$ should be crossed at least one. In order to ensure that our solution is a tour we also need to specify that for any node $v \in V$, the number of arcs entering v is equal to the number of arcs leaving v in the tour. This integer formulation can be stated as follows:

$$\min \sum_{e \in A} c_e x_e$$

$$\sum_{e \in \delta^+(v)} x_e = \sum_{e \in \delta^-(v)} x_e \qquad \text{for all } v \in V$$

$$\sum_{e \in \delta^+(S)} x_e \geq 1 \qquad \text{for all } S \in \mathcal{F}$$

$$x_e \text{ integer} \qquad \text{for all } e \in A$$

If one specific vertex u should belong to the optimal tour, we just need to set $\sum_{e \in \delta^+(u)} x_e = \sum_{e \in \delta^-(u)} x_e \geq 1$ in the formulation. Replacing the integrality constraints by

$$x_e \geq 0 \text{ for all } e \in A$$

we obtain the linear programming relaxation. We use DToC(G) to denote the convex hull of all vectors x satisfying the above constraints (those of the linear programming relaxation). Clearly minimizing the linear cost function $\sum_{e \in A} c_e x_e$ over DToC(G) can be done in polynomial time since the separation problem of the cut constraint can be solved in polynomial time. Indeed, given a candidate solution x and for every $e \in A$, let us consider x_e as the capacity on arc e. For each pair of arcs e_1, $e_2 \in E$, we compute the minimum capacity cuts $\delta^+(U)$ separating them.

3. Parsimonious Property for Directed Eulerian Graphs

3.1. Connectivity and the Splitting Operation

Given a positive integer k, suppose that we want to find a minimum cost strongly k-connected Eulerian directed subgraph of G, *i.e.*, there are at least k arcs-disjoint paths between every pair of vertices. Moreover, each vertex in the subgraph should have the outgoing and ingoing degree both equal to k. The metric ATSP is a special case of this problem when $k = 1$. An integer formulation for the problem can be stated as follows.

$$\min \sum_{e \in A} c_e x_e \tag{1}$$

subject to

$$
\begin{aligned}
x(\delta^+(u)) = k && \text{for all } u \in V \\
x(\delta^-(u)) = k && \text{for all } u \in V \\
x(\delta^+(S)) \geq k && \text{for all } S \subset V \text{ such that } |S| \geq 2 \\
x_e \geq 0 \text{ and integer} && \text{for all } e \in A
\end{aligned}
$$

Let us consider the linear programming relaxation of (1).

$$\min \sum_{e \in A} c_e x_e \tag{2}$$

subject to

$$
\begin{aligned}
x(\delta^+(u)) = k && \text{for all } u \in V \\
x(\delta^-(u)) = k && \text{for all } u \in V \\
x(\delta^+(S)) \geq k && \text{for all } S \subset V \text{ such that } |S| \geq 2 \\
x_e \geq 0 && \text{for all } e \in A
\end{aligned}
$$

The parsimonious property for Eulerian directed graphs says that if the costs are metric, in (2), we do not need to impose that the degree of each vertex should be exactly k but only at least k. This means that the following linear program

$$\min \sum_{e \in A} c_e x_e \tag{3}$$

subject to

$$x(\delta^+(u)) \geq k \qquad\qquad \text{for all } u \in V$$

$$x(\delta^+(u)) = x(\delta^-(u)) \qquad\qquad \text{for all } u \in V$$

$$x(\delta^+(S)) \geq k \qquad\qquad \text{for all } S \subset V \text{ such that } |S| \geq 2$$

$$x_e \geq 0 \qquad\qquad \text{for all } e \in A$$

is equivalent to (2).

Theorem 1 *If the costs* c *satisfy the triangle inequality then the optimum of* (3) *is equal to the optimum of the linear programming relaxation* (2).

The parsimonious property can be also formulated for undirected Eulerian graphs. We simply replace the outgoing and ingoing cuts in (3) and (2) by the corresponding cut to obtain the two linear programs for the undirected case.

The undirected version of Theorem 1 (in a little more general form) has been proved by Goemans and Bertsimas [6]. Their proof consists mainly in showing the following lemma.

Lemma 1 *[6] Let $G = (V, E)$ be an Eulerian multigraph. Let $c_G(i, j)$ $(i, j \in V)$ denote the maximum number of edge-disjoint paths between i and j. Let s be any vertex of G and let u be any neighbor of s. Then there exists another neighbor of s, say v, such that, by splitting su and sv,* i.e., *removing the edges su and sv and adding the edge uv, we obtain a multigraph G' satisfying the following conditions:*

1. *$c_{G'}(i, j) = c_G(i, j)$ for all $i, j \in V \setminus \{s\}$ and*

2. *$c_{G'}(s, j) = \min(c_G(s, j), d_G(s) - 2)$ for all $j \in V \setminus \{s\}$, where $d_G(s)$ represents the degree of the vertex s in G.*

Condition 1 of Lemma 1 is a result due to Lovasz [7] on the connectivity properties of Eulerian multigraphs. Condition 2, added by Goemans and Bertsimas, states that splitting operation can be performed while maintaining most connectivity requirements involving vertex s. For the connectivity of Eulerian multi-digraphs, Jackson [8] shows a similar result as Lovasz's one (*i.e.*, Condition 1) for Eulerian multigraphs. This result can be stated in the following lemma.

Lemma 2 *[8] Given an Eulerian graph G. Let s be any node of G and let u be any outneighbor of s. Then there exists another inneighbor of s, say v, such that, by splitting (s, u) and (v, s),* i.e., *removing the arcs (s, u) and (v, s) and adding the arc (v, u), we obtain a multigraph G' satisfying*

1. *$p_{G'}(i, j) = p_G(i, j)$ for all $i, j \in V \setminus \{s\}$.*

Like the work of Goemans and Bertsimas for the undirected case, we will subsequently prove that we can add a similar condition as Condition 2 of Lemma 1 to Lemma 2 which express the connectivities involving s. We obtain the following lemma.

Lemma 3 *Let s be any node of G and let u be any outneighbor of s. Then there exists another inneighbor of s, say v, such that, by splitting (s, u) and (v, s),* i.e., *removing the arcs (s, u) and (v, s) and adding the arc (v, u), we obtain a multigraph G' satisfying the following conditions:*

1. $p_{G'}(i,j) = p_G(i,j)$ *for all* $i,j \in V \setminus \{s\}$ *and*

2. $p_{G'}(s,j) = \min(p_G(s,j), d_G^+(s) - 1)$.

Proof.   Our proof for Lemma 3 is very similar to the Goemans and Bertsimas' proof for Lemma 1. This latter is also in major part inspired from the one of Lovasz for Lemma 1 (Condition 1) which proceeds along the following lines.

1. There exists at most one set S satisfying:

   (a1) $s \in S, u \notin S$,

   (b1) $|\delta_G(S)| = c_G(i,j)$ for some $i,j \in V$, $i \in S$, $j \notin S$ and $i \neq s$ and

   (c1) S is minimal with respect to the above two conditions.

2. If there is no such S, then any neighbor v of s can be used for the splitting operation.

3. If such a S exists then there exists at least one neighbor of s in S. Moreover, any neighbor of s in S can be used for the splitting operation.

We will show that this proof procedure can be used for the directed case. Let us remove the orientation on the arcs of G, then G becomes an undirected Eulerian multigraph. The following remark expresses the relation between $c_G(i,j)$, which is the number of edge-disjoint paths between i and j in undirected G, and $p_G(i,j)$, which is the number of arc-disjoint paths from i to j (or $p_G(j,i)$ for the number of arc-disjoint paths from j to i) in the original directed graph G.

Remark 1 *Let $\delta_G(S)$ be the cut associated to S in the undirected G. We have*

- $|\delta_G(S)| = 2|\delta_G^+(S)| = 2|\delta_G^-(S)|$ *for all* $S \subset V$.

- $c_G(i,j) = 2p_G(i,j) = 2p_G(j,i)$.

Proof.   We have $\delta_G(S) = \delta_G^+(S) \cup \delta_G^-(S)$ and as $\delta_G^+(S) \cap \delta_G^-(S) = \emptyset$, $|\delta_G(S)| = |\delta_G^+(S)| + |\delta_G^-(S)|$. Since G is Eulerian, we have $|\delta_G^+(S)| = |\delta_G^-(S)|$ for all $S \subset V$. Hence, $|\delta_G(S)| = 2|\delta_G^+(S)| = 2|\delta_G^-(S)|$. For all pair i, j in V with $i \neq j$, as $p_G(i,j) = \min\{|\delta^+(T)| : i \in T \text{ and } j \in V \setminus T\}$, $p_G(j,i) = \min\{|\delta^-(T)| : i \in T \text{ and } j \in V \setminus T\}$ and $|\delta_G^+(T)| = |\delta_G^-(T)|$ for all $T \subset V$, we have $p_G(i,j) = p_G(j,i)$ and $c_G(i,j) = 2p_G(i,j) = 2p_G(j,i)$.

Hence, given any $s \in V$, applying the Lovasz's procedure for undirected G, we have

- either there exists exactly a S satisfying **(a1), (b1)** and **(c1)**. Reconsider S in the original directed G, we prove that S satisfies

   (a2) $s \in S, u \notin S$,

   (b2) $|\delta_G^+(S)| = p_G(i,j)$ for some $i,j \in V$, $i \in S$, $j \notin S$ and $i \neq s$ and

   (c2) S is minimal with respect to the above two conditions.

Indeed, by Remark 1, if

$$\delta_G(S) = c_G(i,j) = \min\{|\delta(T)| : \ i \in T \text{ and } j \in V \setminus T\}$$

then

$$\delta_G^+(S) = p_G(i,j) = \min\{|\delta^+(T)| : \ i \in T \text{ and } j \in V \setminus T\}.$$

The conditions **(a1)** and **(a2)** (or **(c1)** and **(c2)**, respectively) are the same, independent from the fact that G is directed or not.

- otherwise, *i.e.*, no S satisfying **(a1), (b1)** and **(c1)** exists. Then for all $S \subset V$ such that $s \in S$ and $u \notin S$, we have $|\delta_G(S)| > c_G(i,j)$ for all $i,j \in V$, $i \in S$, $j \notin S$ and $i \neq s$. By $|\delta_G(S)| = 2|\delta_G^+(S)| = 2|\delta_G^-(S)|$, we have also in the original directed graph G, $|\delta_G^+(S)| > p_G(i,j)$ for all $i,j \in V$, $i \in S$, $j \notin S$ and $i \neq s$. Hence, we can select any inneighbor v of s and split (s,u) and (v,s) without changing the value of $p_G(i,j)$ for all $i,j \in V \setminus \{s\}$.

Hence we can summarize below the directed version of Lovasz's proof procedure.

1. There exists at most one set S satisfying **(a2), (b2)** and **(c2)**.

2. If there is no such S, then any inneighbor v of s can be used for the splitting operation.

3. If such a S exists, then there exists at least one inneighbor of s in S. Moreover, any inneighbor of s in S can be used for the splitting operation.

To show Lemma 3, we add a new node $\hat{s}$ to G and $d_G^+(s) - 1$ arcs from $\hat{s}$ to s and $d_G^-(s) - 1$ arcs from s to $\hat{s}$. Let $\hat{G}$ be this new graph. Clearly, $\hat{G}$ is Eulerian. Moreover,

$$p_{\hat{G}}(i,j) = p_G(i,j) \text{ for all } i,j \in V \setminus \{s\},$$

$$p_{\hat{G}}(\hat{s},j) = \min(p_G(s,j), d_G^+(s) - 1) \text{ for all } j \in V \setminus \{s\}.$$

Now applying the proof procedure to $\hat{G}$, we can see that there are two possible cases:

1. There is no $S \subset V \cup \{\hat{s}\}$ that satisfies **(a2), (b2)** and **(c2)** expressed for $\hat{G}$. Then we can select any inneighbor $v \neq \hat{s}$ of s and do the splitting operation on the arcs (v,s) and (s,u). Let G' be the resulted graph after this splitting operation. Since the splitting operation does not affect the connectivity of all the nodes in $V \cup \{\hat{s}\} \setminus \{s\}$, we have $p_{G'}(\hat{s},j) = p_{\hat{G}}(\hat{s},j)$ for all $j \in V \setminus \{s\}$. But as $d_{G'}^+(s) = d_G^+(s) - 1$,

$$p_{\hat{G}}(\hat{s},j) = \min(p_G(s,j), d_G^+(s) - 1) \text{ for all } j \in V \setminus \{s\} \text{ and}$$

$$p_{G'}(\hat{s},j) = \min(p_{G'}(s,j), d_{G'}^+(s)) \text{ for all } j \in V \setminus \{s\}.$$

As $p_{\hat{G}}(\hat{s},j) = p_{G'}(\hat{s},j)$ then $p_{G'}(s,j)$ can not be strictly smaller than $\min(p_G(s,j), d_G^+(s) - 1)$ and as $p_{G'}(s,j) \leq p_G(s,j)$, by construction, $p_{G'}(s,j)$ can not be strictly greater than $\min(p_G(s,j), d_G^+(s) - 1)$. Hence $p_{G'}(s,j) = \min(p_G(s,j), d_G^+(s) - 1)$.

2. Otherwise, *i.e.*, there exists exactly one set $S \subset V \cup \{\hat{s}\}$ satisfying (**a2**), (**b2**) and (**c2**). Note that the condition (**b2**) expressed for $\hat{G}$ is $|\delta_{\hat{G}}^+(S)| = p_{\hat{G}}(i,j)$ for some $i, j \in V \cap \{\hat{s}\}$, $i \in S$, $j \notin S$ and $i \neq s$.

Proposition 1 *If such a S exists then $\hat{s} \in S$.*

Proof. Indeed if $\hat{s} \notin S$ then $|\delta_{\hat{G}}^+(S)| \geq d_G^+(s)$ since both u and $\hat{s} \notin S$ and $\hat{G}$ is Eulerian. Moreover $|\delta_{\hat{G}}^+(S)|$ would be equal to $p_{\hat{G}}(i, \hat{s})$ for some $i \in S \setminus \{s\}$. This follows by (**b2**) and the fact that $|\delta^+(S \cup \{\hat{s}\})_{\hat{G}}| < |\delta_{\hat{G}}^+(S)|$ (since $s \in S$) which implies that there is no i and j in $V \setminus \{s\}$ with $i \in S$ and $j \notin S$ such that $|\delta_{\hat{G}}^+(S)| = p_{\hat{G}}(i,j)$. This leads to a contradiction since $d_G^+(s) \leq |\delta_{\hat{G}}^+(S)| = p_{\hat{G}}(i, \hat{s}) \leq d_{\hat{G}}^-(\hat{s}) < d_{\bar{G}}^-(s) = d_G^+(s)$.

Proposition 2 *If a S satisfying (a2)-(c2) exists, there must exist some* v $\in S$ *with* $v \neq \hat{s}$ *such that* v *is an inneighbor of* s.

Proof. Indeed, if S does not contain any inneighbor of s other than $\hat{s}$, then $|\delta_{\hat{G}}^-(S) \geq d_G^-(s)$. Hence

$$|\delta_{\hat{G}}^+(S)| = |\delta_{\hat{G}}^-(S)| \geq d_G^-(s) > d_G^-(s) - 1 = d_{\hat{G}}^+(\hat{s}) \geq p_{\hat{G}}(\hat{s}, j)$$

Hence, there would exist some $i \in S$, $s \neq i \neq \hat{s}$ and $j \notin S$ such that $\delta_{\hat{G}}^+(S) = p_{\hat{G}}(i, j)$. This leads to a contradiction to the fact that S is minimal since $S \setminus \{s, \hat{s}\}$ would separate i from j and as

$$|\delta^+(S \setminus \{s, \hat{s}\})_{\hat{G}}| = |\delta_{\hat{G}}^+(S)| + |(S : \{s\})_{\hat{G}}| - |(\{s\} : V \setminus S)_{\hat{G}}|$$

and $|(S : \{s\})_{\hat{G}}| = 0$ as $s \in S$ and $|(\{s\} : V \setminus S)_{\hat{G}}| > 0$, we have

$$|\delta^+(S \setminus \{s, \hat{s}\})_{\hat{G}}| < |\delta_{\hat{G}}^+(S)|$$

Therefore, there exists some $v \neq \hat{s}$ such that by splitting (s, u) and (v, s) we obtain a graph $\hat{G}'$ with $p_{\hat{G}}(i, j) = p_{\hat{G}'}(i, j)$ for all $i, j \in V \cup \{\hat{s}\} \setminus \{s\}$. After removing $\hat{s}$, we obtain a graph G' which can be also obtained from G by splitting off (s, u) and (s, v). G' satisfies

$$p_{G'}(i, j) = p_{\hat{G}'}(i, j) = p_{\hat{G}}(i, j) = p_G(i, j) \text{ for all } i, j \in V \setminus \{s\}$$

and

$$p_{G'}(s, j) = p_{\hat{G}'}(s, j) \geq p_{\hat{G}'}(\hat{s}, j) = p_{\hat{G}}(\hat{s}, j)$$
$$= \min(p_G(s, j), d_G^+(s) - 1) \text{ for all } j \in V \setminus \{s\}.$$

Moreover, due to splitting operation $p_{G'}(s, j) \leq p_G(s, j)$ and $p_{G'}(s, j) \leq d_G^+(s) - 1$.

We finish here the proof of Lemma 3.

Proof of Theorem 1: Let C be the optimal value of (3) and C' be the optimal value of (2). As (3) is a relaxation of (2), we have $C \leq C'$. To show $C \geq C'$ let x be an optimal solution of (3), *i.e.*, $C = \sum_{e \in A} c_e x_e$. We can assume without loss of generality that x is integer since there exists some

integer p such that px is integer and we can replace k by $p.k$ in (3) and (2). Let $G = (V, A)$ be the multigraph which has x_e copies of each arc e such that $x_e > 0$. Applying repeatedly the splitting operation described in Lemma 3 with s chosen among the nodes of V such that $x(\delta^+(s)) > k$ until $d^+_{G'}(i) = k$ for all $i \in V$, we will obtain a multigraph G' such that

- $p_{G'}(i, j) \geq k$ for all $i, j \in V$

- $d^+_{G'}(i) = k$ for all $i \in V$.

Let x' be the point such that for each $e \in A$, x'_e is equal to the number of copies of e in G'. We can see that x' is a solution of (2). Since the costs satisfy the triangle inequality, each time we perform a splitting operation the cost of the solution does not increase, which implies that $C = \sum_{e \in A} c_e x_e \geq \sum_{e \in A} c_e x'_e \geq C'$.

3.2.   *Held-Karp Relaxation for ATSP and the Parsimonious Property*

Let us consider the Asymmetric Traveling Salesman Problem (ATSP) on G. The Held-Karp bound for this problem can be computed by solving the following linear program:

$$\min \sum_{e \in A} c_e x_e$$

subject to

$$
\begin{array}{ll}
x(\delta^+(S)) \geq 1 & \text{for all } S \subset V \text{ such that } |S| \geq 2 \\
x(\delta^+(v)) = 1 & \text{for all } v \in V \\
x(\delta^-(v)) = 1 & \text{for all } v \in V \\
0 \leq x_e \leq 1 & \text{for all } e \in A
\end{array}
$$

This linear program is called *Held-Karp relaxation* for ATSP. In fact the upper bound constraint $x_e \leq 1$ for all $e \in A$ is redundant due to the degree constraints $x(\delta^+(v)) = 1$ and $x(\delta^-(v)) = 1$ for all $v \in V$. Thus, we can rewrite the Held-Karp relaxation as follows:

$$\min \sum_{e \in A} c_e x_e \tag{4}$$

subject to

$$
\begin{array}{ll}
x(\delta^+(S)) \geq 1 & \text{for all } S \subset V \text{ such that } |S| \geq 2 \\
x(\delta^+(v)) = 1 & \text{for all } v \in V \\
x(\delta^-(v)) = 1 & \text{for all } v \in V \\
x_e \geq 0 & \text{for all } e \in A
\end{array}
$$

The following theorem is a direct application of Theorem 1.

Theorem 2 *If the costs* c *satisfy the triangle inequality then the optimum of* (4) *is equal to the optimum of:*

$$\min \sum_{e \in A} c_e x_e \tag{5}$$

subject to

$$x(\delta^+(S)) \geq 1 \qquad \text{for all } \emptyset \neq S \subset V \text{ (included } S \text{ singleton)}$$

$$x(\delta^+(v)) = x(\delta^-(v)) \qquad \text{for all } v \in V$$

$$x_e \geq 0 \qquad \text{for all } e \in A$$

4. Algorithm

We are now ready to state our algorithm for directed tour cover which is heavily inspired from the one for ToCP in [2].

(1) Let x^* be the vector minimizing cx over DToC(G);

(2) Let $U \leftarrow \{v \in V | x^*(\delta^+(\{v\})) \geq \frac{1}{2}\}$;

(3) Let $G_U = (V_U, A_U)$ be a graph with vertex set $V_U = U$ and the arc set A_U is built as follows: for each pair of nodes $i, j \in U$, if there exists a path from i to j in G, create an arc (i, j) in G_U with the cost c_{ij}^U being equal to the cost of the shortest path from i to j in G;

(4) Run the Frieze, Galbiati and Maffioli heuristic [4] to find an approximate minimum traveling salesman directed tour on G_U. Note that the linear program in step **(1)** can be solved in polynomial time by using the ellipsoid method with a minimum cut computation as separation oracle. The algorithm outputs a directed tour which spans U. We can see that U is vertex cover of G. Since for any arc $e = (u, v) \in A$, $x^*(\delta^+(\{u, v\})) \geq 1$ and $x^*(\delta^+(\{u, v\})) = x^*(\delta^+(u)) + x^*(\delta^+(v)) - 2x_e^*$, at least $x^*(\delta^+(u))$ or $x^*(\delta^+(v))$ is greater or equal to $\frac{1}{2}$, *i.e.*, at least u or v should belong to U. Therefore, the algorithm outputs a directed tour cover of G.

5. Performance Guarantee

Let x^* be the vector minimizing $\sum_{e \in A} c_e x_e$ over DToC(G) and $U = \{v \in V \mid x^*(\delta^+(\{v\})) \geq \frac{1}{2}\}$. Let $y^* = 2x^*$, and let k be the minimum integer such that ky^* is integer. Let $\tilde{G} = (V, \tilde{A})$ be the multi-graph that has ky_e^* copies of arc e for each $e \in \tilde{A}$. Note that $\tilde{G}$ is Eulerian. For any $i \neq j \in V$, let $p_{\tilde{G}}(i, j)$ be number of disjoint paths from i to j in $\tilde{G}$.

Remark 2 $p_{\tilde{G}}(u, j) \geq k$ *for all* $u \in U$ *and* $j \in V \setminus \{u\}$.

Proof. To prove the remark, we will show that for any $S \subset V$ such that $u \in S$ and $j \in V \setminus S$, we have $|\delta_{\tilde{G}}^+(S)| \geq k$. Indeed, if $A(S) \neq \emptyset$ then by construction, $x^*(\delta_G^+(S)) \geq \frac{1}{2}$ which implies $y^*(\delta_G^+(S)) \geq 1$ and consequently $|\delta_{\tilde{G}}^+(S)| \geq k$. Otherwise, if $A(S) = \emptyset$ then $x^*(\delta_{\tilde{G}}^+(S)) \geq x^*(\delta^+(u)) \geq \frac{1}{2}$. Equivalently, $y^*(\delta_G^+(S)) \geq y^*(\delta^+(u)) \geq 1$. Hence, we also have $|\delta_{\tilde{G}}^+(S)| \geq k$.

In $\tilde{G}$, for each $i \in V \setminus U$, repeat the splitting operation on i until $d_{\tilde{G}}^+(i) = d_{\tilde{G}}^-(i) = 0$. Consequently, after these operations, all the arcs in $\tilde{G}$ having the end-nodes in U. Let $\tilde{G}_U = (U, \tilde{A}_U)$ be the subgraph of induced by U of $\tilde{G}$.

Remark 3 *After these splitting operations,* $p_{\tilde{G}_U}(u, v) \geq k$ *for all* $u \in U$ *and* $v \in U \setminus \{u\}$.

Consider the graph $G_U = (U, A_U)$ that is defined in Step **(2)** of the algorithm. Note that all the arcs in $\tilde{A}_U$ (with eventually multiple copies) are in A_U but there may exist arcs in A_U that are not in $\tilde{A}_U$. Let x^U be the vector in $\mathbb{R}^{|A_U|}$ such that for each $e \in A_U$, $x_e^U = \frac{p_e}{k}$, where p_e is the number of copies of e in $\tilde{A}_U$ ($p_e = 0$ if $e \notin \tilde{A}_U$) .

Lemma 4 *We have $2cx^* = cy^* \geq c^U x^U$.*

Proof. We have seen that kx^U is obtained from ky^* by repeated splitting operations on the vertices in $V \setminus U$. By definition of c^U which takes the shortest path cost, we have $c^U(kx^U) \leq c(ky^*)$ which implies $c^U x^U \leq cy^* = 2cx^*$.

Let z^* be an optimal solution of Held-Karp relaxation over G_U.

Lemma 5 $c^U x^U \geq c^U z^*$.

Proof. We can see that x^U satisfies the following linear inequalities

$$x(\delta^+_{G_U}(u)) = x(\delta^-_{G_U}(u))$$

$$x(\delta^+_{G_U}(S)) \geq 1 \text{ for all } \emptyset \subset S \subset U$$

which are the inequalities of (5) applied to G_U. By Theorem 2, z^* minimizes the cost function $c^U x$ over (5). Hence we have $c^U x^U \geq c^U z^*$.

We are ready now to state the main theorem

Theorem 3 *The algorithm outputs a directed tour cover of cost no more than $2\log_2(n)$ times the cost of the minimum directed tour cover.*

Proof. The algorithm return, as a solution of DToCP, the traveling salesman directed tour output T by Frieze *et al.*'s heuristic applied on G_U. Let x^T be its incidence vector. The cost of T is then equal to $c^U x^T$. Williamson [5] proved that the cost of T is not more than $\log_2(n)$ times the cost of the optimal solution of Held-Karp relaxation over G_U, *i.e.*, $c^U x^T \leq \log_2(n) c^U z^*$. By Lemmas 5 and 4, we have $c^U x^T \leq 2\log_2(n) cx^*$.

6. Conclusions

In a recent paper [9], Asadpour *et al.* showed that one can find an asymmetric traveling salesman tour of cost at most $O(\log n / \log\log n)$ times the cost of the optimal solution of the Held-Karp. Hence we can use their algorithm in Step 4 of our algorithm in place of the Frieze *et al.* heuristic to obtain a ratio of $O(\log n / \log\log n)$ for DToCP. In fact, Asadpour *et al.* gave a $(2 + 8\log n / \log\log n)$-approximate solution for ATSP. This ratio is asymptotically better but worse for realistic value of n ($n \leq 10^{20}$) than $\log_2(n)$, which is the ratio achieved by the Frieze *et al.* heuristic. An obvious extension is to design a combinatorial approximation algorithm for DToCP. In [1], the authors built combinatorially an undirected tour cover by duplicating some trees over a vertex cover. Since duplicating a branching does not give a directed tour, this idea can not be applied for the directed case. Thus a combinatorial approximation algorithm for DToCP remains wide open. Another possible extension is to use the parsimonious property for Eulerian digraphs in Section 3 to design approximation algorithms for other problems which involve a tour in directed graphs. In another paper [10], we have successfully applied this idea to design an approximation algorithm for the prize collecting asymmetric traveling salesman problem.

References

1. Arkin, E.M.; Halldórsson, M.M.; Hassin R. Approximating the tree and tour covers of a graph. *Inf. Pro. Lett.* **1993**, *47*, 275–282.
2. Könemann, J.; Konjevod, G.; Parekh, O.; Sinha, A. Improved Approximations for Tour and Tree Covers. *Algorithmica* **2003** , *38*, 441–449.
3. Kaplan, H.; Lewenstein, M.; Shafrir, N.; Sviridenko, M. Approximation algorithms for asymmetric TSP by decomposing directed regular multigraphs. *J. ACM* **2005**, *52*, 602–626.
4. Frieze, A.M.; Galbiati, G.; Maffioli, F. On the worst case performance of some algorithms for the asymmetric traveling salesman problem. *Networks* **1982**, *12*, 23–39.
5. Williamson, D.P. Analysis of the Held-Karp heuristic for the traveling salesman problem. M.D. Thesis, Massachusettes Institute of Technology, Cambridge, MA, USA, 1990.
6. Goemans, M.X.; Bertsimas, D.J. On the parsimonious property of connectivity problems. In *Proceeding of the First Annual ACM-SIAM Symposimum on Discrete Algorithm*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 1990; pp. 388–396.
7. Lovasz, L. On some connectivity properties of Eulerian graphs. *Acta Mathematica Hungarica* **1976**, *28*, 129–138.
8. Jackson, B. Some remarks on arc-connectivity, vertex splitting, and orientation in graphs and digraphs. *J. Graph Theory* **1988**, *12*, 429–436.
9. Asadpour, A.; Goemans, M.X.; Madry, A.; Oveis Gharan, S.; Saberi, A. An O(log *n*/log log *n*)–approximation algorithm for the asymmetric traveling salesman problem. In *Procceding of 21st ACM-SIAM Symposium on Discrete Algorithms*, 2010.
10. Nguyen, V.H.; Nguyen, T.T.T. Approximating the asymmetric profitable tour. *Electron. Notes Discrete Math.* **2010**, *36*, 907–914.