Separation and Lifting of TSP inequalities*

Robert Carr†

GSIA

Carnegie Mellon University, Pgh. PA

Abstract

The problem of *separating* a class of TSP inequalities consists in devising a method that, given a fractional point x^* for a relaxation of the TSP, either finds an inequality in the given class of TSP inequalities that is violated by x^* or determines that there are no such violated inequalities. Both *heuristic* and *exact* efficient separation algorithms are sought after so that we can more easily solve TSP instances when using a cutting plane approach such as *branch-and-cut*.

We present efficient methods for exactly separating over very large classes of TSP inequalities. These inequality classes arise from *node lifting* familiar TSP inequalities.

Key words: traveling salesman problem,separation,lifting,polytope, inequalities.

1 Introduction

Given a complete graph $K_n = (V, E)$ with costs on the edges, the *traveling salesman problem* (TSP) consists in finding a minimum cost Hamilton cycle in K_n. This problem is also called the *symmetric traveling salesman problem* to distinguish it from the *asymmetric traveling salesman problem* which is formulated on a directed graph. It is known to be NP-hard, even when the

*This research was partially supported by NSF grant DMS9509581.

†E-mail: rc1a@andrew.cmu.edu

costs satisfy the triangle inequality, i.e. when $c_{ij} + c_{jk} \geq c_{ik}$ for all $i, j, k \in V$ (see [8]).

For any edge set $F \subseteq E$ and $x \in \mathbf{R}^E$, let $x(F)$ denote the sum $\sum_{e \in F} x_e$. For any node set $W \subset V$, let $\delta(W)$ denote $\{uv \in E \,|\, u \in W,\, v \notin W\}$. The usual integer programming (IP) formulation for the TSP is as follows:

$$
\begin{aligned}
&\text{minimize } cx && && (1)\\
\text{subject to } x(\delta(v)) &= 2 && \text{for all } v \in V, && (2)\\
x(\delta(S)) &\geq 2 && \text{for all } S \subset V, && (3)\\
& && 3 \leq |S| \leq n-3, &&\\
0 \leq x_e &\leq 1 && \text{for all } e \in E, && (4)\\
x & \text{ integer.} && && (5)
\end{aligned}
$$

The vector x here is interpreted as an *incidence vector* of a Hamilton cycle, i.e. $x_e = 1$ for each $e \in E$ that is in the Hamilton cycle, and $x_e = 0$ for all other edges. Constraints (2) are called the *degree constraints*, and constraints (3) are called the *subtour elimination constraints*.

In an attempt to solve a given instance of the TSP, one may solve a linear programming relaxation of the TSP for this instance. For instance, (1) without constraints (5) is a natural linear programming relaxation of the TSP called the *subtour relaxation*. If such a relaxation yields an integral solution, then this solution is the incidence vector of a minimum cost Hamilton cycle for this instance. In trying to acheive such an integral optimal solution for this relaxation, one can use some of the valid inequalities of the TSP, including but not limited to those in (1), as constraints of the relaxation. Usually, this attempt will fail at first, yielding a fractional optimal solution x^* for this relaxation.

However, if one can find a valid inequality of the TSP that cuts off x^*, then one could add this inequality as a constraint in this linear programming relaxation, and resolve this relaxation. The task of finding such an inequality efficiently generally goes by the name of the *separation problem*. This inequality which can now be added to the linear programming relaxation is called a *cutting plane*.

The following Cutting plane algorithm could in theory be used to solve the traveling salesman problem, if one could solve the separation problem.

(i) Let an initial LP relaxation of the TSP be the current LP relaxation.

(ii) Find the optimal solution x^* to the current linear programming relaxation of the TSP. If x^* is the incidence vector of a Hamilton cycle, STOP.

(iii) Solve the separation problem, obtaining a valid TSP inequality $ax \geq b$ such that $ax^* < b$.

(iv) Add the cutting plane $ax \geq b$ to the current LP relaxation and make the resulting new LP relaxation the current LP relaxation. Go to step (ii).

However, in practice, the more practical *Branch and Cut* algorithm is employed for solving instances of the TSP; see [14].

A valid inequality is said to be *facet-defining* if it is not a valid equation and it also can not be expressed as a linear combination of valid inequalities and equations, where the multipliers for the inequalities are all non-negative. Prevailing opinion has it that the best inequalities to use in a Branch and Cut or a Cutting plane algorithm are the facet-defining ones. Many classes of facet-defining and valid inequalities of the TSP have been found; see [9].

A separation method that has a guarantee of finding a violated inequality in a class of inequalities or showing that none of the inequalities in this class are violated by x^* is called an *exact* separation method. Both heuristic and exact separation methods have been developed for some of these classes of inequalities of the TSP. Examples of heuristic separation methods can be found in [15], and more recently in [11] and [1]. We wish for both the exact and heuristic separation methods to be able to be carried out in a polynomial amount of time in terms of the number n of vertices in our TSP instance.

Until recently, only two classes of facet-defining inequalities could be separated exactly in polynomial time. The subtour elimination inequalities have an exact separation algorithm based on finding a minimum cut in a weighted graph with weights given by x_e^* for all $e \in E$. The other class of inequalities which could be separated exactly were the *2-matching constraints*, using the algorithm developed by Padberg and Rao (1982); see [13]. More recently, we have learned how to separate *bipartition inequalities* (which the *comb inequalities* and *clique tree inequalities* are subclasses of) with a fixed number of *handles* and *teeth*; see [2]. A little progress has also been made on separating *comb inequalities* regardless of the number of teeth; see [6].

In this paper, we develop an easy exact method for separating all classes of inequalities which are defined using a lifting procedure called *node lifting*, introduced in [12]. Node lifting is a generalization of *0 node-lifting*, see [12]. Zero node-lifting will be defined in the next section. If additional variables are allowed, such an exact separation method was already accomplished by the author in [3] and [4], in which a linear programming relaxation called *cycle-shrink*, which has the usual variables plus additional cycle-shrink variables, was used.

2 0 Node-Lifting

Two valid inequalities for a polyhedron are said to be *equivalent* if one can obtain one of these inequalities by adding a non-negative multiple of the other inequality and a linear combination of equations of the polyhedron together. For the TSP, the equations are the degree constraints. When we talk about TSP inequalities, we tend to think of equivalent inequalities as different forms expressing the same inequality. For instance, given vertex sets $H, T_1, T_2, T_3 \subset V$ satisfying certain properties, we can define a *three-tooth comb inequality*. But, because of the degree constraints, there is a class of equivalent inequalities which we would describe as all being forms of a given individual three-tooth comb inequality.

We now define the class of three-tooth comb inequalities and illustrate two different forms in which any individual three-tooth comb inequality can be expressed. A *three-tooth comb* consists of a *handle* $H \subset V$ and three *teeth* $T_i \subset V$ for $i = 1, 2, 3$ satisfying the following:

(i) The three teeth are pairwise disjoint.

(ii) The handle intersects all three teeth.

(iii) None of the teeth are included in the handle.

We show a three-tooth comb in Figure 1.

Traditionally, a three-tooth comb inequality with handle H and teeth T_i, $i = 1, 2, 3$ is written in *closed form* as follows:

$$x(E(H)) + \sum_{j=1}^{3} x(E(T_j)) \leq (|H| - 1) + \sum_{j=1}^{3}(|T_j| - 1) - 1 \tag{6}$$

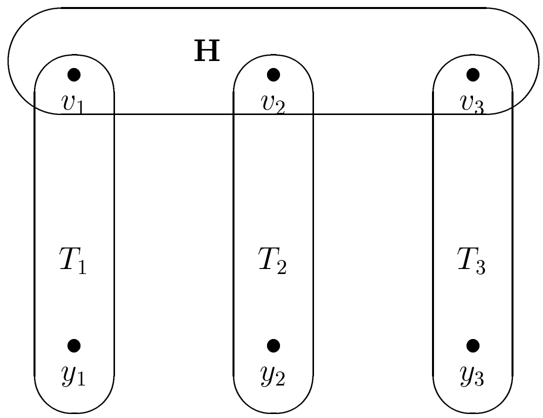

Figure 1: a three-tooth comb

Of course, there may be more than one term involving a common variable x_e, which means one must combine these like terms to obtain the final expression for this inequality.

An equivalent form of this three-tooth comb is as follows:

$$x(\delta(H)) + \sum_{j=1}^{3} x(\delta(T_j)) \geq 10 \tag{7}$$

Denote the coefficient for variable x_e after like terms are collected by α_e. One interesting thing about this form is that these coefficients satisfy the triangle inequality; i.e. for any three distinct vertices $i, j, k \in V$, we have $\alpha_{ik} \leq \alpha_{ij} + \alpha_{kj}$. However, if one subtracts any multiple of the degree constraint $x(\delta(i)) = 2$ from this inequality, then these coefficients no longer satisfy the triangle inequality, regardless of which vertex $i \in V$ is used. When the coefficients α_e satisfy these two properties, the inequality is said to be in *tight triangular form.*

Any TSP inequality can be put uniquely, up to a constant multiple, into tight triangular form, which was introduced by Naddef and Rinaldi; see [12]. An inequality $\alpha x \geq \alpha_0$ is in tight triangular (TT) form if and only if:

i) The coefficients of α satisfy the triangle inequality.

ii) For all $w \in V$, there exists $u, v \in V$ such that $\alpha_{uv} = \alpha_{uw} + \alpha_{vw}$.

Consider a facet-defining inequality $\alpha x \geq \alpha_0$ in TT form. If none of the left hand side coefficients are 0, then $\alpha x \geq \alpha_0$ is said to be a *simple inequality.* Defining $H := \{v_1, v_2, v_3\}$ and $T_i := \{v_i, y_i\}$ for $i = 1, 2, 3$ (see Figure 1) yields a simple three-tooth comb inequality. But if $T_1 = \{v_1, y_1, j\}$ for some vertex $j \in V$ that wasn't in the original comb, but the handle and the other teeth remain the same, then the resulting comb inequality would not be simple, but would be a *0 node-lifting* of the previous inequality, where the vertex j is a 0 node-lifting of vertex y_1. More formally, let $hx \geq h_0$ be a facet-defining TSP inequality in TT form on the complete graph $K_n = (V_{(n)}, E_{(n)})$ of n vertices. Add k more vertices to $V = V_{(n)}$, obtaining the vertex set $V_{(n+k)}$. We 0 node-lift node u to obtain the inequality $h^* x^* \geq h_0$, where:

i) $h^*_e = h_e$ for all $e \in E_{(n)}$

ii) $h^*_{ij} = h_{uj} \quad \forall i \in V_{(n+k)} \setminus V_{(n)} \quad \forall j \in V_{(n)} \setminus \{u\}$

iii) $h^*_{ij} = 0 \quad \forall i, j \in (V_{(n+k)} \setminus V_{(n)}) \cup \{u\}$

We call the vertices in $V_{(n+k)} \setminus V_{(n)}$ *copies* of u. It has been proven that every facet-defining inequality is derivable from a simple inequality through 0 node-lifting; see [12]. It would be nice if the 0 node-liftings of any simple facet-defining inequality were always facet-defining. For the simple inequalities examined so far, this has been observed to be the case. In [12], sufficient conditions for this to hold are given.

We now show that the concept of 0 node-lifting can be used to define classes of inequalities. Consider the subcollection of simple inequalities defined on complete underlying graphs having possibly different vertex sets, but with each of these vertex sets being a subset of $V := \{1, \ldots, n\}$. We partition this subcollection into classes of inequalities in the following natural way. Call two simple inequalities *isomorphic* if a bijective mapping of vertices transforms one of the inequalities into the other one. That is, if $\alpha x \geq \alpha_0$ is defined on a complete underlying graph $H = (W, E)$ and $\alpha' x \geq \alpha_0$ is defined on a complete underlying graph $H' = (W', E')$, then these two inequalities are isomorphic if there exists a bijective mapping $\pi : W' \to W$ such that

$$\alpha_{\pi(i)\pi(j)} = \alpha'_{ij} \qquad \text{for all } (i,j) \in E' \tag{8}$$

and hence

$$\sum_{(i,j)\in E'} \alpha_{\pi(i)\pi(j)} x_{\pi(i)\pi(j)} = \sum_{(i,j)\in E'} \alpha'_{ij} x_{ij} \tag{9}$$

This partitions the simple inequalities into isomorphism equivalence classes.

For example, suppose one has the following simple comb inequality on an underlying graph $H = (W, E)$, where $W := \{1, 2, 3, 4, 5, 6\}$:

$$x(\delta\{1,2,3\}) + x(\delta\{1,4\}) + x(\delta\{2,5\} + x(\delta\{3,6\}) \geq 10 \tag{10}$$

Suppose one also has the simple comb inequality on an underlying graph $H' = (W', E')$, where $W' := \{1, 3, 5, 7, 9, 11\}$:

$$x(\delta\{1,3,7\}) + x(\delta\{1,5\}) + x(\delta\{3,9\} + x(\delta\{7,11\}) \geq 10 \tag{11}$$

The mapping $\pi : W' \to W$ defined by

$$\begin{array}{lll} \pi(1) = 1 & \pi(3) = 2 & \pi(7) = 3 \\ \pi(5) = 4 & \pi(9) = 5 & \pi(11) = 6 \end{array}$$

shows that these inequalities are isomorphic. That is, replacing 3 in (11) with 2, and 7 in (11) with 3, etc., makes inequality (11) identical to inequality (10). Hence, one of these equivalence classes would be the class of simple three-tooth comb inequalities defined on a 6 node graph.

One can describe a particular simple inequality in this equivalence class by giving just an ordered listing of vertices in the inequality, where the ordering has a meaning specific to the class of inequalities to which our particular inequality belongs. For instance, one could describe a particular simple 6 node three-tooth comb by giving first the vertex in the intersection of the handle and tooth 1, then the vertex in the intersection of the handle and tooth 2, and so on. So, the ordered listing of vertices for (11) would be $(1, 3, 7, 5, 9, 11)$. Of course, we may permute the teeth, so $(1, 7, 3, 5, 11, 9)$ would also describe (11), but this should not bother us. The meaning of the ordering in such a listing of vertices will also clearly change depending on which class of simple inequalities the inequality we are describing belongs to. We shall refer to such an ordered listing as a *backbone* of the particular inequality. We shall refer to the set of the vertices in the backbone as the *backbone set*.

Now, let $fx \geq f_0$ be a facet-defining TT inequality. As stated before, it is then derivable from some simple inequality $hx \geq f_0$ through zero node-lifting.

Call the equivalence class of simple inequalities to which $hx \geq f_0$ belongs S. We then extend the inequalities in the above partition to include all the

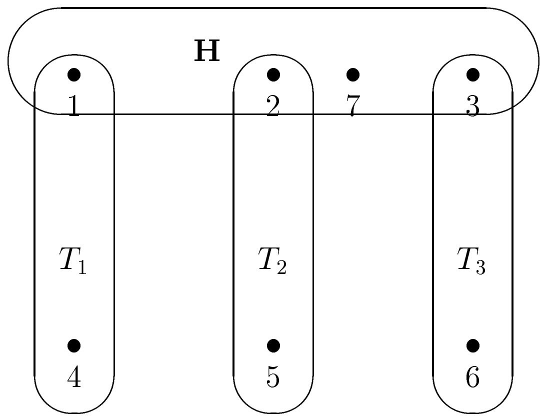

Figure 2: a simple 7 node three-tooth comb

zero node-liftings of $hx \geq f_0$ and its isomorphisms so that $fx \geq f_0$ is put into the equivalence class S as well. Moreover, we extend the definition of a backbone so that a backbone for $hx \geq f_0$ is now also a backbone for $fx \geq f_0$.

For example, consider the comb inequality:

$$x(\delta\{1,1',2,2',2'',3\})+x(\delta\{1,1',4\})+x(\delta\{2,2',2'',5\})+x(\delta\{3,6\}) \geq 10 \quad (12)$$

Here, $1'$ is a zero node-lifting of node 1, and $2'$ and $2''$ are zero node-liftings of node 2. Hence, inequality (12) belongs in the same equivalence class of inequalities as inequality (10). So, if S_6 were the class of all simple three-tooth comb inequalities defined on a 6 node graph, than S_6 would be extended to include a large subclass of the three-tooth comb inequalities. This new class would still not be the entire class of three-tooth comb inequalities because it would be missing the following simple three-tooth comb inequalities.

7 node simple three-tooth comb inequality: The simple three-tooth comb inequality on an underlying graph $H = (W, E)$, where $W := \{1, 2, 3, 4, 5, 6, 7\}$, defined by:

$$x(\delta\{1,2,3,7\}) + x(\delta\{1,4\}) + x(\delta\{2,5\} + x(\delta\{3,6\}) \geq 10 \qquad (13)$$

Denote the set of all isomorphisms of such an inequality by S_7. The corresponding 7 node three-tooth comb is shown in Figure 2.

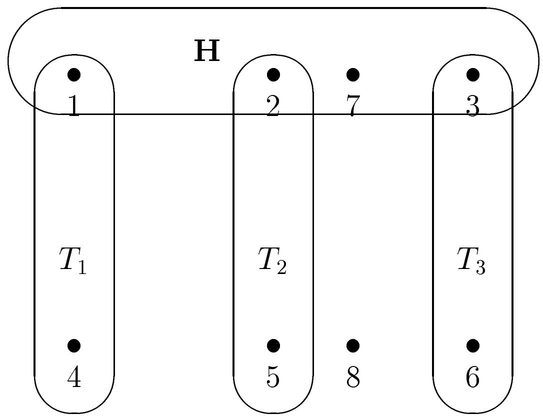

Figure 3: a simple 8 node three-tooth comb

8 node three-tooth comb inequality: The same inequality as that in S_7, but defined on an underlying graph $H = (W, E)$, where $W := \{1, 2, 3, 4, 5, 6, 7, 8\}$. Denote the set of all isomorphisms of such an inequality by S_8. The corresponding 8 node three-tooth comb is shown in Figure 3.

Note that the inequality obtained from an inequality in S_6 by simply adding a node to the underlying graph is isomorphic to an inequality in S_7 since the handle can be complemented to include this added node.

The class $S_{\text{all}} = S_6 \cup S_7 \cup S_8$ is the class of all simple three-tooth comb inequalities. When one takes all the 0 node-liftings of inequalities in S_{all}, then one finally obtains the class C_3 of three-tooth combs. In the next section, we show that by also adding the lifting procedure called *1-node lifting*, we can obtain the class C_3 by lifting inequalities in S_6.

3 1-Node and Triangular Lifting

Let $cx \geq c_0$ be a simple inequality in TT form defined on the underlying graph $K_n = (V, E)$ whose vertex set V is $\{1, \ldots, n\}$. Suppose we wish to add node $n + 1$ to our underlying graph and this inequality. Consider choosing numbers λ_i for $i = 1, \ldots, n$ such that the following constraints are satisfied:

$$\begin{array}{rl} \lambda_i + \lambda_j \geq c_{ij} & \forall ij \in E \\ \lambda_i \geq 0 & \forall i \in V \end{array} \tag{14}$$

Define $\overline{c}$ by

$$\overline{c}_{ij} = \begin{cases} c_{ij} & \text{for } i, j \neq n+1 \\ \lambda_i & \text{for } j = n+1 \end{cases} \tag{15}$$

We then have the following theorem, which was proved by Naddef and Rinaldi in [12].

Theorem 1 *$\overline{c}x \geq c_0$ is a valid TSP inequality on K_{n+1}.*

Proof: Let a Hamilton cycle $\overline{H}$ in K_{n+1} be given. Denote the incidence vector of $\overline{H}$ by $\overline{x}$. Denote the neighbors of node $n+1$ in $\overline{H}$ by i and j. Construct a Hamilton cycle H in K_n by removing node $n+1$ from $\overline{H}$ and adding edge ij. Denote the incidence vector of H by $\hat{x}$. Since $cx \geq c_0$ is a valid TSP inequality, we have:

$$c\hat{x} \geq c_0.$$

But, we have:

$$\overline{c} \cdot \overline{x} = c\hat{x} + \lambda_i + \lambda_j - c_{ij} \geq c\hat{x},$$

because of the constraints in (14) which the numbers λ_i must satisfy. Combining these inequalities, we get

$$\overline{c} \cdot \overline{x} \geq c_0,$$

as required. □

Consider λ defined by:

$$\lambda_i = \begin{cases} c_{ik} & \text{for } i \neq k \\ 0 & \text{for } i = k \end{cases} \tag{16}$$

Then, with $\overline{c}$ defined by (15), the resulting inequality $\overline{c}x \geq c_o$ is a 0 node-lifting of $cx \geq c_0$, with node $n+1$ being a copy of node k. However, we shall see that we obtain more than just the 0 node-lifted inequalities with this lifting procedure. Naddef and Rinaldi in [12] called this lifting procedure *1-node lifting* whether or not it coincides with 0 node-lifting.

We are only interested in those vectors λ that are extreme points of the polyhedron P defined by the constraints in (14), as the following theorem shows:

Theorem 2 *If the vector λ is not an extreme point of P then, with $\overline{c}$ defined by (15), the inequality $\overline{c}x \geq c_0$ is not facet-defining for the TSP on K_{n+1}.*

Proof: Suppose λ is not an extreme point of P. Then, we can express λ by

$$\lambda = \frac{1}{2}\lambda^1 + \frac{1}{2}\lambda^2,$$

where $\lambda^1, \lambda^2 \in P \setminus \{\lambda\}$. For $i = 1, 2$, define $\overline{c}^i$ using λ^i in (15). Define $\overline{c}$ using λ in (15). Then, it follows that:

$$\overline{c} = \frac{1}{2}\overline{c}^1 + \frac{1}{2}\overline{c}^2$$

But $\overline{c}^1 x \geq c_0$ and $\overline{c}^2 x \geq c_0$. Therefore, $\overline{c}x \geq c_0$ is not facet-defining for the TSP on K_{n+1}. □

As before, let $cx \geq c_0$ be a simple inequality in TT form defined on the underlying graph $K_n = (V, E)$ whose vertex set V is $\{1, \ldots, n\}$. Suppose this time we wish to add nodes $n+1$ through $n+k$ to our underlying graph and this inequality. The underlying complete graph is now $K_{n+k} = (V_{n+k}, E_{n+k})$. Consider choosing coefficients $\overline{c}_{ij}$ subject to the following constraints:

$$\begin{aligned} \overline{c}_{ij} \leq \overline{c}_{ik} + \overline{c}_{jk} \qquad & \forall i \neq j \neq k \in V_{n+k} \\ \overline{c}_{ij} = c_{ij} \qquad & \forall ij \in E \end{aligned} \tag{17}$$

Naddef and Rinaldi noted in [12] that $\overline{c}x \geq c_0$ is a valid TSP inequality on K_{n+k}. In fact, oftentimes one can increase the right hand side by some amount and still have a valid inequality. They called this lifting procedure as *node lifting*. For completeness, we give a proof of this here.

Theorem 3 *$\overline{c}x \geq c_0$ is a valid TSP inequality on K_{n+k}.*

Proof: Let a Hamilton cycle H^{n+k} in K_{n+k} be given. Denote the incidence vector of H^{n+k} by x^{n+k}. Do the following procedure for each $l = n+k, n+k-1, \ldots, n+1$, starting with $l = n+k$:

(i) Denote the neighbors of l in the Hamilton cycle H^l in the graph K_l by i_l and j_l.

(ii) Construct the Hamilton cycle H^{l-1} in the graph K_{l-1} by removing node l from H^l and adding edge $i_l j_l$. Denote the incidence vector of H^{l-1} by x^{l-1}.

(iii) Decrease l by 1 and if $l \geq n+1$, go back to (i).

Consider the vectors x^l for each $l = n, \ldots, n+k-1$ to have a component for each edge $e \in K_{n+k}$ by setting $x^l_e = 0$ when it is otherwise undefined.

Let $l \in \{n+1, \ldots, n+k\}$ be given. We have that

$$\overline{c}x^l = \overline{c}x^{l-1} + (\overline{c}_{i_l l} + \overline{c}_{j_l l} - \overline{c}_{i_l j_l}).$$

Thus, by constraints (17), it follows that

$$\overline{c}x^l \geq \overline{c}x^{l-1}.$$

Taking the corresponding result for each $l \in \{n+1, \ldots, n+k\}$, and combining them yields

$$\overline{c}x^{n+k} \geq \overline{c}x^n.$$

Since $cx \geq c_0$ is a valid TSP inequality on K_n, and $\overline{c}$ is constrained to coincide with c on this graph, we have that

$$\overline{c}x^n = cx^n \geq c_0.$$

Combining these inequalities, we get

$$\overline{c}x^n \geq c_0,$$

as required. □

Once again, we are only interested in those vectors $\overline{c}$ that are extreme points of the polyhedron P defined by the constraints in (17), as the following theorem shows:

Theorem 4 *If the vector $\overline{c}$ is not an extreme point of P, then the inequality $\overline{c}x \geq c_0$ is not facet-defining for the TSP on K_{n+k}.*

Proof: This follows from the same argument as in Theorem 2. □

Whenever the right hand side can not be increased, call the inequality $\overline{c}x \geq c_0$ on K_{n+k} that we obtained from $cx \geq c_0$ on K_n the *triangular lifting* of $cx \geq c_0$. Hence, triangular lifting could be 0 node-lifting or more generally 1-node lifting. It may also yield something distinct from these two types of liftings, although we do not yet know of an example of this. In the next section, we show 1-node liftings for various classes of simple inequalities.

4 Various 1-Node and Triangular Liftings

Recall from (10) the simple 6 node comb inequality, i.e. in S_6, on an underlying graph $H = (W, E)$, where $W := \{1, 2, 3, 4, 5, 6\}$:

$$x(\delta\{1,2,3\}) + x(\delta\{1,4\}) + x(\delta\{2,5\} + x(\delta\{3,6\}) \geq 10 \tag{18}$$

Denote this inequality by $cx \geq 10$. When we define $\overline{c}$ by (15) where λ satisfies (14), then by Theorem 1, $\overline{c}x \geq 10$ is a valid TSP inequality. Due to Theorem 2, we need to consider only the extreme points λ in the polyhedron defined by the constraints in (14).

Consider an extreme point λ for (14) defined by:

$$\lambda_i = \begin{cases} 1 & \text{for } i = 1,2,3 \\ 2 & \text{for } i = 4,5,6 \end{cases} \tag{19}$$

The resulting inequality $\overline{c}x \geq 10$ is in fact an inequality in S_7, defined by (13). Indeed, this choice of λ corresponds to placing vertex 7 in the handle of a three-tooth comb as indicated by Figure 2.

Now consider λ defined by:

$$\lambda_i = \begin{cases} 2 & \text{for } i = 1,2,3 \\ 1 & \text{for } i = 4,5,6 \end{cases} \tag{20}$$

The resulting inequality $\overline{c}x \geq 10$ is again an inequality in S_7, but corresponds this time to placing vertex 7 outside of the handle and all the teeth.

We can lift again on either 7 node comb inequality. Suppose we lift on the first of these S_7 inequalities, and use λ defined by:

$$\lambda_i = \begin{cases} 2 & \text{for } i = 1,2,3, \\ 1 & \text{for } i = 4,5,6,7 \end{cases} \tag{21}$$

The resulting inequality $\overline{c}x \geq 10$ is then an S_8 inequality whose corresponding three-tooth comb is shown in Figure 3.

We can also choose for λ one of the extreme points defined by (16). This is a 0 node-lifting, and corresponds to putting vertex $n+1$ in the same place of the three-tooth comb as vertex k.

We called the special case of node lifting where the right hand side remained the same as triangular lifting. The process of starting with a TSP inequality in TT form and repeatedly choosing λ each time we wish to add a node to our inequality is *sequential 1-node lifting*. We also use the term *sequential triangular lifting* since Naddef and Rinaldi proved in [12] that the right hand side does not increase when the resulting inequality is facet-defining. We can see that by performing sequential 1-node lifting on the three-tooth comb inequalities in S_6, we can obtain all three-tooth comb inequalities (C_3). Call the polyhedron from which we choose λ when performing sequential 1-node lifting the *sequential lifting polyhedron*.

There is also the triangular lifting which produces the inequality $\overline{c}x \geq c_0$ on K_{n+k} from $cx \geq c_0$ via the constraints in (17). Call this kind of lifting *simultaneous triangular lifting*. Call the polyhedron defined by (17) the *simultaneous lifting polyhedron*.

For all the cases involving the three-tooth comb inequality examined so far, 1-node lifting has a geometric interpretation of putting the new node in a particular place in the three-tooth comb. We will show that all 1-node liftings of a three-tooth comb inequality have this geometric interpretation, and give sufficient conditions for when all the 1-node liftings of a TT inequality have a similar geometric interpretation. As a result, we see easily that (19),(20),(21), and (16) are the only kinds of 1-node liftings for the three-tooth comb.

Call a TT inequality $cx \geq c_0$ *cut-based* if we have

$$cx = \sum_i \alpha_i x(\delta(H_i)). \tag{22}$$

The three-tooth comb inequality is thus a cut-based inequality. In the cut-based representation of an inequality, we may complement any shore H_i. For our subsequent convenience, we complement the appropriate shores so that there are no shores H_i and H_j in the cut-based representation such that $H_i \cup H_j = V$. Call such a cut-based representation a *proper representation*.

By choosing 4 of the shores H_i in (22) and possibly complementing some of these 4, we may produce a *claw* structure, as shown in Figure 4.

One can never get a claw by complementing the shores H_i of the usual cut-based representation of any bipartition inequality, and in particular any comb or clique-tree inequality. If there exists a cut-based representation where this is the case, we say that the given inequality *has no claws*. We are now ready to prove our theorem.

Theorem 5 *The only 1-node liftings of a cut-based TT inequality $cx \geq c_0$ are those having a geometric interpretation of putting the new node in the*

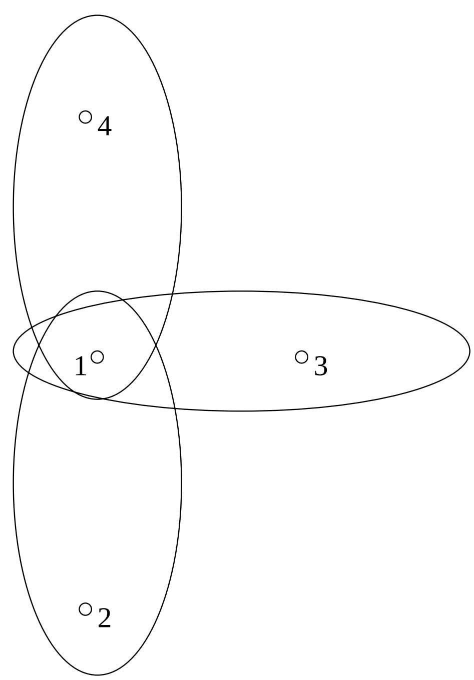

Figure 4: A claw.

intersection of a specified set of intersecting shores H_i in a proper representation (and no such other shores H_j) if and only if $cx \geq c_0$ has no claws.

Proof: We prove this theorem by showing that the only extreme points of the sequential lifting polyhedron are those that have this geometric interpretation. This in turn is shown by actually exhibiting what the minimum of the following linear program for any arbitrary non-negative objective function b is provided our inequality has no claws.

$$\begin{array}{lll} \text{minimize} & b \cdot \lambda & \\ \text{subject to} & & \\ & \lambda_i + \lambda_j \geq c_{ij} & \forall ij \in E \\ & \lambda_i \geq 0 & \forall i \in V \end{array} \tag{23}$$

If there is a negative b_i, then (23) is an unbounded linear program. Without loss of generality, let $\sum_{i \in V} b_i = 2$.

Consider the proper representation

$$cx = \sum_{i \in I} \alpha_i x(\delta(H_i)).$$

We construct our proposed optimal solution λ^* algorithmically as follows. For each $i \in I$, define

$$T_i = \begin{cases} H_i & \text{if } \sum_{j \in H_i} b_j > 1, \\ V \setminus H_i & \text{otherwise.} \end{cases} \tag{24}$$

Then do the following.

(i) Start with $\lambda_j^* = 0$ for all $j \in V$.

(ii) For each $i \in I$, add α_i to each λ_j^* for which $j \in V \setminus T_i$.

The resulting solution λ^* is now feasible for (23). Denote the (sequential) 1-node lifted inequality that results from the sequential lifting polyhedron point λ^* by $\bar{c}x \geq c_0$. Define $J := \{i \in I \,|\, T_i = H_i\}$. With $n+1$ being the new node, it can be easily verified that

$$\bar{c}x = \sum_{i \in J} \alpha_i x(\delta(H_i \cup \{n+1\})) + \sum_{i \in I \setminus J} \alpha_i x(\delta(H_i)).$$

Since the new node $n+1$ can only be added to shores H_i and H_j only if H_i and H_j already intersect because of (24) and $\sum_{i \in V} b_i = 2$, the required geometric interpretation of this 1-node lifting is satisfied.

Note that it follows from (ii) that

$$b \cdot \lambda^* = \sum_{i \in I} \alpha_i b(V \setminus T_i).$$

Hence, by (24), it is clear that λ^* is optimal among all extreme points with a geometric interpretation. What remains to be shown is that λ^* is an optimal solution to (23) in general. The dual to (23) is as follows:

$$\begin{array}{lll} \text{minimize} & c \cdot w & \\ \text{subject to} & & \\ & w(\delta(i)) \leq b_i & \forall i \in V \\ & w_{ij} \geq 0 & \forall ij \in E \end{array} \tag{25}$$

Suppose there existed a dual feasible vector w^* satisfying the complementary slackness conditions and one other condition, as listed below:

(i)$\lambda_i^* > 0 \Rightarrow w^*(\delta(i)) = b_i \quad \forall i \in V.$

(ii)$\lambda_i^* + \lambda_j^* > c_{ij} \Rightarrow w_{ij}^* = 0 \quad \forall ij \in E.$

(iii)$\lambda_i^* = 0 \Rightarrow w^*(\delta(i)) = \min\{b_i, \sum_{j \in I \setminus \{i\}} b_j\}.$

Then complementary slackness would imply that λ^* and w^* were optimal solutions for (23) and (25) respectively.

Such a vector w^* is constrained to be 0 on precisely the edges in the set

$$\bigcup_{i \in I} E(V \setminus T_i). \tag{26}$$

The existence of such a solution w^* that also satisfies (i) and (iii) is a fractional b-matching feasibility question, where b_j is the demand at vertex j for each $j \in V$. Note that because of (24), for no $i \in I$ do we have

$$b(V \setminus T_i) > b(T_i), \tag{27}$$

Also, because of (iii), for no $v \in V$ do we ever have

$$b_v > b(V \setminus \{v\}). \tag{28}$$

Either (27) or (28) would cause our fractional b-matching problem to be infeasible if it ever occured. In [5], it is proved that if the only edges not allowed in the fractional b-matching are those in (26) and that (27) and (28) never occurred, then the fractional b-matching problem is always feasible if and only if the set of shores $\{V \setminus T_i \mid i \in I\}$ does not have a claw. A feasible such fractional b-matching point w^* would imply that λ^* was optimal for (23). Conversely, no such feasible fractional b-matching point w^* would imply that λ^* was not optimal for (23) even though by construction λ^* is optimal among those extreme points of (23) that have a geometric interpretation. Therefore, this theorem follows. $\square$

There are several things to be noted here. The first is, the author does not know of any facet-defining inequality that is cut-based but has claws. On the other hand, even if there aren't any such TSP inequalities, there may be simultaneous triangular liftings even of the three-tooth comb which do not have the geometric interpretation stated in Theorem 5. Clearly, 1-node lifting (or sequential triangular lifting) is a special case of (simultaneous) triangular lifting. It is not clear, however, that simultaneous triangular lifting ever produces an inequality which could not also be obtained by sequential triangular lifting. The only thing in this respect that the author does know at present is that simultaneous triangular lifting of the subtour elimination constraints doesn't produce any other inequalities, as the following theorem shows.

Theorem 6 *The simultaneous triangular liftings of a subtour elimination inequality are all subtour elimination inequalities.*

Proof: The simple subtour elimination inequality for which 1 and 2 are on opposite shores is an inequality on only the 2 node graph K_2, and is merely $x_{12} \geq 2$. Suppose all the 0 node-liftings of this inequality (subtour elimination inequalities $x(\delta(H)) \geq 2$ where $1 \in H$, $2 \notin H$) were satisfied by a vector x^*. By the max-flow min-cut theorem, when capacities on the arcs (i,j) and (j,i) are both given by x^*_{ij} for each edge $ij \in E$, there exists a feasible 1-2 flow f in this directed graph of 2 units. Decompose this flow f into r flows, where the k-th flow has volume α_k and is along a directed 1-2 path P_k for $k = 1, \ldots, r$, with $\sum_{k=1}^{r} \alpha_k \geq 2$, and it is assumed without loss of generality that there are no cycles of flow. Let $cx \geq 2$ be any simultaneous triangular lifting of $x_{12} \geq 2$. Define $c_{(i,j)} = c_{(j,i)} := c_{ij}$ for each edge $e \in E$.

Because $cx \geq 2$ is a simultaneous triangular lifting of $x_{12} \geq 2$, we have for each path P_k that

$$c(E(P_k)) \geq c_{12} = 1.$$

We thus have

$$cx^* \geq c \cdot f = \sum_{k=1}^{r} \alpha_k c(E(P_k)) \geq \sum_{k=1}^{r} \alpha_k \geq 2.$$

Therefore, $cx \geq 2$ is implied by the set of subtour elimination inequalities $x(\delta(H)) \geq 2$ such that $1 \in H$, $2 \notin H$. Hence, this Theorem follows. $\square$

We are now ready to examine some of the less obvious 1-node liftings of what may be a non cut-based inequality. Call the subgraph $G = (V, E')$ of the complete graph $K_n = (V, E)$, *Hamiltonian* if it contains a Hamilton cycle. Call G *hypo-Hamiltonian* if G is non-Hamiltonian, but the graph $G - v$ obtained by removing any vertex v is always Hamiltonian. For every hypo-Hamiltonian graph G, there is a corresponding *hypo-Hamiltonian* inequality. The hypo-Hamiltonian inequalities were formulated and proven to be facet-defining by Grötschel in [7]. The hypo-Hamiltonian inequality for a hypo-Hamiltonian graph G is

$$x(E \setminus E') \geq 1.$$

This is a valid inequality because every Hamilton cycle must use an edge that is not in the hypo-Hamiltonian graph G. In tight triangular form, the hypo-Hamiltonian inequality for G is

$$x(E') + 2x(E \setminus E') \geq |V| + 1. \tag{29}$$

One such hypo-Hamiltonian graph is the *Petersen graph*, which is shown in Figure 5.

The right hand side of the *Petersen inequality* when expressed as in (29) is 11. Denote this Petersen inequality by $px \geq 11$. One can obtain all the 0 node-liftings of the Petersen inequality by the sequential 1-node liftings indicated by (16). But, we will show that the following is also an extreme point of the sequential lifting polyhedron:

$$\lambda_i^* = \begin{cases} 1/2 & i = 9, 10, \\ 3/2 & otherwise. \end{cases} \tag{30}$$

Denote the resulting 1-node lifted inequality by $\overline{p}x \geq 11$. One can easily see that λ^* is an extreme point since the objective function $\lambda_9 + \lambda_{10} + \lambda_2$

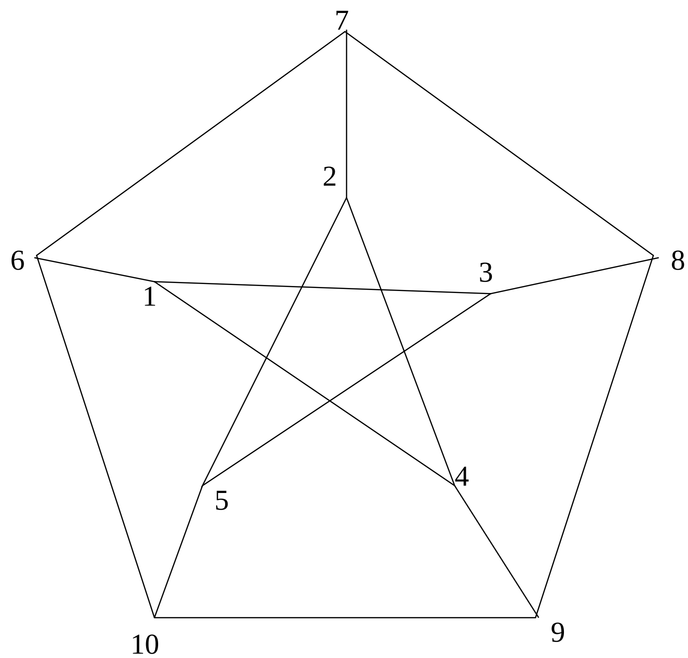

Figure 5: The Petersen graph.

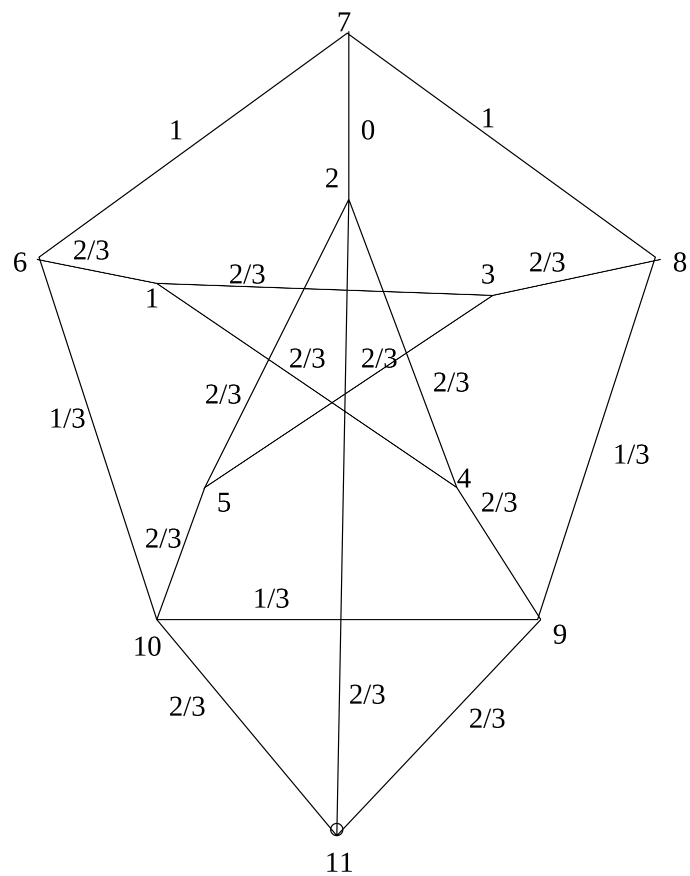

Figure 6: The subtour point x^*.

is minimized by λ^*, and given that $\lambda_9^* = \lambda_{10}^* = \frac{1}{2}$ and $\lambda_2^* = \frac{3}{2}$ and the constraints on λ imposed by the Petersen inequality, we must have $\lambda_i^* \geq 3/2$ for all $i \neq 2, 9, 10$.

This analysis leads us to define the subtour elimination point x^* shown in Figure 6. We have that $\overline{p}x^* = 10\frac{2}{3} < 11$ even though x^* satisfies all of the 0 node-liftings of the Petersen inequality.

In the next section, we show how to separate over classes of inequalities defined by all the (simultaneous) triangular liftings of a given inequality.

5 Separating Inequalities

Suppose you are trying to solve a TSP instance, and your branch and cut integer linear programming algorithm has just produced the fractional solution x^*. You suspect that x^* may violate one of the three-tooth comb inequalities, which you can then add as a cutting plane to cut the fractional solution x^* off. But how are you going to find which if any three-tooth comb inequalities it violates? We now know that every three-tooth comb inequality can be obtained by a (simultaneous) triangular lifting of one of the simple 6 node three-tooth comb inequalities in S_6. In fact, the sequential triangular liftings (1-node liftings) by themselves give you all the three-tooth comb inequalities. Since the class C of all simultaneous triangular liftings of inequalities in S_6 contains the class of three-tooth comb inequalities, we should be happy if we can can find that inequality in C which violates x^* the most or show that there are no such violating inequalities in polynomial time. The measure of the violation of $cx \geq c_0$ by x^* which we will use is simply $c_0 - cx^*$, which is greater than zero if and only if there is a violation.

More generally, suppose we are given the class S of all isomorphisms of a simple TT inequality $cx \geq c_0$. The class S_6 is an example of such a class, where the right hand side c_0 is 10. Consider the class C of all simultaneous triangular liftings of S. The question then is how do we find the most violated inequality in C?

Each inequality in C is a simultaneous triangular lifting of one or more of the inequalities in S. Suppose there are n vertices in our TSP instance, and k vertices in each simple inequality in S. Then there are $\binom{n}{k}$ different backbone sets B for inequalities in S, and hence $\frac{n!}{(n-k)!}$ different backbones, which is polynomial ($O(n^k)$) in terms of n. Each backbone corresponds to a particular inequality in S. So, if for each backbone, we could find in polynomial time the most violated inequality in C that is a simultaneous triangular lifting of that inequality in S which has that backbone, then we could find the most violated inequality in C in polynomial time. We could do this because we would just have to repeat this procedure for all $O(n^k)$ backbones.

So, consider a given backbone $\overline{B}$ of an inequality in S. Denote this inequality by $cx \geq c_0$. Given the fractional point x^*, how then, do we find the most violated inequality in C which is a simultaneous triangular lifting of $cx \geq c_0$? Here's what we do. Denote by B the backbone set for the backbone $\overline{B}$. So, B is a subset of our set V of vertices in our TSP instance on $K_n = (V, E)$. Define $E(B) := \{uv \in E \,|\, u \in B, v \in B\}$. Form the following linear program:

$$\begin{array}{lll} \text{minimize} & x^* \cdot \overline{c} & \\ \text{subject to} & & \\ & \overline{c}_{ij} = c_{ij} & \forall ij \in E(B) \\ & \overline{c}_{ij} \leq \overline{c}_{ik} + \overline{c}_{jk} & \forall i \neq j \neq k \in V \end{array} \tag{31}$$

Theorem 7 *The optimum value $\overline{c}^*$ to (31) corresponds to the valid TSP inequality $\overline{c}^* x \geq c_0$ which is violated by x^* the most among all simultaneous triangular liftings of $cx \geq c_0$.*

Proof: The polyhedron of feasible vectors for the linear program (31) is the same as the polyhedron in (17), namely the simultaneous lifting polyhedron. The simultaneous triangular liftings of $cx \geq c_0$ and the points in this polyhedron are in one to one correspondence. Since we are minimizing $x^* \cdot \overline{c}$, clearly we are obtaining the most violated such simultaneous triangular lifting, or, if $\overline{c}^* \cdot x^* \geq c_0$, we are showing there are no such violated inequalities. $\square$

Let S be the class of isomorphisms of the simple inequality $cx \geq c_0$ on a k node graph. Let C be the class of all simultaeous triangular liftings of inequalities in S. We now have the following theorem.

Theorem 8 *The class C of inequalities can be separated over in polynomial time.*

Proof: Let x^* be our fractional extreme point. For each of the $O(n^k)$ inequalities $c'x \geq c_0$ in S, we solve the linear program (31). The solution c^* which is the smallest among all $O(n^k)$ solutions of (31) corresponds to the inequality $c^* x \geq c_0$. This inequality is the most violated of all the inequalities in C by x^*. Hence, we have found the most violated inequality or shown that there are no such violated inequalities in the time needed to solve $O(n^k)$ linear programs (31). $\square$

Unfortunately, the result described in Theorem 8 is not a very practical one in its present form. What the author hopes is that clever methods for choosing the backbone will make Theorem 8 practically useful. We next will produce an even more general separation idea.

One should note that the inequality $cx \geq c_0$ could be any valid k node TSP inequality. So why not make the c_{ij}'s be variables also? Denote the class of all TSP inequalities on the k nodes in B whose coefficients satisfy the triangle inequality and which have a positive right hand side by $S(B)$, where each one is scaled so that the right hand side is 2. Denote all the simultaneous triangular liftings of these inequalities by $C(B)$. Denote by $H(B)$ the set of Hamilton cycles on the nodes of B. Consider the following linear program:

$$
\begin{array}{lll}
\text{minimize} & x^* \cdot \overline{c} & \\
\text{subject to} & & \\
& h \cdot c \geq 2 & \forall h \in H(B). \\
& \overline{c}_{ij} = c_{ij} & \forall ij \in E(B). \\
& \overline{c}_{ij} \leq \overline{c}_{il} + \overline{c}_{jl} & \text{for all } i \neq j \neq l \in V.
\end{array} \tag{32}
$$

Consider a feasible solution $(c, \overline{c})$ to (32). Note that since $h \cdot c \geq 2 \quad \forall h \in H(B)$, it follows that $cx \geq 2$ is a valid TSP inequality on the set of vertices B. We have imposed that the coefficients in c satisfy the triangle inequality. The analysis which follows is indifferent to whether or not we impose these constraints instead of replacing the current triangle inequality constraints in (32) by

$$\overline{c}_{ij} \leq \overline{c}_{il} + \overline{c}_{jl} \text{ for all } i \neq j \neq l \in V \text{ such that } l \notin B. \tag{33}$$

However, replacing the current triangle inequality constraints in (32) by (33) may lead to unbounded solutions, whereas adding large enough multiples of the degree constraints to any valid TSP inequality $cx \geq c_0$ will ensure that its coefficients satisfy the triangle inequality.

We now have the following theorem.

Theorem 9 *The optimal solution $(c^*, \overline{c}^*)$ to (32) corresponds to the valid TSP inequality $\overline{c}^* x \geq 2$ which is violated by x^* the most among the inequalities of $C(B)$.*

Proof: Let $(c, \overline{c})$ be a feasible point of (32). Since $h \cdot c \geq 2 \quad \forall h \in H(B)$, and the coefficients of c satisfy the triangle inequality, we have that $cx \geq 2$ is a valid TSP inequality in $S(B)$. By the arguments in Theorem 3, the simultaneous triangular lifting $\overline{c}x \geq 2$ of $cx \geq 2$ is a valid TSP inequality. Conversely, let $(c', \overline{c}')$ be given so that $c'x \geq 2$ is in $S(B)$ and $\overline{c}'x \geq 2$ is a simultaneous triangular lifting of $c'x \geq 2$. Then $(c', \overline{c}')$ is a feasible solution to (32). Since feasible solutions to (32) and inequalities in $C(B)$ are in one-to-one correspondence and $\overline{c}^*$ is the optimal solution to (32), it follows that

$\overline{c}^* x \geq 2$ is violated by x^* the most among the inequalities of $C(B)$, which is our theorem. $\square$

Define $S(k) := \bigcup_{|B|=k} S(B)$. Define $C(k)$ to be the set of all simultaneous triangular liftings of inequalities in $S(k)$. We now have:

Theorem 10 *The class $C(k)$ of inequalities can be separated over in polynomial time.*

Proof: The proof proceeds as in Theorem 8, except that this time we are repeatedly solving linear programs (32) for different backbone sets B. We have to solve $\binom{n}{k} = O(n^k)$ such linear programs in all. $\square$

6 Conclusion

We have two powerful separation techniques, maybe the better of which involves solving the linear program (32). Are these techniques of practical value? The author does not know at present since there are unresolved theoretical and practical questions. It would be nice to know how tight a formulation of the TSP we get for each k with the following linear program:

$$\begin{array}{ll} \text{minimize} & dx \\ \text{subject to} & \\ & cx \geq 2 \quad \forall c \in C(k). \end{array} \tag{34}$$

A proof that these formulations are fairly tight would be good news for these separation techniques.

The separation method of solving (32) could be quite useful if one could make smart guesses for the vertices in the backbone set B. The question then is, by examining x^* and maybe also the objective function d, how would one decide what the backbone set B should be. One could perhaps pick several different backbone sets, but the procedure for picking them would have to be pretty good for the separation method of solving (32) to be useful.

Another question is, just how fast can (32) be solved. The constraints of (32) don't have to be all explicitly put in since we have the separation algorithms of finding the minimum cost Hamilton cycle in a small (k node) graph and finding all-pairs shortest paths, where the distances are given by $\overline{c}$. But what is the fastest way to find the minimum cost Hamilton cycle in a small graph? Also, we do not in fact need all of the triangle inequality constraints given in (32). We need just enough of these constraints for the conclusion of Theorem 3 to be valid. So, we can reduce the number of constraints from $O(n^3)$ to $O(n^2)$. Alternatively, we do not need to find the all-pairs shortest paths in a separation algorithm for these constraints, but merely the shortest paths between all pairs of nodes in the backbone set.

We implemented a separation algorithm based on (32) as follows. We considered the support graph $G^* = (V, E^*)$ for a fractional solution x^*. We introduced a variable $\overline{c}_e$ for each edge $e \in E^*$. Since the objective function for (32) involves only edges which are in E^*, we can find the optimal solution $\overline{c}^*$ to (32) by considering only these edges, and allowing the value $\overline{c}_e^*$ on any edge $e \in E \setminus E^*$ to be as large as possible while still satisfying the triangle inequality constraints of (32). This means that for each edge $e \in E \setminus E^*$, we can assign to $\overline{c}_e^*$ the shortest path distance in G^* between the endpoints of e, where the length of edge $e \in E^*$ is given by $\overline{c}_e^*$.

We chose a backbone set B. We used an LP based algorithm which provides us with a tentative (possibly infeasible) optimal solution $\overline{c}^*$ to (32). We would like the following from this solution.

(i) $\overline{c}^* x \geq 2$ is a valid TSP inequality.

(ii) $\overline{c}^*$ satisfies triangle inequality constraints.

(iii) $\overline{c}^* \cdot x^* < 2$.

We tried finding constraints that are satisfied by feasible solutions to (32), but are violated by $\overline{c}^*$ as follows. Given $\overline{c}^*$, we found the shortest path distances in G^* between every pair of vertices in B. Denote these distances by c_e^* for $e \in E(B)$. We then found the minimum cost Hamilton cycle on the nodes of B, where the costs are given by c^*. If the cost of this Hamilton cycle H was less than 2, we then found a constraint that is satisfied by feasible solutions to (32), but is violated by $\overline{c}^*$ by the following. For each edge $e \in E(H)$ in this Hamilton cycle, replace this edge by the edges of the shortest path in G^* between the endpoints of e. This results in a Eulerian multi-graph which spans all the vertices of B and possibly other vertices using only edges in G^*. Denote the incidence vector of this Eulerian multi-graph by h^*. Then consider

$$h^* \cdot \overline{c} \geq 2. \tag{35}$$

Then (35) is satisfied by feasible solutions to (32) since when the coefficients $\overline{c}$ obey the triangle inequality, if Hamilton cycles on the vertices of B all have cost at least 2, then all of the Eulerian tours on the vertices of B have cost at least 2 as well. However, (35) is violated by $\overline{c}^*$.

So, we used the following algorithm:

(i) Start with an LP having variables for each edge $e \in E^*$, having only non-negativity constriants, and having the objective function of (32). This is the current LP.

(ii) Solve the current LP using dual simplex, obtaining the solution $\overline{c}^*$.

(iii) Find a constraint of the form (35) that $\overline{c}^*$ violates. If there are no such constraints, go to (v).

(iv) Add this constraint to the current LP. Go to (ii).

(v) Find $\overline{c}_e^*$ for $e \in E \setminus E^*$ by taking shortest path distances in G^* as previously described.

As we will prove, this algorithm finds the optimal solution to (32). This was our separation algorithm.

We now see why this separation algorithm works:

Theorem 11 *The resulting solution $\overline{c}^*$ from the separation algorithm just described is also the optimal solution to (32).*

Proof: We have already seen that the constraints of the form (35) are valid for (32). By assumption, $\overline{c}^*$ satisfies all the constraints of the form (35).

Suppose $\overline{c}^*$ violates the triangle inequality somewhere. This means that for some edge $ij \in E^*$, the shortest path distance between i and j in G^*, using $\overline{c}_e$ as the length of an edge $e \in E^*$, is less than $\overline{c}_{ij}^*$. But, we can redefine $\overline{c}_{ij}^*$ to be this shortest path distance. Then, all the constraints of the form (35) will still be satisfied, but $x^* \cdot \overline{c}^*$ will be strictly smaller. Hence, we know that $\overline{c}^*$ satisfies the triangle inequality constraints.

Define $c^* := \overline{c}^*|_{E(B)}$. Suppose $c^*x \geq 2$ is not a valid TSP inequality on the vertices of B, hence making $\overline{c}^*$ infeasible for (32). Let h be the incidence vector of a Hamilton cycle on the vertices of B such that $c^* \cdot h < 2$. Then one can find a corresponding Eulerian tour in G^*, whose incidence vector we denote by h^*, which spans all the vertices in B and possibly other vertices such that $\overline{c}^* \cdot h^* = c^* \cdot h$. This is because the values of $\overline{c}^*_e$ for $e \in E(B) \setminus E^*$ are determined by the shortest path computations on G^* done in (v). Hence, $h^* \cdot \overline{c}^* \geq 2$ would be a violated constraint of the form (35), which is a contradiction.

Therefore, $\overline{c}^*$ is feasible for (32). Hence this separation algorithm ultimately produces an LP whose optimal solution is $\overline{c}^*$, uses only inequalities valid for (32), and has the same objective function as (32). Since $\overline{c}^*$ is also feasible for (32), our theorem follows. □

Our separation algorithm required us to find the minimum cost Hamilton cycle on the vertices of B. We appealed to a 3-opt heuristic to do this, and checked to see whether it gave us the best tour only when necessary (such as at the end of our separation algorithm). We tried our separation algorithm for backbones of up to 39 vertices with success, although the algorithm is currently too slow to be practical. We hope that this separation algorithm can be improved upon in this respect.

References

[1] D. Applegate, R. Bixby, V. Chvätal, and W. Cook (1994). Finding Cuts in the TSP (a preliminary report), *Unpublished manuscript.*

[2] R. Carr (1997). Separating clique trees and bipartition inequalities having a fixed number of handles and teeth in polynomial time. *Mathematics of Operations Research.*

[3] R. Carr, (1995), Polynomial separation procedures and facet determination for inequalities of the traveling salesman polytope, *PhD Thesis, Department of Mathematics, Carnegie Mellon University.*

[4] R. Carr, (1996), Separating Over Classes of TSP Inequalities Defined by 0 Node-Lifting in Polynomial Time, *IPCO proceedings.*

[5] G. Christopher, *personal communication.*

[6] Fleischer, L., E. Tardos (1997). Separating Maximally Violated Comb Inequalities in Planar Graphs, *Technical Report.*

[7] M. Grötschel (1980). On the monotone symmetric travelling salesman problem: hypohamiltonian/hypotraceable graphs and facets, *Mathematics of Operations Research* **5** *285-292.*

[8] Johnson, D.S., C.H. Papadimitriou, Computational Complexity, in: E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, and D.B. Shmoys, eds., *The Traveling Salesman Problem* (John Wiley & Sons, Chichester, 1985, 37-85).

[9] M. Jünger, G. Reinelt, and G. Rinaldi, The Traveling Salesman Problem, in: M.O. Ball et al., eds., *Handbooks in OR & MS, Vol. 7* (Elsevier Science B.V., 1995).

[10] M. Jünger, G. Reinelt, and G. Rinaldi, The Traveling Salesman Problem, in: M.O. Ball et al., eds., *Handbooks in OR & MS, Vol. 7* (Elsevier Science B.V., 1995).

[11] D. Naddef and J. M. Clochard (1994). Some Fast and Efficient Heuristics for Comb Separation in the Symmetric Traveling Salesman Problem, *Technical Report, Universite Joseph Fourier Grenoble I*

[12] D. Naddef, G. Rinaldi (1992), The graphical relaxation: A new framework for the Symmetric Traveling Salesman Polytope, *Mathematical Programming 58, 53-88*

[13] Padberg, M.W., M.R. Rao (1982). Odd minimum cut sets and b-matchings, *Mathematics of Operations Research* **7** *67-80.*

[14] Padberg, M., G. Rinaldi (1989). A branch-and-cut approach to a travelling salesman problem with side constraints. *Management Sci.*, Vol. 35, No. 11, Nov., 1393-1412.

[15] Padberg, M., G. Rinaldi (1990). Facet identification for the symmetric traveling salesman polytope, *Mathematical Programming* **47** *219-257.*