The Lifted-Cut Relaxation of the Steiner Forest Problem

by

Siddharth Rajaram

An essay
presented to the University of Waterloo
in fulfilment of the
thesis requirement for the degree of
Master of Mathematics
in
Combinatorics and Optimization

Waterloo, Ontario, Canada, 2007

Abstract

In this essay we analyze properties of the lifted-cut linear programming relaxation of the Steiner Forest problem introduced by Könemann, Leonardi and Schäfer. We introduce this new formulation and prove some basic properties establishing that it is in fact a valid relaxation and analyzing the integrality gap. We are interested in analyzing the possible half-integrality of this relaxation and construct some families of graphs for which the unweighted minimum spanning tree instance defined by this relaxation has half-integral optimal solutions. Finally we described some computational and implementation issues, namely describing a polynomial sized flow formulation and rounding procedures that we used to find half-integral optimal solutions. We list the results of a computational study we conducted on Steiner Tree instances, stating half-integrality properties, as well as comparing the integrality gap with that of the standard undirected-cut relaxation.

Contents

Chapter 1

Introduction

We present the lifted-cut linear programming relaxation for the Steiner forest problem by Könemann, Leonardi, Schäfer and van Zwam in [3]. The formulation arises from an algorithm for constructing approximately budget-balanced cross-monotonic cost sharing mechanism for the Steiner forest problem. The approximation algorithm by Könemann, Leonardi, Schäfer in [2], is a modification of the primal-dual algorithm of Agarwal, Klein and Ravi in [1], which constructs a primal and dual solution for the standard undirected-cut linear programming relaxation simultaneously. However, the dual solution constructed by this new algorithm assigns values to variables that are not even present in the undirected-cut dual, prompting the question of constructing an alternate LP relaxation for which this dual is feasible, answered by the lifted-cut relaxation. In this chapter, we first describe the algorithm to motivate the new formulation, and then proceed to examine some basic properties of this relaxation.

1.1 Preliminaries

The problem under consideration is the *Steiner forest problem*, which is known to be NP-hard. It is a generalization of the well-studied Steiner Tree problem, as we will see in the following chapters. We are given an undirected graph $G = (V, E)$, a non negative cost function on the edge set $c : E \to \mathbb{R}^+$ and a set of $k \in \mathbb{N}$ *terminal pairs* $R = \{(s_1, t_1), ..., (s_k, t_k)\}$. The problem is to find a minimum-cost set of edges $F \subseteq E$ such that each terminal pair (s_i, t_i) lies in the same component of the forest induced by F. We abuse notation and also use the set R to denote the set of all terminals, i.e., $R = \bigcup_{i=1}^{k} \{s_i, t_i\}$. The set we are referring to will be clear from the context. We assume that each terminal $v \in R$ belongs to a unique terminal pair without loss of generality, as we can add a copy of v for each additional terminal pair and connect these copies to the original by zero-cost edges. We refer to the (unique) mate of any terminal $v \in R$ as $\bar{v}$, i.e. $(v, \bar{v}) \in R$.

We define the *time of death* (or *death time*) of a terminal pair $(v, \bar{v}) \in R$ to be $d(v, \bar{v}) = \frac{1}{2}c(v, \bar{v})$, where $c(v, \bar{v})$ denotes the cost of the miminum cost $v, \bar{v}$-path in G. Define the death time of a terminal $v \in R$ to be equal to the death time of its corresponding terminal pair, i.e.,

$$d(v) = d(\bar{v}) = \frac{1}{2}c(v, \bar{v}) \qquad \forall v \in R.$$

We now define some terms relating to cuts before we describe the working of the algorithm, which we will call KLS (introduced in [2]). A set of vertices is called a *Steiner cut* if it separates at least one terminal pair in R. Formally, $U \subset V$ is a Steiner cut if there exists a terminal pair $(s, t) \in R$ such that $|\{s, t\} \cap U| = 1$. Let $\mathscr{S}$ denote the set of all Steiner cuts. A *non-Steiner cut* is a set that contains at least one terminal pair and does not separate any terminal pairs in R. Formally, $U \subseteq V$ is a non-Steiner cut if there exists a terminal pair $(s, t) \in R$ such that $s, t \in U$ and for all terminal pairs $(s', t') \in R$, $|\{s', t'\} \cap U| \neq 1$. Let $\mathscr{N}$ denote the set of all non-Steiner cuts. Further, let $\mathscr{U} = \mathscr{S} \cup \mathscr{N}$ be the set of all Steiner and non-Steiner cuts.

1.2 The KLS Algorithm

The KLS algorithm of Könemann, Leonardi and Schäfer, introduced in [2], computes cross-monotonic cost shares for the terminal pairs, in order to construct an approximately budget-balanced group strategyproof cost sharing mechanism for the Steiner Forest game. For explanations of these terms and more details on this refer to the aforementioned paper [2] and the paper by Jain and Vazirani

[6], that outlines a general technique for constructing these mechanisms.

The algorithm computes a cost share for each terminal pair in R, while assigning values to 'dual variables' y_U for all $U \in \mathscr{U}$. The algorithm can be viewed as a continuous process over a time variable $\tau \geq 0$. The algorithm maintains a 'dual solution' y^τ, as well as a forest F^τ at all times τ. We use the term *moats*, to refer to the components of the forest F^τ. Let $\mathscr{A}^\tau$ denote the set of *active moats at time* τ, defined as moats that contain at least one terminal with death time greater than τ. We initialize F^0 to be the set of all singleton terminals and $y^0_U = 0$ for all $U \in \mathscr{U}$.

KLS uniformly raises the dual variables for all active moats, i.e. sets in $\mathscr{A}^\tau$, uniformly at all times $\tau \geq 0$. Suppose an edge from a vertex a in an active moat U to a Steiner vertex b outside it becomes *tight* at time τ, i.e.

$$c(\{a,b\}) = \sum_{\substack{\{a,b\} \in \delta(S) \\ S \in \mathscr{U}}} y^\tau_S.$$

We then add $\{a,b\}$ to F, replacing U with the set $U \cup \{b\}$ in $\mathscr{A}^\tau$ and continue. Similarly, suppose two active moats U_1 and U_2 collide at time τ, i.e. there is an edge $\{a,b\}$ with $a \in U_1$ and $b \in U_2$ such that $\{a,b\}$ becomes tight at time τ. Then, we add $\{a,b\}$ to F^τ, effectively removing U_1 and U_2 and adding $U_1 \cup U_2$ to $\mathscr{A}^\tau$. The algorithm terminates when $\mathscr{A}^\tau$ is empty. This must happen as all death-times are finite.

For terminal $v \in R$ with death time $\tau \leq d(v)$, let $U^\tau(v)$ denote the moat in $\mathscr{A}^\tau$ that contains v. Additionally, let $a^\tau(v)$ be the number of active terminals in $U^\tau(v)$, i.e. terminals with death time at least τ. The cost share of v is then given by:

$$\xi_R(v) = \int_{\tau=0}^{d(v)} \frac{1}{a^\tau(v)} d\tau$$

and, let $\xi_R(v, \bar{v}) = \xi_R(v) + \xi_R(\bar{v})$.

It is shown in [3] that the cost of the dual produced by KLS, i.e., $\sum_{U\subseteq V} y_U$, is at most twice the cost of the optimal Steiner forest. So we pose the problem of designing an LP relaxation for which the y constructed by the algorithm is a feasible dual solution and the quantity $\sum_{U\subseteq V} y_U$ is the objective value. The lifted-cut LP relaxation is formulated as a solution to this problem. We may thus interpret KLS as a primal-dual 2-approximation algorithm for Steiner Forests with this new LP relaxation.

1.3 Lifted-Cut Relaxation

In this section we describe the lifted-cut LP relaxation and prove some basic properties that establish that it is a valid relaxation, as it is not immediately evident looking at the relaxation itself. The relaxation arises when we try to design a LP relaxation for which the dual variables generated by the KLS algorithm are feasible. We also establish that in terms of objective value, this relaxation is at least as good as the classical relaxation, which we analyze experimentally with a computational study in Chapter 3. The results in this section were established in the paper [3] that introduced this relaxation.

To define the relaxation, we first index the terminal pairs $R = \{(s_1, t_1), ..., (s_k, t_k)\}$ so as to satisfy $d(s_1, t_1) \leq ... \leq d(s_k, t_k)$, without loss of generality. Define a precedence order $\prec$ on the set of terminal pairs by $(s_i, t_i) \prec (s_j, t_j)$ iff $i \leq j$. We extend this order to terminals by setting $s_1 \prec t_1 \prec s_2 \prec t_2 \prec ... \prec s_k \prec t_k$. We assume $v \prec v$ for all $v \in R$.

We look to construct a dual LP based on the algorithm, to ensure feasibility of the constructed solution. Let $\mathscr{S}_v \subseteq \mathscr{S}$ be the set of Steiner cuts that separate v and $\bar{v}$, where $(v, \bar{v})$ is the highest ranked terminal pair separated by that cut, i.e.,

$$\mathscr{S}_v = \{U \in \mathscr{S} \mid v \in U, \bar{v} \notin U, (u, \bar{u}) \prec (v, \bar{v}) \text{ for all } (u, \bar{u}) \in R(U)\}$$

where $R(U) = \{(s,t) \in R \mid |\{s,t\} \cap U| = 1\}$. Note that the sets $\mathscr{S}_v$ partition the set of all Steiner cuts $\mathscr{S}$. We say that v is *responsible* for a Steiner cut U, if $U \in \mathscr{S}_v$. Similarly, let $\mathscr{N}_v \subseteq \mathscr{N}$

be the set of non-Steiner cuts where $(v, \bar{v})$ is the highest ranked terminal pair, i.e.,

$$\mathscr{N}_v = \{U \in \mathscr{N} \mid \{v, \bar{v}\} \subseteq R \cap U, (u, \bar{u}) \prec (v, \bar{v}), \forall \{u, \bar{u}\} \subseteq R \cap U\}.$$

We say that v is responsible for a non-Steiner cut, if $U \in \mathscr{N}_v$. Note that if v is responsible for a non-Steiner cut, then $\bar{v}$ is also responsible for it. Also note that every $U \in \mathscr{U}$ has some (exactly one) terminal that is responsible for it. Then the *lifted-cut* dual, which we denote (LC-D) is given by:

$$\begin{aligned} \mathrm{opt}_{\text{LC-D}} &= \max \sum_{U \in \mathscr{U}} y_U && && (1.3.1) \\ \text{s.t.} \quad \sum_{\substack{U \in \mathscr{U} \\ e \in \delta(U)}} y_U &\leq c(e) && \forall e \in E && (1.3.2) \\ \sum_{U \in \mathscr{S}_v} y_U + \sum_{U \in \mathscr{N}_v} y_U &\leq d(v) && \forall v \in R && (1.3.3) \\ y_U &\geq 0 && \forall U \in \mathscr{U}. && (1.3.4) \end{aligned}$$

We can see intuitively that this dual is formulated with the KLS dual solution in mind, as we stop increasing y_U (i.e., remove U from $\mathscr{A}^\tau$) precisely when either one of the two inequalities 1.3.2 and 1.3.3 become tight. The primal, which we label (LC-P), is then given by:

$$\text{opt}_{\text{LC-P}} = \min \quad \sum_{e \in E} c(e) \cdot x_e + \sum_{v \in R} d(v) \cdot x_v \tag{1.3.5}$$

$$\text{s.t.} \quad \sum_{e \in \delta(U)} x_e + x_v \geq 1 \qquad \forall v \in R, \forall U \in \mathscr{S}_v \tag{1.3.6}$$

$$\sum_{e \in \delta(U)} x_e + x_v + x_{\bar{v}} \geq 1 \qquad \forall v \in R, \forall U \in \mathscr{N}_v \tag{1.3.7}$$

$$x_e, x_v \geq 0 \qquad \forall e \in E, \forall v \in R. \tag{1.3.8}$$

This LP was defined so that the dual constructed by the KLS algorithm is feasible for (LC-D). We will now show that the above LP is indeed a relaxation for the Steiner forest problem, meaning that integral solutions do correspond to feasible Steiner forests and that the optimal objective value of the LP is at most the cost of the optimal Steiner forest. We then state some properties of the performance of this LP, particularly in comparison to the classical undirected-cut formulation. A simple projection to the edge variables does not make this a valid relaxation. We need to convert the terminal variables into edges as follows.

1.3.1 Lemma. *Let x be a feasible integral solution for (LC-P). Then there is a feasible Steiner forest of cost at most*

$$\sum_{e \in E} c(e) \cdot x_e + \sum_{v \in R} d(v) \cdot x_v.$$

Proof. We initially define the set $F = \{e \in E \ : \ x_e = 1\}$ that will be the edge set of our feasible Steiner forest. Note that if F contains cycles, we may successively delete cycle edges to ensure it is a forest. If this forest is infeasible, there exists a Steiner cut $U \in \mathscr{S}$, such that $\delta(U) \cap F = \emptyset$. Let v be the terminal responsible for U in our LP relaxation and let $\bar{v}$ be the mate of v. Note that it must be the case that $\bar{U} = V \backslash U \in \mathscr{S}_{\bar{v}}$. Since $\delta(U) = \delta(\bar{U})$, the constraint (1.3.6) for U and $\bar{U}$ give us that $x_v = x_{\bar{v}} = 1$.

We can then add all the edges along the shortest path between v and $\bar{v}$ to F for an additional cost of $2d(v, \bar{v})$. This then satisfies the constraint (1.3.6) for all Steiner cuts in $\mathscr{S}_v$ and $\mathscr{S}_{\bar{v}}$. We can repeat this for all U that violate the cut requirement for a total cost of at most $\sum_{v\in R} d(v)\cdot x_v$. The initial cost of F was at most $\sum_{e\in E} c(e)\cdot x_e$, and adding the two gives us the required bound on the cost. □

We now show that the objective value of the LP is bounded above by the cost of the optimal Steiner tree, which we denote opt_R.

1.3.2 Lemma. *Let F be a feasible solution for the given Steiner forest instance. There exists a half-integral feasible solution x to (LC-P) satisfying:*

$$\sum_{e\in E} c(e)\cdot x_e + \sum_{v\in R} d(v)\cdot x_v \leq c(F).$$

Therefore, $opt_{LC-P} \leq opt_R$.

Proof. We construct a solution x that is feasible for (LC-P) and show the following inequality that for each tree $T \in F$ with vertex-set $V(T)$ and edge-set $E(T)$:

$$\sum_{e\in E(T)} c(e)\cdot x_e + \sum_{v\in R\cap V(T)} d(v)\cdot x_v \leq c(T).$$

The inequality stated then follows by summing over all $T \in F$. Let T be an arbitrary tree in F and let u be the highest ranked terminal in $V(T)$. Let $P_{u\bar{u}}$ denote the shortest path between u and $\bar{u}$ in T. Let $x_e = \frac{1}{2}$ for all edges $e \in E(P_{u\bar{u}})$ and $x_e = 1$ for all other edges in $E(T)$. Also, let $x_u = x_{\bar{u}} = \frac{1}{2}$ and $x_v = 0$ for all other $v \in R \cap V(T)$. By the definition of death time $\frac{1}{2}\cdot c(P_{u\bar{u}}) \geq d(u,\bar{u})$. So the objective value of x restricted to T satisfies:

$$\sum_{e\in E(T)} c(e)\cdot x_e + \sum_{v\in R\cap V(T)} d(v)\cdot x_v \leq c(T).$$

We can do the same for each $T \in F$ and add up these inequalities to get the required bound on the cost of x. It only remains to be shown that x thus constructed is feasible. So consider some Steiner cut $U \in \mathscr{S}_v$ where $v \in R \cap V(T)$. If U contains exactly one of u and $\bar{u}$, then it must be the case that $v = u$ or $v = \bar{u}$. The boundary $\delta(U)$ must also intersect $P_{u\bar{u}}$ at least once, at some edge e. Then

$x_u + x_e = 1$ (or $x_{\bar{u}} + x_e = 1$) satisfies the constraint (1.3.6). If U contains none or both of u and $\bar{u}$, then it either does not intersect $P_{u\bar{u}}$ or intersects it at least twice. In the first case $\delta(U)$ contains some edge $e \in E(T)\backslash E(P_{u\bar{u}})$ and we have $x_e = 1$. In the second case let e and e' be two edges in $E(P_{u\bar{u}}) \cap \delta(U)$, then $x_e + x_{e'} = 1$, showing that x always satisfies constraint (1.3.6).

For a non-Steiner cut $U \in \mathscr{N}_v$ with $v \in V(T)$, we have two cases again. Either $v \in \{u, \bar{u}\}$ or not. If $v \notin \{u, \bar{u}\}$, we know that both u and $\bar{u}$ are not in U, otherwise it would either be a Steiner cut or u and $\bar{u}$ would be responsible for U. But then similar to above, U either does not intersect $P_{u\bar{u}}$ or intersects it at least twice, satisfying constraint (1.3.7). If $v = u$ or $v = \bar{u}$, then since U is a non-Steiner cut, it must contain both u and $\bar{u}$ and $x_u + x_{\bar{u}} = 1$ shows that constraint (1.3.7) is always satisfied, proving the lemma. □

We will now prove a result stating the lifted-cut dual (LC-D) is at least as strong as the standard undirected-cut relaxation dual. The undirected-cut primal, which we denote (P) is given by:

$$\text{opt}_{\text{P}} = \max \sum_{e \in E} c(e) x_e \tag{1.3.9}$$

$$\text{s.t.} \quad \sum_{e \in \delta(U)} x_e \geq 1 \qquad \forall U \in \mathscr{S}, \tag{1.3.10}$$

$$x_e \geq 0 \qquad \forall e \in E. \tag{1.3.11}$$

The corresponding dual (D) is given by:

$$\text{opt}_{\text{D}} = \max \sum_{U \in \mathscr{S}} y_U \tag{1.3.12}$$

$$\text{s.t.} \quad \sum_{\substack{U \in \mathscr{S} \\ e \in \delta(U)}} y_U \leq c(e) \qquad \forall e \in E \tag{1.3.13}$$

$$y_U \geq 0 \qquad \forall U \in \mathscr{S}. \tag{1.3.14}$$

1.3.3 Lemma. *Let y be a feasible solution for (D), then there is a feasible solution y' for (LC-D) that satisfies*

$$\sum_{U \in \mathscr{S}} y_U \leq \sum_{U \in \mathscr{U}} y'_U$$

implying that $opt_D \leq opt_{LC-D}$.

Proof. We construct y' from y by letting

$$y'_U = y'_{\bar{U}} = \frac{y_U + y'_{\bar{U}}}{2}$$

and $y'_U = 0$ for all non-Steiner cuts $U \in \mathscr{N}$.

To see that y' satisfies all constraints, first note that U is a Steiner cut if and only if $\bar{U} = V \backslash U$ is one too. Then the constraint (1.3.13) gives us:

$$c(e) \geq \sum_{\substack{U \in \mathscr{S} \\ e \in \delta(U)}} y_U = \sum_{\substack{U \in \mathscr{S} \\ e \in \delta(U)}} \frac{y_U + y_{\bar{U}}}{2} = \sum_{\substack{U \in \mathscr{S} \\ e \in \delta(U)}} y'_U = \sum_{\substack{U \in \mathscr{U} \\ e \in \delta(U)}} y_U$$

where the last equality holds because $y'_U = 0$ for all non-Steiner cuts U. Thus, y' satisfies (1.3.2).

Now, suppose that y' violates constraint (1.3.3) for some terminal $v \in R$ with mate $\bar{v}$. Since all non-Steiner cuts are zero, this implies

$$\sum_{U \in \mathscr{S}_v} y'_U > d(v) = \frac{c(P_{v\bar{v}})}{2} \tag{1.3.15}$$

where $P_{v\bar{v}}$ is the minimum cost $v, \bar{v}$ path in G. Now note that the set $\mathscr{S}_v \cup \mathscr{S}_{\bar{v}} \subseteq \{U \in \mathscr{S} \ : e \in \delta(U), e \in E(P_{v\bar{v}})\}$, since every Steiner cut separating v and $\bar{v}$ must intersect an edge of the path $P_{v\bar{v}}$. Adding up the y' variables over the set $\mathscr{S}_v \cup \mathscr{S}_{\bar{v}}$ gives

$$\sum_{U \in \mathscr{S}_v} y'_U + \sum_{U \in \mathscr{S}_{\bar{v}}} y'_U \leq \sum_{e \in E(P_{v\bar{v}})} \sum_{\substack{U \in \mathscr{S} \\ e \in \delta(U)}} y_U.$$

Noting the symmetry in the definition of y' (i.e., $y'(U) = y'(\bar{U})$) and the fact that $U \in \mathscr{S}_v \iff U \in \mathscr{S}_{\bar{v}}$, the LHS above becomes $2 \cdot \sum_{U \in \mathscr{S}_v} y'_U$. Then adding up constraints (1.3.13) over all edges in $P_{v\bar{v}}$ we get the RHS, implying

$$2 \cdot \sum_{U \in \mathscr{S}_v} y'_U \leq c(P_{v\bar{v}})$$

contradicting (1.3.15). □

It is shown in [1] that the $\text{opt}_D \leq \text{opt}_R \leq 2 \cdot \text{opt}_D$, which with the above result along with Lemma 1.3.2 imply the same bound for (LC-D), i.e. $\text{opt}_{LC-D} \leq \text{opt}_R \leq 2 \cdot \text{opt}_{LC-D}$. This also shows that the forest generated by the KLS algorithm is a 2-approximation to the optimal Steiner forest.

Könemann, Leonardi, Schäfer and van Zwam [3] also showed that there exist instances where the above inequality is strict by considering a spanning tree instance on a cycle C_n with unit weight edges. In section 2.2 we show that in this case the optimal solution for the lifted-cut relaxtion has cost approximately $\frac{3n}{4}$, while the optimal for the undirected-cut relaxation is $\frac{n}{2}$ (obtained by setting $x_e = \frac{1}{2}$ for each edge). Similarly, they show that the IP/LP gap for this relaxtion is exactly 2 by considering a spanning tree instance on a complete graph K_n with unit weight edges, which clearly has $\text{opt}_R = n - 1$. In section 2.1 we show that the optimal solution for the lifted-cut relaxation is $\frac{n}{2}$, showing the gap is arbitrarily close to 2.

Chapter 2

Half-Integrality Properties

We are interested in classifying instances when we are guaranteed optimal half-integral solutions (when all variables in the optimal solution have values 0, 1/2 or 1) to (LC-P), as rounding up to an integer solution immediately gives us an LP-based 2-approximation algorithm that is at least as strong as KLS (or as strong as solving the undirected-cut LP). Certain Steiner tree instances have been shown to yield integer polytopes with the undirected-cut formulation, particularly instances defined on series-parallel graphs, as shown by Goemans in [4]. We are interested in a weakened version of this question for (LC-P), that is just to classify some half-integral instances. In this chapter, we first show that the classical undirected-cut LP relaxation is not always half-integral. We then show that for certain special cases of the problem, the polytope defined by the lifted-cut relaxation always admits a half-integral solution.

We construct some special classes of unweighted (i.e. unit weight on all edges) graphs for which the minimum spanning tree instance defined by (LC-P) has a half-integral optimal solution. Note that we define the terminal set R for the MST problem by picking some root vertex v_0 and letting $R = \{(v, v_0) | v \in V \backslash \{v_0\}\}$. Of course, by our convention of at most one terminal per vertex we need to create $|V| - 1$ new terminal vertices that are connected to v_0 by zero-cost edges. For convenience, we will still refer to x_{v_0}, but will be explicit in mentioning its mate. When we refer to x_v for any other vertex, the corresponding pair is implicitly (v_0, v).

For the primal solution, we may just assign $x_e = 1$ for all these zero-cost edges without affecting optimality. Therefore any cuts containing such edges in the boundary are trivially satisfied. Similarly by the dual constraint (1.3.2) on these zero edges, the dual variables for these cuts must also be zero. Hence, when we speak of cuts, we may contract all the terminals onto v_0 for simplicity.

We need to be explicit with the choice of root (when we are talking about a Steiner tree or spanning tree instance) and the ordering of terminal pairs (only in case of ties), as the objective value and half-integrality of the optimal solution can vary based on these parameters. We are interested if there always exists a root and an ordering such that the unweighted spanning tree instance has a half-integral optimal solution.

It is well known that the undirected-cut relaxation is not always half-integral. We begin by presenting an example by Vazirani from [7] showing this. The optimal lifted-cut solution was given in [3].

2.0.4 Lemma. *Consider the Peterson graph $G = (V, E)$ and the spanning tree problem (i.e. $R = (v, w)$ for some $v \in V$ and all $w \in V \backslash \{v\}$) with each edge having unit cost. The undirected cut relaxation (P) does not have a half-integral optimal solution for this instance, while the lifted-cut relaxation (LC-P) does have a half-integral optimal solution.*

Proof. Consider the solution $\hat{x}$ to (P) given by assigning $\hat{x}_e = \frac{1}{3}$ for all edges $e \in E$. We know that the Peterson graph is 3-edge connected, so every cut must be crossed by at least three edges, satisfying the cut constraint. We argue that this must be optimal. The sum of all x_e adjacent to a given vertex must be at least 1, by the corresponding cut constraints. Summing up this quantity over all vertices, gives us twice the sum of the x_e over all edges.

$$10 \leq \sum_{v \in V} \sum_{e \in \delta(\{v\})} x_e = 2 \sum_{e \in E} x_e.$$

But this lower bound is matched by the solution $\hat{x}$, proving that it must be optimal. Now suppose we had a half-integral solution $\tilde{x}$ of the same cost. If $\tilde{x}_{\bar{e}} = 1$ for some edge $\bar{e} = \{u, v\}$, then consider the cut $\{u, v\}$. The constraint for this cut implies that for some $e' \neq \bar{e}$ adjacent to u, we have $x_{e'} > 0$, without loss of generality. But from above we know that since $\tilde{x}$ is optimal, it must be the case that

$$\sum_{e \in \delta(\{u\})} \tilde{x}_e = 1,$$

a contradiction. So, now consider the set of edges $\tilde{E} = \{e \in E \mid \tilde{x}_e = \frac{1}{2}\}$. Since we know the cost of the solution to be 5, we can conclude that $|\tilde{E}| = 10$. By the analysis above, there are exactly two edges of $\tilde{E}$ adjacent to each vertex in V. But this implies that $\tilde{E}$ are the edges of a Hamiltonian cycle, a contradiction because the Peterson graph is known to have no Hamiltonian cycles.

Now, we construct a feasible half-integral primal solution and a dual solution of the same cost for this instance with the lifted-cut relaxation. We may pick any vertex v_0 as the root (all vertices are equivalent due to node symmetry) for the terminal set. Note that there is always a Hamiltonian path starting at v_0 and terminating at some $\bar{v}_0$ at maximum distance (which is 2) from v_0. For instance, Figure 2.1, shows such a Hamiltonian path, and under automorphisms this exists for all choices of root.

Then, we construct the half-integral optimal solution x' as follows. Let $\bar{v}_0$ be some vertex that is at distance 2 away from v_0. Set $(v_0, \bar{v}_0)$ to be the top ranked terminal pair. Note that the ordering of the rest of the pairs is irrelevant in the case of multiple pairs with equal death times. Let $P = \langle v_0, \ldots, \bar{v}_0 \rangle$ be the Hamiltonian path from above. Then, let $x'_{v_0} = \frac{1}{2}$ from the pair $(v_0, \bar{v}_0)$ and $x'_{\bar{v}_0} = \frac{1}{2}$ and $x'_v = 0$ for all other terminals. Let $x'_e = \frac{1}{2}$ for all $e \in E(P)$ and $x'_e = 0$ otherwise.

Any Steiner cut containin neither or both end points of P must intersect P twice, and thus $x'_e = \frac{1}{2}$ for those two edges ensures that the constraint 1.3.6 is satisfied. Any Steiner cut containing exactly one of v_0 and $\bar{v}_0$ intersect the path at least once and the $\frac{1}{2}$ on that edge together with $x'_{v_0} = \frac{1}{2}$ or $x'_{\bar{v}_0} = \frac{1}{2}$ satisfies the constraint 1.3.6. The only non-Steiner cut for an MST instance is V itself and the

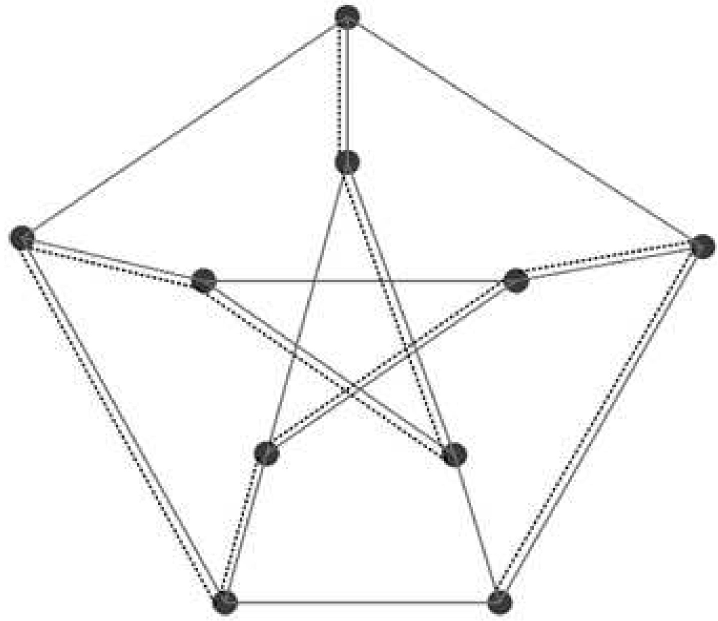

Figure 2.1: Hamiltonian path between top ranked pair on the Peterson graph

corresponding constraint 1.3.7 is satisfied by $x'_{v_0} = x'_{\bar{v}_0} = \frac{1}{2}$. Thus, x' is feasible for (LC-P) and has cost 5.5.

We construct a feasible dual solution y' of equal cost, proving optimality of both by weak duality. Let $y'_{\{v\}} = \frac{1}{2}$ for all $v \in V$, $y'_V = \frac{1}{2}$ and $y'_U = 0$ for all other subsets $U \subset V$. All vertices except v_0 and $\bar{v}_0$ contain exactly one terminal on them with death time $\frac{1}{2}$ or 1 and are only responsible for the singleton cuts (among all non-zero cuts), satisfying constraint 1.3.3. Terminal v_0 on pair $(v_0, \bar{v}_0)$ and terminal $\bar{v}_0$ are both responsible for the non-Steiner cut V as well as the respective singleton cuts, satisfying constraints 1.3.3 since their death times are 1. Constraints 1.3.2 are satisfied trivially, since the only non-zero cuts containing any edges in their boundary are singletons and edges have unit cost. One can see that the solution y' has the same cost 5.5, concluding the proof. □

2.1 Hamiltonian Path Between Top Pair

The only property of the Peterson graph that we used to construct the half-integral optimal solution above was the existence of the Hamiltonian path between the top-ranked terminal pair. We then naturally generalized the above solution as follows. Suppose we have some pair of vertices u and v, such that $c(u, v) \geq c(u, w)$ for all $w \in V$ and there exists a Hamiltonian path $P = \langle u, \ldots, v \rangle$. Then letting u be the root for the terminal set and setting (u, v) as the highest ranked terminal pair, we get a half integral solution of cost $\frac{|V|-1}{2} + d(u, v)$ as follows. Note that the ordering of the rest of the pairs is irrelevant in the case of multiple pairs with equal death times. We explicitly construct this solution as follows: $x_u = \frac{1}{2}$ for the pair (u, v), $x_v = \frac{1}{2}$ and $x_w = 0$ for all other $w \in R$, $x_e = \frac{1}{2}$ for all $e \in E(P)$ and 0 otherwise.

As any Hamiltonian path must contain $|V| - 1$ edges, the cost of the edges on this path is $\frac{|V|-1}{2}$. The solution x is feasible, since any Steiner cuts that contain neither u or v, or both u and v, must intersect P twice on edges e_1 and e_2 and $x_{e_1} + x_{e_2} = 1$. Steiner cuts containing exactly one of u or v must intersect P at least once at some edge e_1 and $x_{e_1} = x_u = x_v = \frac{1}{2}$ (for the pair (u, v)) satisfies the constraint. The only non-Steiner cut for any MST instance is V itself, and since we set (u, v) to be the highest ranked pair and $x_u + x_v = 1$ for this pair, this proves feasibility for the primal.

We construct a feasible dual solution of equal cost as follows: $y_{\{u\}} = \frac{1}{2}$ for all $u \in V$, $y_V = d(u, v) - \frac{1}{2}$ and $y_U = 0$ for all other $U \in \mathscr{U}$. Since the cost of all the singletons is $\frac{|V|}{2}$, this solution can be seen to have the same cost as the primal above, so we just need to check for feasibility. By construction, the constraints (1.3.2) are satisfied with equality for each edge of the graph. Because the death time of each terminal is at least $\frac{1}{2}$, all terminals on vertices except $V \backslash \{u, v\}$ satisfy constraint (1.3.3). The vertices u and v are both responsible for the non-Steiner cut V as well as the singleton Steiner cuts and thus satisfy (1.3.3) with equality, proving dual feasibility and half-integrality for such graphs. We can therefore conclude the following theorem.

2.1.1 Theorem. *Given an graph $G = (V, E)$, if there exists a pair of vertices u and v, such that $c(u, v) \geq c(u, w)$ for all $w \in V$ and a Hamiltonian path from $\langle u, \cdots, v\rangle$, then for a certain choice of root and certain ordering, there exists a half-integral optimal solution to (LC-P) on the unweighted MST instance defined on G with objective value $\frac{|V|-1}{2} + \frac{c(u,v)}{2}$.*

2.2 Cycles

We also constructed optimal half-integral solutions for the unit-edge-weight MST on cycles. Let us fix some root v and denote $(v, \bar{v})$ as the top ranked terminal pair. The choice of root is irrelevant as cycles are vertex transitive and similarly if there are two choices for $\bar{v}$, they are equivalent under automorphism. First, we assume our cycle has 4 or more vertices, as C_3 is half-integral (i.e., for a certain choice of root and certain ordering, there exists a half-integral optimal solution to (LC-P) on the unweighted MST instance defined on C_3) by the Hamiltonian path solution above.

In cycles, there are two edge-disjoint paths connecting v and $\bar{v}$, say P_1 and P_2. Let $E(P_1)$ and $E(P_2)$ respectively be the edge sets of these paths. Since, our cycles have at least 4 vertices, we know that $|P_1|, |P_2| \geq 3$. Consider the following primal solution: $x_e = \frac{1}{2}$ for all edges e, $x_v = \frac{1}{2}$ (on the pair $(v, \bar{v})$), $x_{\bar{v}} = \frac{1}{2}$ and $x_u = 0$ for all other $u \in R$. For any cut $\subsetneq \neq U \subsetneq V$, $\delta(U)$ contains at least two edges e_1 and e_2 and $x_{e_1} + x_{e_2} = 1$ satisfies the constraint (1.3.6). For $U = V$, we have the top terminal pair giving us $x_v + x_{\bar{v}} = 1$, satisfying the only non-Steiner cut constraint (1.3.7). The cost of this primal solution is $\frac{|V|}{2} + d(v, \bar{v}) = \frac{|V|}{2} + \frac{1}{2}\left\lfloor \frac{|V|}{2} \right\rfloor$.

The corresponding dual solution is given by: $y_{\{u\}} = \frac{1}{2}$ for all $u \in V \backslash \{v, \bar{v}\}$, $y_{P_1} = y_{P_2} = \frac{1}{2}$, $y_V = d(v, \bar{v}) - \frac{1}{2}$ and $y_U = 0$ for all other $U \in \mathscr{U}$. To prove feasibility of this dual solution, let $P_1 = \langle v, u_1, u_2, \ldots, u_j, \bar{v}\rangle$. For all edges $u_i u_{i+1}$, constraint (1.3.2) is satisfied with equality, as the singleton sets are the only cuts containing them in their boundary. For edges vu_1 and $u_j\bar{v}$, the edge constraint (1.3.2) is satisfied because the only non-zero cuts containing them in the boundary are P_2 and the singletons $\{u_1\}$ and $\{u_2\}$ respectively.

Each terminal on vertices in $V\backslash\{v, \bar{v}\}$ has death time at least $\frac{1}{2}$, and each terminal on these vertices is only responsible for the singleton set containing it. So all these terminals satisfy constraint (1.3.3). Since, we are considering cycles of order 4 or greater, both $P_1\backslash\{v, \bar{v}\}$ and $P_2\backslash\{v, \bar{v}\}$ are non-empty. Let w_1 and w_2 be highest ranked terminals in each of the two sets respectively. We know that $d(v, w_1), d(v, w_2) \geq \frac{1}{2}$, because all edges are unit weight. Then, the terminal on v from the pair (v, w_2) is responsible for the cut P_1 and terminal on v from the pair (v, w_1) is responsible for the cut P_2, so both those terminals also satisfy the constraints (1.3.3). The terminal v (from pair $(v, \bar{v})$) and $\bar{v}$ are responsible for the only non-Steiner cut V, so the corresponding constraint (1.3.3) is satisfied with equality. All remaining terminals (all on vertex v) are not responsible for any cuts with positive value in the dual solution, proving that this dual is feasible and we can easily see that it has the same cost $\frac{|V|}{2} + \frac{1}{2}\left\lfloor\frac{|V|}{2}\right\rfloor \approx \frac{3|V|}{4}$. We can then conclude with the following theorem.

2.2.1 Theorem. *For any choice of root and ordering, there exists a half-integral optimal solution to (LC-P) on the unweighted MST instance defined on a cycle C_n with objective value $\frac{n}{2} + \frac{1}{2}\left\lfloor\frac{n}{2}\right\rfloor$.*

2.3 Cycles with Chords from a Single Vertex

We extended our solution of cycles to consider cycles with chords added from one fixed vertex. More precisely, for some cycle C with edge set $E(C)$, we consider the graph with the same vertex set V and edges $E(C) \cup \{(v, v') \mid v' \in V'\}$ for some $v \in V$ and some non-empty subset of vertices $V' \subseteq V\backslash\{v\}$ (see Fig. 2.2). First, we eliminate the easy case where $V' = V\backslash\{v\}$. If we set v as the root for the MST, notice that all terminal pairs have death time $\frac{1}{2}$. Then consider the cycle C and one of the two vertices u adjacent to v on this cycle, if we set (v, u) to be the top ranked pair, we have a Hamiltonian path from v to u ($E(C)\backslash\{vu\}$) and are done by the earlier solution. So we assume that there is at least one vertex (other than v itself) that is not adjacent to v.

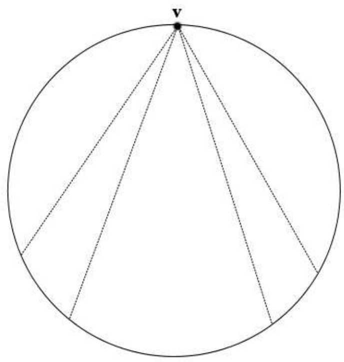

Figure 2.2: Cycles with Chords from a Single Vertex

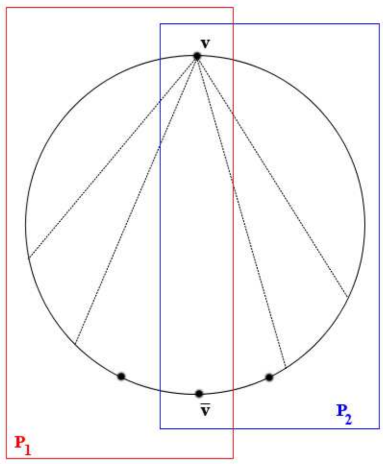

Figure 2.3: Paths P_1 and P_2

In this case if there are multiple choices for top ranked terminal pair, it is irrelevant which one we pick. Let $(v, \bar{v})$ be the top ranked pair. As in the previous case, we consider the two edge-disjoint paths P_1 and P_2 between v and $\bar{v}$ along the cycle C. Let $x_e = \frac{1}{2}$ for all $e \in E(P_1) \cup E(P_2)$, $x_v = \frac{1}{2}$ for the pair $(v, \bar{v})$ and $x_{\bar{v}} = \frac{1}{2}$ and $x_e, x_u = 0$ for all other $e \in E, u \in R$. The feasibility follows exactly from the previous case, since the boundary of every Steiner cut must intersect $E(C)$ twice and $x_v + x_{\bar{v}} = 1$ for the top ranked pair, which satisfies the non-Steiner cut V. The cost of the solution is also $\frac{|V|}{2} + d(v, \bar{v})$.

The dual solution also follows the same structure as before $y_{\{u\}} = \frac{1}{2}$ for all $u \in V \backslash \{v, \bar{v}\}$, $y_{P_1} = y_{P_2} = \frac{1}{2}$ and $y_V = d(v, \bar{v})$. Let $P_1 = \langle v, u_1, \ldots, u_j, \bar{v} \rangle$. All edges with both ends in $E(P_1) \backslash \{vu_1, u_j\bar{v}\}$ satisfy constraint (1.3.2), because the singletons are the only non-zero cuts containing them in their boundaries. The only non-zero cuts containing any edge vu_i ($u_i \in P_1$) in their boundaries are P_2 and the singleton $\{u_i\}$, satisfying (1.3.2) and the same holds for the edge $u_j\bar{v}$. Doing a similar analysis for P_2, we see that constraint (1.3.2) is satisfied for all edges in the graph.

As before, we can see that none of the terminals on $V \backslash \{v\}$ violate constraint (1.3.3). And the terminal on v from the pair (v, w_2) is responsible for P_1 (where w_2 is the top ranked terminal in $P_2 \backslash \{v, \bar{v}\}$), satisfying (1.3.3), and similarly for P_2. The terminal on v from $(v, \bar{v})$ and $\bar{v}$ are only responsible for the non-Steiner cut V (among non-zero cuts), satisfying the constraint. All other terminals on v are not responsible for any non-zero cuts, giving us the following result.

2.3.1 Theorem. *Let G be a graph that is a cycle with chords that are all adjacent a fixed vertex. For a given root vertex and ordering, there exists a half-integral optimal solution to (LC-P) on the unweighted MST instance defined on G.*

2.4 Cycles with Two Crossing Chords

We were also able to show half-integrality for the MST on cycles with two crossing chords (as shown in Fig. 2.4) using a similar solution structure to the previous two cases, but requiring a more

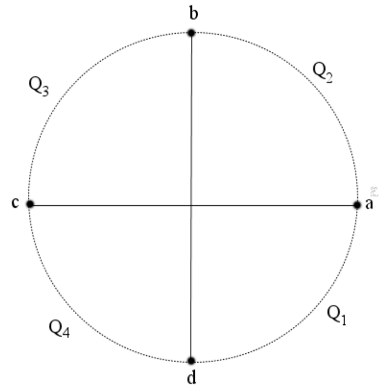

Figure 2.4: 2 crossing chords

thoughtful analysis for picking the root vertex and ordering for the formulation. Consider a cycle C of length k, with two extra edges (a, c) and (b, d) such that both the paths on C connecting a and b (excluding a and b) contain exactly one of c and d. As shown in the figure 2.4, let the path on C between vertices d and a be called Q_1; a and b be Q_2; b and c be Q_3; c and d be Q_4.

Denote the number of edges in Q_i as l_i. Assume WLOG that $l_1 \geq l_2, l_3, l_4$ and $l_2 \geq l_4$. For some $r \in \{a, b, c, d\}$ as the root, we show that there exists a vertex $\bar{r}$ of maximum distance from r ($c(r, \bar{r}) \geq c(r, v)$ for all $v \in V$) on one of the two paths Q_i containing r. We may assume that the death time $d(\bar{r}) \geq 1$, because if $d(\bar{r}) = \frac{1}{2}$ for $r = a, b, c$ and d, then it is easy to see that our graph is necessarily K_4, which can be handled by the first case of Hamiltonian paths. Then we choose $(r, \bar{r})$ as the highest ranked terminal pair and construct a solution the same way as the previous two cases.

2.4.1 Lemma. *Let G be a cycle with two crossing chords, different from K_4. Then we can always choose a root $r \in \{a, b, c, d\}$ to define the MST instance for (LC-P) on G, such that there exists a vertex $\bar{r}$ of maximum distance from r ($c(r, \bar{r}) \geq c(r, v)$ for all $v \in V$) on one of the two paths Q_i containing r.*

Proof. We prove the above claim with many cases, by first considering $r = a$ and in the cases where the statement is not true, we show that it must be true for a different choice of r (namely, d). So suppose the statement is not true for $r = a$, i.e. suppose there is some terminal on a vertex in $Q_3 \cup Q_4 \backslash \{b, d\}$ with death time strictly greater than the death time of all terminals on vertices in $Q_1 \cup Q_2 \backslash a$. Then let s be a terminal with the maximum death time. Clearly $s \neq c$ as we assumed that the max death time is at least 1. Now consider the shortest path S_1 from a to s (of length $c(a, s)$) and consider the neighbor of s that does not lie on this path (s has degree 2, so this is uniquely defined) and call it $\tilde{s}$. Consider the shortest path S_2 from a to $\tilde{s}$. Define the circuit $C_{a,s} = \langle a, S_1, \tilde{s}, S_2 \rangle$ and note that by definition, the death time of each terminal u (except those on v) on $C_{a,s}$ is exactly half the minimum distance along this circuit. We can think of $C_{a,s}$ as the shortest circuit that contains a and s.

In addition, we know that $|E(S_2)| \leq |E(S_1)| \leq |E(S_2)| + 1$, where the first inequality holds by the maximality of the death time of s and the second holds by the minimality of S_1. Then, we can get an expression for $c(a, s)$ as follows:

$$c(a, s) = |E(S_1)| = \left\lfloor \frac{|E(S_1)| + |E(S_2)| + 1}{2} \right\rfloor = \left\lfloor \frac{|E(C_{as})|}{2} \right\rfloor .$$

Suppose $s \in Q_4$. Notice that if S_1 contains the subpath $\langle c, \ldots, s \rangle$ of Q_3, then S_2 necessarily contains the subpath $\langle \tilde{s}, \ldots, b \rangle$, and the same holds if we switch b and c in the above statement. So the path Q_3 must be contained in $C_{a,s}$. We can then enumerate all possibilities for the circuit $C_{a,s}$ (see figure 2.5):

(i) $\langle a, c, Q_4, b, Q_2 \rangle$

(ii) $\langle a, c, Q_4, Q_1 \rangle$

(iii) $\langle a, c, Q_4, b, Q_3, a \rangle$

(i) In this case, $s \in Q_3$ implies that either S_1 or S_2 contains Q_2. If S_1 contains Q_2 then since $d(a, s) \geq d(a, d)$, we have that $l_2 + 1 < |E(S_1)| \Rightarrow l_2 \leq |E(S_1)| - 2$, and $|E(S_2)| < l_4 + 1 \Rightarrow |E(S_2)| \leq l_4$. But since $l_4 \leq l_2$, we get that $|E(S_2)| \leq |E(S_1)| - 2$, a contradiction. If S_2 contains Q_2, then the same argument above holds, switching S_1 and S_2, yielding $|E(S_1)| \leq |E(S_2)| - 2$, also a contradiction.

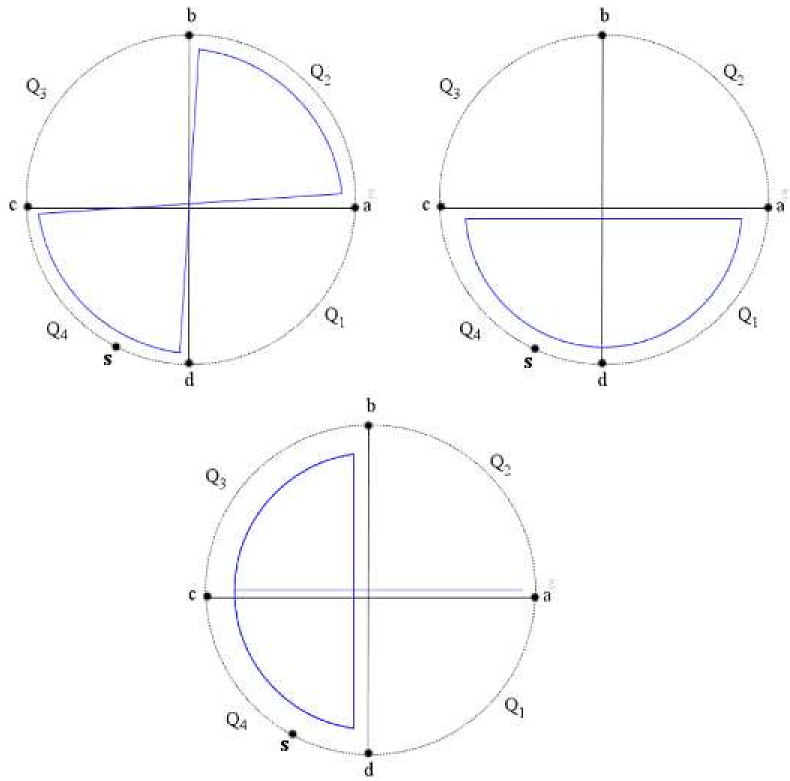

Figure 2.5: The three possibile circuits (i), (ii), (iii) (from top-left in clockwise order) for $s \in Q_4$

(ii) Similar to the previous case, either S_1 or S_2 contains Q_1. If S_1 contains Q_1 then since $d(a,s) \geq d(a,d)$, we have that $l_1 + 1 < |E(S_1)| \Rightarrow l_1 \leq |E(S_1)| - 2$, and $|E(S_2)| < l_4 + 1 \Rightarrow |E(S_2)| \leq l_4$. And $l_4 \leq l_1$ gives us that $|E(S_2)| \leq |E(S_1)| - 2$, a contradiction as before. And repeating the argument for the case when S_2 contains Q_1, we get another contradiction with $|E(S_1)| \leq |E(S_2)| - 2$.

(iii) In this case, we know that either S_1 or S_2 contains Q_3. It is easy to verify, by an argument similar to the previous two cases, that this can happen only if $l_3 < l_4$, because if $l_3 > l_4$, we would have $s \in Q_3$, and if we had equality then s would necessarily be either b or d, which we ruled out in this case. So we know that

$$c(a,s) = \left\lfloor \frac{l_3 + l_4 + 2}{2} \right\rfloor .$$

Now consider the terminal s' with maximum death time among all vertices on $Q_1 \backslash \{a\}$. If $s' = d$, then the max distance of any vertex in $Q_2 \backslash \{a\}$ is $l_2 \geq l_4$, a contradiction to the maximality of s. So $s' \neq d$ and we analogously construct a circuit $C_{as'}$, the smallest circuit containing a and s', which must contain Q_1. Since $c_1 \geq l_2 \geq l_4 > l_3$, we can conclude that $C_{as'}$ must necessarily be $\langle a, c, Q_3, d, Q_1 \rangle$, and

$$c(a,s') = \left\lfloor \frac{l_1 + l_3 + 2}{2} \right\rfloor \geq \left\lfloor \frac{l_3 + l_4 + 2}{2} \right\rfloor = c(a,s).$$

Since it must be the case that $s' \in Q_1$, this contradicts $d(s) > d(u)$ for all $u \in Q_1 \cup Q_2$.

Now that we have eliminated the possiblity of $s \in Q_4 \backslash \{c, d\}$, consider the case when $s \in Q_3 \backslash \{c, b\}$ and all vertices on $Q_1 \cup Q_2 \backslash \{a\}$ have a death time strictly smaller than $\bar{a}$. Then, as in the previous case C_{as} must contain Q_3 and again there are three possibilities for C_{as} (as shown in figure 2.6):

(i) $\langle a, c, Q_3, d, Q_1 \rangle$

(ii) $\langle a, c, Q_3, Q_2 \rangle$

(iii) $\langle a, c, Q_3, d, Q_4, a \rangle$

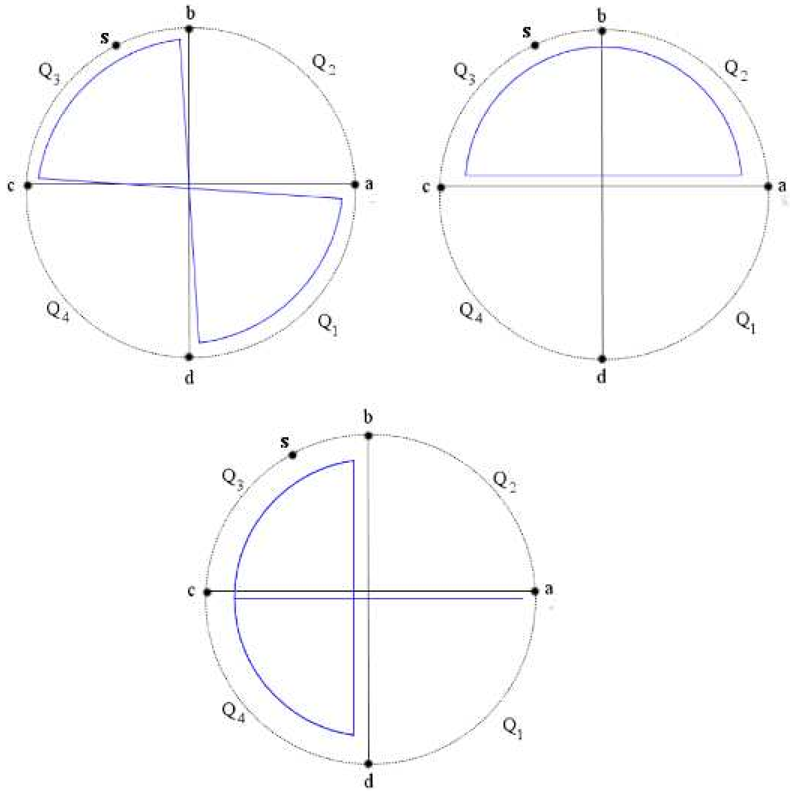

Figure 2.6: The three possibile circuits (i), (ii), (iii) (from top-left in clockwise order) for $s \in Q_3$

(i) Similar to the first subcase in the previous case, we know that either S_1 or S_2 contains Q_1. Suppose it is S_1. Then since $d(a,s) > d(a,b)$, we know that $l_1 + 1 < |E(S_1)| \Rightarrow l_1 \leq |E(S_1)| - 2$. Also, $l_3 + 1 > |E(S_2)| \Rightarrow |E(S_2)| \leq l_3$. Which gives us the contradiction that $|E(S_2)| \leq |E(S_1)| - 2$ (since $l_3 \leq l_1$). In the case S_2 contains Q_1, we get the contradiction that $|E(S_1)| \leq |E(S_2)| - 2$.

(ii) In this case, since $s \in Q_3 \backslash \{b, c\}$, we can conclude that $l_3 > l_2$, by considering the cases where either S_1 or S_2 contains Q_2, as we have done before.

Now if we change the root to d, we claim that the vertex that is furthest away from d lies on Q_1. We can compute the terminals with maximum death time lying on all Q_i, call them w_i respectively, since we know the shortest circuit that contains them must necessarily contain Q_4, as $l_1 \geq l_3 > l_2 \geq l_4$. Thus, the death times of w_i are:

$$
\begin{aligned}
c(d, w_1) &= \left\lfloor \frac{l_1 + l_4 + 1}{2} \right\rfloor && \text{on circuit } C_{dw_1} = \langle d, Q_1, c, Q_4 \rangle, \\
c(d, w_2) &= \left\lfloor \frac{l_2 + l_4 + 2}{2} \right\rfloor && \text{on circuit } C_{dw_2} = \langle d, b, Q_2, c, Q_4 \rangle, \\
c(d, w_3) &= \left\lfloor \frac{l_3 + l_4 + 1}{2} \right\rfloor && \text{on circuit } C_{dw_3} = \langle d, b, Q_3, Q_4 \rangle, \\
w_4 = c, \quad c(d, w_4) &= l_4 && \text{on circuit } C_{dw_4} = \langle d, Q_4, Q_4 \rangle.
\end{aligned}
$$

We can see that $c(d, w_1)$ is clearly the largest of the four distances and thus, clearly w_1 has the largest death time among all terminals. But as $w_1 \in Q_1$, (d, w_1) can be chosen as the top ranked pair to satisfy the property we want.

(iii) First, we claim that $l_3 > l_4$. We know that the shortest path between a and b along the circuit $C_{a,s}$ is $\langle a, c, Q_4, b \rangle$, as otherwise b would have a greater death time than s. Since this is the strictly shorter of the two paths from a to b (along $C_{a,s}$):

$$
\left\lfloor \frac{l_4 + 2}{2} \right\rfloor < \left\lfloor \frac{l_3 + 1}{2} \right\rfloor \Rightarrow l_4 < l_3.
$$

Next, we use this fact to prove that $l_3 > l_2$. Suppose not, i.e. $l_2 \geq l_3$. Then consider the terminal with the maximum death time on $Q_2 \backslash \{a\}$. If $s' = b$, then clearly

$$c(a, s') = l_2 \geq \left\lfloor \frac{l_3 + l_4 + 2}{2} \right\rfloor = c(a, s)$$

since $l_2 \geq l_3, l_4 + 1$, contradicting our assumption on s. Therefore, we may conclude that $s' \neq b$ and we can construct $C_{a,s'}$, the minimum circuit containing s' as before. Since $l_4 + 1 \leq l_1, l_2, l_3$, one can easily check that $C_{a,s'} = a - c - Q_4 - b - Q_2$ and the distance of s' from a is then

$$c(a, s') = \left\lfloor \frac{l_2 + l_4 + 2}{2} \right\rfloor \geq \left\lfloor \frac{l_3 + l_4 + 2}{2} \right\rfloor = c(a, s),$$

a contradiction to the maximality condition on s. Thus, we have shown that $l_3 > l_2$. But now as in the previous case, we can just switch the root to d with exactly the same proof. □

Now, we use this root r from the above lemma with top ranked pair $(r, \bar{r})$ to construct the solution as in the previous two cases. Let P_1 and P_2 be the paths on the cycle C between r and $\bar{r}$. Let $x_e = \frac{1}{2}$, $x_r = \frac{1}{2}$ (on the pair $(r, \bar{r})$) and $x_{\bar{r}} = \frac{1}{2}$. This solution is clearly feasible, as we have seen previously.

For the dual, we again consider the same construction: $y_{\{u\}} = \frac{1}{2}$ for all $u \in V \backslash \{r, \bar{r}\}$, $y_{Q_1} = y_{Q_2} = \frac{1}{2}$, $y_V = d(r, \bar{r})$ and $y_U = 0$ for all other $U \in \mathscr{U}$. Constraints (1.3.2) are satisfied for all edges in the cycle and constraints (1.3.3) is feasible for all terminals exactly for the same reasons as before. So we only need to check constraint (1.3.2) for the edges ac and bd. Due to our choice of $(r, \bar{r}$ from the lemma, one of P_1 or P_2 (say P_1 WLOG) contains all $\{a, b, c, d\}$ and P_2 only contains r. Therefore, the only non-zero cuts containing the edges ac and bd in the boundary are P_2 and the singletons. Note that if $\delta(P_2)$ contains one of these edges then only one singleton also contains it in its boundary, otherwise two singletons contain it. In either case constraint (1.3.2) is satisfied, concluding the proof and giving us the following result.

2.4.2 Theorem. *Let G be a cycle with two crossing chords For a given root vertex and ordering, there exists a half-integral optimal solution to (LC-P) on the unweighted MST instance defined on G.*

Chapter 3

Computational Study

In this chapter, we describe the techniques and results from a computational study of the performance of the lifted-cut relaxation. We attempted to find half-integral solutions when possible and report the integrality gap of (LC-P), as well as the corresponding integrality gap for the undirected-cut formulation (P), for comparison. We first describe an equivalent formulation of the LP in terms of network flows, similar to the flow formulations found in [5]. Then we describe the results obtained by solving the Steinlib ([8]) instances of Steiner tree problems with this LP using the commercial LP solver CPLEX ([9]). All the work in this chapter is the original work of the author.

3.1 Flow Formulation

It is easy to see that the number of constraints in the primal (LC-P) and the number of variables in the dual (LC-D) are exponential in the size of the graph, as the number of cuts are exponential in $|V|$. We now present an equivalent formulation of this problem as the union of $4k$ flow feasibility instances, with the variables x_e, x_v as common arc capacities for all the problems. We then minimize the x variables, with respect to our cost function, while preserving flow feasibility for each of the instances. Then the max-flow-min-cut theorem gives us feasibility of x for (LC-P).

We are given a graph $G = (V, E)$, a non-negative edge cost function c and a set of terminal pairs R (where each vertex has at most one terminal on it). We first construct an auxiliary digraph $\hat{G} = (\hat{V}, \hat{E})$ (see Fig. 3.1) on which we will define the flow feasibility problems. Let $R = \{(s_1, t_1), ..., (s_k, t_k)\}$, with $(s_i, t_i) \prec (s_j, t_j)$ iff $i \leq j$. As above, we extend this order to the terminals. First we add a *sink* $\hat{s}_i$ for each terminal pair (s_i, t_i). Denote the set of sinks as $\hat{S}$. For each edge $pq \in E$, we replace it by two arcs (p, q) and (q, p). We then add arcs $(s_i, \hat{s_j})$ and $(t_i, \hat{s_j})$ for each $1 \leq j \leq i$ and each $1 \leq i \leq k$ (from each terminal to the sinks of all lower ranked terminals). For notational convinience we partition $\hat{E}$ based on the three types of edges:

$$\begin{aligned}
\hat{E}_1 &= \{(p, q), (q, p) \mid pq \in E\}, \\
\hat{E}_2 &= \{(s_i, \hat{s_j}), (t_i, \hat{s_j}) \mid 1 \leq j \leq i, 1 \leq i \leq k\}, \\
\hat{E}_3 &= \{(s_i, t_i), (t_i, s_i) \mid \forall (s_i, t_i) \in R\}.
\end{aligned}$$

So,

$$\hat{V} = V \cup \bigcup_{i=1}^{k} \hat{s}_i = V \cup \hat{S},$$

$$\hat{E} = \hat{E}_1 \cup \hat{E}_2 \cup \hat{E}_3.$$

Now, for each terminal $v \in R$, we define two flow feasibility problems on $\hat{G}$, $f^{v,\mathscr{S}}$ and $f^{v,\mathscr{N}}$. If $v \in \{s_i, t_i\}$ for some terminal pair $(s_i, t_i) \in R$, then let $\hat{s}(v) = \hat{s}_i$ denote the corresponding sink. Let $f_e^{v,\mathscr{S}}$ and $f_e^{v,\mathscr{N}}$ be the flow variables denoting the flow on edge $e \in \hat{E}$ for the flow problems $f^{v,\mathscr{S}}$ and $f^{v,\mathscr{N}}$ respectively. In both problems, we then want a feasible $v - \hat{s}(v)$ flow of value 1:

$$f^{v,\mathscr{S}}(\delta^+(p)) - f^{v,\mathscr{S}}(\delta^-(p)) = \begin{cases} 1 & p = v \\ -1 & p = \hat{s}(v) \\ 0 & p \in \hat{V} \backslash \{v, \hat{s}(v)\} \end{cases} \tag{3.1.1}$$

$$f^{v,\mathscr{S}}_{(p,q)}, f^{v,\mathscr{S}}_{(q,p)} \leq x_{pq} \qquad \forall pq \in E \tag{3.1.2}$$

$$f^{v,\mathscr{S}}_{(u,\hat{s})} \leq \begin{cases} 1 & u = \bar{v}, \hat{s} = \hat{s}(v), \\ x_v & u = v, \hat{s} = \hat{s}(v), \\ 0 & \text{otherwise} \end{cases} \qquad \forall (u, \hat{s}) \in \hat{E}_2 \tag{3.1.3}$$

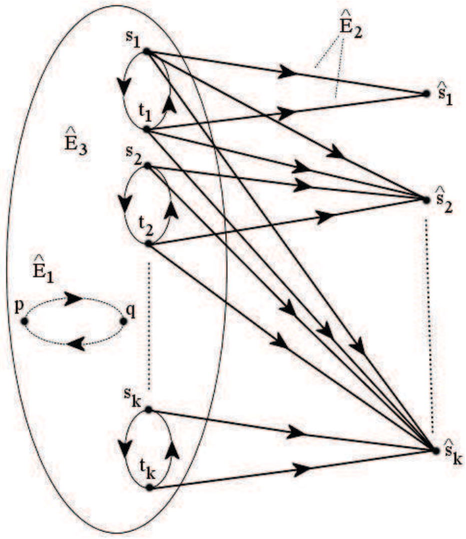

Figure 3.1: Auxiliary digraph G'

$$f^{v,\mathscr{S}}_{(u,\bar{u})} \leq \begin{cases} 1 & (u,\bar{u}) \succ (v,\bar{v}) \\ 0 & \text{otherwise} \end{cases} \qquad \forall (u,\bar{u}) \in \hat{E}_3 \tag{3.1.4}$$

$$0 \leq f^{v,\mathscr{S}}_e \qquad \forall e \in \hat{E}, \tag{3.1.5}$$

and

$$f^{v,\mathscr{N}}(\delta^+(p)) - f^{v,\mathscr{N}}(\delta^-(p)) = \begin{cases} 1 & p = v \\ -1 & p = \hat{s}(v) \\ 0 & p \in \hat{V} \backslash \{v, \hat{s}(v)\} \end{cases} \tag{3.1.6}$$

$$f^{v,\mathscr{N}}_{(p,q)}, f^{v,\mathscr{N}}_{(q,p)} \leq x_{pq} \qquad \forall pq \in E \tag{3.1.7}$$

$$f^{v,\mathscr{N}}_{(u,\hat{s})} \leq \begin{cases} x_v & u = v, \hat{s} = \hat{s}(v), \\ x_{\bar{v}} & u = \bar{v}, \hat{s} = \hat{s}(v), \\ 1 & u \succ v, u \notin \{v, \bar{v}\}, \hat{s} = \hat{s}(v) \\ 0 & \text{otherwise} \end{cases} \qquad \forall (u,\hat{s}) \in \hat{E}_2 \tag{3.1.8}$$

$$f^{v,\mathscr{N}}_{(u,\bar{u})} \leq 1 \qquad \forall (u,\bar{u}) \in \hat{E}_3 \tag{3.1.9}$$

$$0 \leq f^{v,\mathscr{N}}_e \qquad \forall e \in \hat{E}. \tag{3.1.10}$$

Thus, our flow formulation LP problem (which we will refer to as **LC-F**) is given by:

$$\text{opt}_{\text{LC-F}} = \max \quad \sum_{e\in E} c(e)x_e + \sum_{u\in R} d(v)x_u \tag{3.1.11}$$

$$\text{subject to (3.1.1)} - \text{(3.1.10)} \qquad \forall v \in R$$

$$x_e, x_u \geq 0 \quad \forall e \in E, \forall u \in V. \tag{3.1.12}$$

Note that the objective value of this LP (3.1.11) is exactly the same as that of LC-P (1.3.5).

3.1.1 Equivalence of LC-P and LC-F

We now want to show that the projection of the feasible region of LC-F onto the variables $\{x_e, x_v\}$ is exactly equal to the feasible region of LC-P. This will imply that the we solve LC-F and project the optimal solution onto $\{x_e, x_v\}$, we are guaranteed an optimal solution if LC-P. So, we need to show that some solution (f, x) is feasible for LC-F if and only if x is feasible for LC-P.

3.1.1 Theorem. *A solution $x = (x_e, x_v)$ of LC-P is feasible (for LC-P) if and only if there exists a feasible solution (f, x') of LC-F such that $x' = x$ (where x' is the projection onto the variables corresponding to x).*

Proof. Assume that there exists a feasible solution (f, x') to LC-F. We need to show that x' is feasible for LC-P. Clearly $x' \geq 0$ due to the constraint (3.1.12) on LC-F. To see that x' satisfies any constraint of the form (1.3.6) corresponding to $U \in \mathscr{S}_v$. Then U is a $v, \hat{s}(v)$-cut on the flow problem $f^{v,\mathscr{S}}$. Since, U is a Steiner cut with v reponsible, it cannot contain any terminal $v \prec u$ with $(u \neq v)$, such that $\bar{u} \notin v$. Thus, the arc $(u, \bar{u}) \notin \delta_{\hat{G}}(U)$. Also, since all arcs of $\hat{E}_2 \cap \delta_{\hat{G}}(U)$ have capacity 0, except $(v, \hat{s}(v))$ and since $\bar{v} \notin U$, the capacity of U is given by:

$$\text{capacity of } \delta_{\hat{G}}(U) = \sum_{e\in \hat{E}_1 \cap \delta_{\hat{G}}(U)} x'_e + x'_v = \sum_{e\in \delta_G(U)} x'_e + x'_v \geq 1, \tag{3.1.13}$$

by max-flow-min-cut, satisfying the constraint.

Now we look at constraints of the form (1.3.7) corresponding to cuts $U \in \mathscr{N}_v$. Again, U is a $v, \hat{s}(v)$-cut in the flow problem $f^{v,\mathscr{N}}$. Similar to above, $\delta_{\hat{G}}(U) \cap \hat{E}_1$ contains exactly one arc for each edge $e \in E \cap \delta_G(U)$ with capacity x'_e. In addition, $\delta_{\hat{G}}(U)$ contains arc $(v, \hat{s}(v))$ and $(\bar{v}, \hat{s}(v))$ with capacities x'_v and $x'_{\bar{v}}$. There are no other arcs in $\hat{E}_2 \cap \delta_{\hat{G}}(U)$ with non-zero capacity, as U contains no $(u, \bar{u}) \succ (v, \bar{v})$, because v is assumed to be responsible for U. There can be no arcs from $\hat{E}_3$ in $\delta_{\hat{G}}(U)$ as U is a non-Steiner cut by assumption. Thus,

$$\text{capacity of } \delta_{\hat{G}}(U) = \sum_{e \in \hat{E}_1 \cap \delta_{\hat{G}}(U)} x'_e + x'_v + x'_{\bar{v}} = \sum_{e \in \delta_G(U)} x'_e + x'_v + x'_{\bar{v}} \geq 1, \tag{3.1.14}$$

proving the feasibility of x' for LC-P.

Now for the other direction, assume we have some x feasible for LC-P and we need to show that there exist feasible flows $f^{v,\mathscr{S}}$ and $f^{v,\mathscr{N}}$ for each $v \in R$, with the values of x acting as the capacities wherever so defined. Suppose not, i.e. suppose there exists some $v \in R$, for which one of the flows $f^{v,\mathscr{S}}$ or $f^{v,\mathscr{N}}$ is infeasible. Then the max flow is less than 1 and by max-flow-min-cut the minimum $v, \hat{s}(v)$-cut, $\hat{U}$ has capacity less than 1. So now we have two cases:

Case: $f^{v,\mathscr{S}}$ is infeasible Note first that the condition implies that $U = \hat{U} \cap V$ must be a Steiner cut, because it cannot contain $\bar{v}$, as the arc $(\bar{v}, \hat{s}(v))$ has capacity 1. Also, if U contains some $v \prec u, (u \neq v)$, then it must necessarily contain $\bar{u}$ also, because the arc $(u, \bar{u})$ has capacity 1. This implies that $U \in \mathscr{S}_v$. Note that the capacity of $\delta_{\bar{G}}(\hat{U})$ is equal to the capacity of $\delta_{\bar{G}}(U)$. This is because the addition of any vertices of $\hat{S} \backslash \{\hat{s}(v)\}$ cannot increase or decrease the capacity of $\delta_{\hat{G}}(U)$. As in the previous direction, the capcity is given by:

$$\text{capacity of } \delta_{\hat{G}}(\hat{U}) = \sum_{e \in \hat{E}_1 \cap \delta_{\hat{G}}(U)} x_e + x_v = \sum_{e \in \delta_G(U)} x_e + x_v \geq 1, \tag{3.1.15}$$

by the feasibility of x, a contradiction.

Case: $f^{v,\mathscr{N}}$ is infeasible This time note that the condition implies that $U = \hat{U} \cap V$ must be a non-Steiner cut, because for each terminal u, the arc $(u, \bar{s}(u))$ has capacity 1, and we know the $\hat{U}$ has capacity strictly less than 1. Now notice that if $\hat{U}$ contains some terminal $u \succ v$, then $\delta_{\hat{G}}(\hat{U})$ contains the arc $(u, \hat{s}(v))$ which has capacity 1, a contradiction. As in the previous case note that the capacity of $\delta_{\bar{G}}(\hat{U})$ is equal to the capacity of $\delta_{\bar{G}}(U)$ because the addition of any vertices of $\hat{S} \backslash \{\hat{s}(v)\}$ cannot increase or decrease the capacity of $\delta_{\hat{G}}(U)$.

Therefore, we must conclude that v is responsible for U and as above, the capacity is given by:

$$\text{capacity of } \delta_{\hat{G}}(\hat{U}) = \sum_{e \in \hat{E}_1 \cap \delta_{\hat{G}}(U)} x'_e + x'_v + x'_{\bar{v}} = \sum_{e \in \delta_G(U)} x'_e + x'_v + x'_{\bar{v}} \geq 1, \tag{3.1.16}$$

by the feasibility of x, a contradiction, proving the equivalence of the two forms. □

3.2 Computational Results

3.2.1 Techniques

We implemented the construction of the digraph and the (LC-F) formulation in C++ using the LEDA [11] libraries, which we then solved using CPLEX [9]. Solving for (LC-F), instead of (LC-P) directly meant that we were guaranteed to find optimal, but not necessarily basic solutions to (LC-P). However, a separation oracle for the (LC-P) problem would be essentially equivalent to (LC-F) and while the above flow formulation is polynomial in size of the graph, we add $O(k^2)$ edges to the auxiliary graph and define $O(k)$ flow instances on that. So especially for larger values of k (particularly, in minimum spanning tree instances, where $k = n-1$), (LC-F) alone is a large LP.

In addition, the question we were interested in the existence of a half-integral solution. Note that the existence of a non-half-integral extreme point in the (LC-P) polytope does not necessarily imply that there exists a cost function c for which it is the unique minimum. This is because, the vector c alone does not define the objective value of (LC-P). The death times $d(v)$ are completely determined by c and the graph G and cannot be set arbitrarily by us. In addition changing c could change the ordering of the terminal pairs and therefore the entire polytope. However, non-half-integral extreme points do suggest problems in designing purely LP rounding based approximation algorithms. Since testing all possible half-integral solutions and checking if they are feasible and optimal was computationally infeasible, if we found a non-half-integral extreme point, we terminated and moved on to the next choice of root vertex. We used the following simple rounding algorithm to look for a half-integral solution, or to verify the existence of a non-half-integral extreme point.

Given an optimal solution x to (LC-F). Repeat while x is not half-integral. If $\exists x_e$ (or x_v) s.t. $x_e < 0.5(x_v < 0.5)$, then add constraint $x_e = 0(x_v = 0)$ and resolve LP. Else, if $\exists x_e$ or x_v s.t. $x_e > 0.5(x_v > 0.5)$, then add constraint $x_e = 1(x_v = 1)$ and resolve LP. If the objective value increases after the addition of one of these constraints, we know that (LC-P) must have a non-half-integral extreme point.

If the original x were a convex combination of half-integral points $x^1, \ldots, x^p$, then $x_e < 0.5$ (or $x_v < 0.5$) implies that at least one of $x^1, \ldots, x^p$ must have $x_e^i = 0$ (or $x_v^i = 0$). Similarly, $x_e > 0.5$ (or $x_v > 0.5$) implies that at least one of $x^1, \ldots, x^p$ must have $x_e^i = 1$ (or $x_v^i = 1$). Thus, repeating this process will eventually lead to a half-integral optimal solution, or adding one of these constraints will increase the cost of the optimal solution, implying that x was not a convex combination of half-integral extreme points.

We repeated this procedure for all choices of roots, as we considered only Steiner tree and spanning tree instances, until a half-integral solution was found. If one was not found, we enumerated all possible ties in the ordering of terminal pairs for each choice of root and solved for alternate orderings.

3.2.2 The Test Sets

First, we solved unit weight minimum spanning tree instances on all graphs of order 8 and less. We enumerated these graphs, using the *geng* utility from the *nauty* package by McKay [10]. We were able to find half-integral solutions for all graphs under some root and some ordering.

We then ran instances from the Steinlib ([8]) test sets for Steiner Tree instances. We also implemented the flow formulation for the undirected cut relaxation (from [5]) in C++ using LEDA libraries and solved the same Steinlib instances using this, to compare the performance of the two relaxations in terms of integrality gap. In the tables in the following section, we list the size of the underlying graph, as well as the number of terminal pairs in the instance, along with the objective value of the half-integral optimal solution found and the IP/LP gap (the costs of the optimal Steiner trees were given in all instances). We also give the optimal for the undirected cut relaxation, and its IP/LP gap for the sake of comparison.

An interesting result of the computational study was that whenever we did not find an optimal half-integral solution, we always managed to find an optimal quarter-integral solution. We indicate such instances with “None Found” in the optimal half-intergral objective column of the tables and then separately list the computational results for quarter-integral optimal solutions in these instances in Table 3.18.

Some instances in some of the test sets were too large and the converge in CPLEX was too slow and were thus abandoned. Therefore, we do not list all the test sets or may not list the results for some problem in some of the sets. In total, we were unable to compute results for 30 out of 264 instances in the test sets we considered. In addition, we did not consider 24 test sets with 811 additional instances, including many instances where the cost of the optimal Steiner tree was not known.

3.2.3 Results

We present the results that we computed for the following Steinlib test sets. The following tables are arranged by the following columns:

- *Instance* - name of the test instance
- $|V|$ - number of vertices in the underlying graph
- $|E|$ - number of edges in the underlying graph
- $|R|$ - number of terminals (not pairs, but individual terminal vertices)
- 1/2-*int. Obj.* - (LC-P) objective value of the first optimal half-integral solution found (we refer to this value as opt^*_{LC-P}) as we iterated through all possible choices of roots and orderings. If we could not find an optimal half-integral solution for any root or ordering through our rounding procedure, we write 'None Found'
- *Optimal* - cost of the optimal Steiner tree, opt_R
- *Gap* - IP/LP gap for the solution reported in 1/2-int. Obj., i.e., opt_R/opt^*_{LC-P}
- *Undir. Obj.* - optimal objective value for the undirected-cut relaxation, opt_P
- *(P) Gap* - IP/LP gap for the optimal undirected cut solution, i.e., opt_R/opt_P
- *Improvement* - percentage improvement of the LP/IP gap of (LC-P) over (P), i.e.,

$$\frac{opt^*_{LC-P} - opt_P}{opt_R} * 100$$

In Table 3.18, where we list the instances where we were not able to find optimal half-integral solutions, the column 1/4-*int. Obj.* lists the (LC-P) objective value of the first optimal quarter-integral solution found and the columns *Gap* and *Improvement* are defined appropriately.

Instance	$\|V\|$	$\|E\|$	$\|R\|$	1/2-int. Obj.	Optimal	Gap	Undir. Obj.	(P) Gap	Improvement
ES10FST01	18	20	18	21315435	22920745	1.08	17352482	1.32	17.29%
ES10FST02	14	13	18	19134104	19134104	1.00	19134104	1.00	0.00%
ES10FST03	17	20	18	23253661	26003678	1.12	21273648	1.22	7.61%
ES10FST04	18	20	18	19469126	20461116	1.05	19350821	1.06	0.58%
ES10FST05	12	11	18	18818916	18818916	1.00	18818916	1.00	0.00%
ES10FST06	17	20	18	24467461	26540768	1.08	20131434	1.32	16.34%
ES10FST07	14	13	18	26025072	26025072	1.00	26025072	1.00	0.00%
ES10FST08	21	28	18	22124912	25056214	1.13	18402047	1.36	14.86%
ES10FST09	21	29	18	19203779	22062355	1.15	16668809	1.32	11.49%
ES10FST10	18	21	18	23769658	23936095	1.01	22944932	1.04	3.45%
ES10FST11	14	13	18	22239535	22239535	1.00	22239535	1.00	0.00%
ES10FST12	13	12	18	19626318	19626318	1.00	19626318	1.00	0.00%
ES10FST13	18	21	18	19483914	19483914	1.00	16525930	1.18	15.18%
ES10FST14	24	32	18	21046882	21856128	1.04	18546733	1.18	11.44%
ES10FST15	16	18	18	18296778	18641924	1.02	15659093	1.19	14.15%

Table 3.1: Test set ES10FST

Instance	$\|V\|$	$\|E\|$	$\|R\|$	1/2-int. Obj.	Optimal	Gap	Undir. Obj.	(P) Gap	Improvement
ES20FST01	29	28	38	33703886	33703886	1.00	33703886	1.00	0.00%
ES20FST02	29	28	38	32639486	32639486	1.00	32639486	1.00	0.00%
ES20FST03	27	26	38	27847417	27847417	1.00	27847417	1.00	0.00%
ES20FST04	57	83	38	23169798	27624394	1.19	20593828	1.34	9.32%
ES20FST05	54	77	38	28125710	34033163	1.21	23476575	1.45	13.66%
ES20FST06	29	28	38	36014241	36014241	1.00	36014241	1.00	0.00%
ES20FST07	45	59	38	27619945	34934874	1.26	24008408	1.46	10.34%
ES20FST08	52	74	38	30882544.5	38016346	1.23	26062921	1.46	12.68%
ES20FST09	36	42	38	32950420.5	36739939	1.12	26689958	1.38	17.04%
ES20FST10	49	67	38	29339650.5	34024740	1.16	23993266	1.42	15.71%
ES20FST11	33	36	38	26905801	27123908	1.01	25714785	1.05	4.39%
ES20FST12	33	36	38	25803445.5	30451397	1.18	21096971	1.44	15.46%
ES20FST13	35	40	38	34063688	34438673	1.01	32495741	1.06	4.55%
ES20FST14	36	44	38	29318085	34062374	1.16	25603540	1.33	10.91%
ES20FST15	37	43	38	31503334	32303746	1.03	30467282	1.06	3.21%

Table 3.2: Test set ES20FST

Instance	$\|V\|$	$\|E\|$	$\|R\|$	1/2-int. Obj.	Optimal	Gap	Undir. Obj.	(P) Gap	Improvement
ES30FST01	79	115	58	33711897	40692993	1.21	30011047	1.36	9.09%
ES30FST02	71	97	58	33223279	40900061	1.23	31105804	1.31	5.18%
ES30FST03	83	120	58	37155773	43120444	1.16	33866900	1.27	7.63%
ES30FST04	80	115	58	35018704.5	42150958	1.20	29015574	1.45	14.24%
ES30FST05	58	71	58	34926884	41739748	1.20	29605537	1.41	12.75%
ES30FST06	83	119	58	29843261	39955139	1.34	27249251	1.47	6.49%
ES30FST07	53	64	58	37173375	43761391	1.18	32572973	1.34	10.51%
ES30FST08	69	93	58	34706656	41691217	1.20	30346305	1.37	10.46%
ES30FST09	43	44	58	31797611	37133658	1.17	25516919	1.46	16.91%
ES30FST10	48	52	58	41816723	42686610	1.02	40090641	1.06	4.04%
ES30FST11	79	112	58	33871930	41647993	1.23	30537220.5	1.36	8.01%
ES30FST12	46	48	58	36383266	38416720	1.06	35635052	1.08	1.95%
ES30FST13	65	84	58	33873481	37406646	1.10	28149408	1.33	15.30%
ES30FST14	53	58	58	42755314	42897025	1.00	42623312	1.01	0.31%
ES30FST15	118	188	58	32821263	43035576	1.31	29599985	1.45	7.49%

Table 3.3: Test set ES30FST

Instance	$\|V\|$	$\|E\|$	$\|R\|$	1/2-int. Obj.	Optimal	Gap	Undir. Obj.	(P) Gap	Improvement
ES40FST01	93	127	78	35216632	44841522	1.27	32418137	1.38	6.24%
ES40FST02	82	105	78	38965960	46811310	1.20	34266438	1.37	10.04%
ES40FST03	87	116	78	38238180	49974157	1.31	34328323.5	1.46	7.82%
ES40FST04	55	55	78	44637601	45289864	1.01	43545704	1.04	2.41%
ES40FST05	121	180	78	40833022.5	51940413	1.27	34663163	1.50	11.88%
ES40FST06	92	123	78	43011514.5	49753385	1.16	38314195	1.30	9.44%
ES40FST07	77	95	78	41718596	45639009	1.09	40870065	1.12	1.86%
ES40FST08	98	137	78	38775163	48745996	1.26	34052834	1.43	9.69%
ES40FST09	107	153	78	40647678	51761789	1.27	34323578	1.51	12.22%
ES40FST10	107	152	78	44332256	57136852	1.29	37656718.5	1.52	11.68%
ES40FST11	97	135	78	36428572	46734214	1.28	31104432.5	1.50	11.39%
ES40FST12	67	75	78	40222166	43843378	1.09	38645053	1.13	3.60%
ES40FST13	78	95	78	49118268	51884545	1.06	45363399	1.14	7.24%
ES40FST14	98	134	78	39311275.5	49166952	1.25	34762114.5	1.41	9.25%
ES40FST15	93	129	78	44695923	50828067	1.14	39963737.5	1.27	9.31%

Table 3.4: Test set ES40FST

Instance	$\|V\|$	$\|E\|$	$\|R\|$	1/2-int. Obj.	Optimal	Gap	Undir. Obj.	(P) Gap	Improvement
ES50FST01	118	160	98	46039276.5	54948660	1.19	37379635	1.47	15.76%
ES50FST02	125	177	98	42496774	55484245	1.31	36116073	1.54	11.50%
ES50FST03	128	182	98	43565984.5	54691035	1.26	39137766.5	1.40	8.10%
ES50FST04	106	138	98	45535736	51535766	1.13	41304397	1.25	8.21%
ES50FST05	104	135	98	46843447.5	55186015	1.18	43529567.5	1.27	6.00%
ES50FST06	126	182	98	42946564.5	55804287	1.30	38643804.5	1.44	7.71%
ES50FST07	143	211	98	39598577.5	49961178	1.26	35576869.5	1.40	8.05%
ES50FST08	83	96	98	50243920	53754708	1.07	50045608	1.07	0.37%
ES50FST09	139	202	98	42823306	53456773	1.25	39189756	1.36	6.80%
ES50FST10	139	207	98	41479316.5	54037963	1.30	38222693	1.41	6.03%
ES50FST11	100	131	98	44988757	52532923	1.17	38784567.5	1.35	11.81%
ES50FST12	110	149	98	None Found	53409291	N/A	35224299.5	1.52	N/A
ES50FST13	92	116	98	42324481	53891019	1.27	39857486	1.35	4.58%
ES50FST14	120	167	98	42033840.5	53551419	1.27	38227440	1.40	7.11%
ES50FST15	112	147	98	45532297	52180862	1.15	40649933.5	1.28	9.36%

Table 3.5: Test set ES50FST

Instance	$\|V\|$	$\|E\|$	$\|R\|$	1/2-int. Obj.	Optimal	Gap	Undir. Obj.	(P) Gap	Improvement
ES60FST01	123	159	118	43632205	53761423	1.23	37486867	1.43	11.43%
ES60FST02	186	280	118	44283979.5	55367804	1.25	40852138.5	1.36	6.20%
ES60FST03	113	142	118	49416145.5	56566797	1.14	43274591	1.31	10.86%
ES60FST04	162	238	118	42875066.5	55371042	1.29	39191853.5	1.41	6.65%
ES60FST05	119	148	118	43100064	54704991	1.27	38391545.5	1.42	8.61%
ES60FST06	130	174	118	49520323	60421961	1.22	46487328.5	1.30	5.02%
ES60FST07	188	280	118	46457940	58978041	1.27	44471211.5	1.33	3.37%
ES60FST08	109	133	118	46621481	58138178	1.25	41581728.5	1.40	8.67%
ES60FST09	151	216	118	43799411	55877112	1.28	40948619	1.36	5.10%
ES60FST10	133	177	118	47550901.5	57624488	1.21	43002107	1.34	7.89%
ES60FST11	121	154	118	47110672.5	56141666	1.19	42799453	1.31	7.68%
ES60FST12	176	257	118	49590635.5	59791362	1.21	44521172	1.34	8.48%
ES60FST13	157	226	118	46073332.5	61213533	1.33	39552251	1.55	10.65%
ES60FST14	118	149	118	49330757.5	56035528	1.14	48640013	1.15	1.23%
ES60FST15	117	151	118	44321722.5	56622581	1.28	41634001	1.36	4.75%

Table 3.6: Test set ES60FST

Instance	$\|V\|$	$\|E\|$	$\|R\|$	1/2-int. Obj.	Optimal	Gap	Undir. Obj.	(P) Gap	Improvement
ES70FST01	154	209	138	53823779	62058863	1.15	48717090	1.27	8.23%
ES70FST02	147	197	138	46966479.5	60928488	1.30	38978895	1.56	13.11%
ES70FST03	181	264	138	50633555	61934664	1.22	44082704	1.40	10.58%
ES70FST04	167	231	138	56011177.5	62938583	1.12	51366338	1.23	7.38%
ES70FST05	169	231	138	47944585.5	62256993	1.30	42738019	1.46	8.36%
ES70FST06	187	268	138	52238427	62124528	1.19	47255253	1.31	8.02%
ES70FST07	167	230	138	54849021.5	62223666	1.13	48677762.5	1.28	9.92%
ES70FST08	209	314	138	47811096.5	61872849	1.29	40054055.5	1.54	12.54%
ES70FST09	161	220	138	49636636.5	62986133	1.27	43367517	1.45	9.95%
ES70FST10	165	225	138	48244618.5	62511830	1.30	40977007	1.53	11.63%
ES70FST11	177	254	138	51587833	66455760	1.29	44565358.5	1.49	10.57%
ES70FST12	142	181	138	56234038.5	63047132	1.12	51242369.5	1.23	7.92%
ES70FST13	160	219	138	51968348.5	62912258	1.21	46190036.5	1.36	9.18%
ES70FST14	143	184	138	50360234	60411124	1.20	43016353	1.40	12.16%
ES70FST15	178	251	138	50893375	62318458	1.22	44685849.5	1.39	9.96%

Table 3.7: Test set ES70FST

Instance	$\|V\|$	$\|E\|$	$\|R\|$	1/2-int. Obj.	Optimal	Gap	Undir. Obj.	(P) Gap	Improvement
ES80FST01	187	255	158	58206511	70927442	1.22	49701775	1.43	11.99%
ES80FST02	183	249	158	None Found	65273810	N/A	45239625	1.44	N/A
ES80FST03	189	261	158	50243139.5	65332546	1.30	48640519.5	1.34	2.45%
ES80FST04	198	280	158	49563927	64193446	1.30	46027375	1.39	5.51%
ES80FST05	172	228	158	54834413	66350529	1.21	46797698	1.42	12.11%
ES80FST06	172	224	158	55495024.5	71007444	1.28	49089523.5	1.45	9.02%
ES80FST07	193	271	158	51404973	68228475	1.33	47285182.5	1.44	6.04%
ES80FST08	217	306	158	51466172	67452377	1.31	47588990	1.42	5.75%
ES80FST09	236	343	158	54541631	69825651	1.28	46878391	1.49	10.97%
ES80FST10	156	197	158	54566863.5	65497988	1.20	47455325	1.38	10.86%
ES80FST11	209	295	158	51621068.5	66283099	1.28	45536836.5	1.46	9.18%
ES80FST12	147	180	158	50748856	65070089	1.28	48893327.5	1.33	2.85%
ES80FST13	164	211	158	53269818	68022647	1.28	50576264	1.34	3.96%
ES80FST14	209	297	158	None Found	70077902	N/A	48800427	1.44	N/A
ES80FST15	197	282	158	55759909	69939071	1.25	50461722.25	1.39	7.58%

Table 3.8: Test set ES80FST

Instance	$\|V\|$	$\|E\|$	$\|R\|$	1/2-int. Obj.	Optimal	Gap	Undir. Obj.	(P) Gap	Improvement
ES90FST01	181	231	178	63678693	68350357	1.07	63100249	1.08	0.85%
ES90FST02	221	313	178	53847761	71294845	1.32	48037843	1.48	8.15%
ES90FST03	284	430	178	54419752.5	74817473	1.37	48281159	1.55	8.20%
ES90FST04	217	299	178	52837842.5	70910063	1.34	46878168	1.51	8.40%
ES90FST05	190	254	178	None Found	71831224	N/A	53579279.5	1.34	N/A
ES90FST06	215	290	178	58881560	68640346	1.17	56175003.5	1.22	3.94%
ES90FST07	175	221	178	57439950.5	72036885	1.25	52451282	1.37	6.93%
ES90FST08	234	332	178	None Found	72341668	N/A	47156067.5	1.53	N/A
ES90FST09	234	331	178	51664636.5	67856007	1.31	44420114.5	1.53	10.68%
ES90FST10	246	356	178	55523451.5	72310409	1.30	50383627.2	1.44	7.11%
ES90FST11	225	323	178	53855825	72310039	1.34	47936750.5	1.51	8.19%
ES90FST12	207	284	178	49215790	69367257	1.41	47065932	1.47	3.10%
ES90FST13	240	349	178	53012657.5	72810663	1.37	47710865.75	1.53	7.28%
ES90FST14	185	243	178	56034321	69188992	1.23	52214018.5	1.33	5.52%
ES90FST15	207	286	178	55066661.5	71778294	1.30	48352119	1.48	9.35%

Table 3.9: Test set ES90FST

Instance	$\|V\|$	$\|E\|$	$\|R\|$	1/2-int. Obj.	Optimal	Gap	Undir. Obj.	(P) Gap	Improvement
ES100FST01	250	354	198	52055234.5	72522165	1.39	47036581.5	1.54	6.92%
ES100FST02	339	522	198	None Found	75176630	N/A	51318478.5	1.46	N/A
ES100FST03	189	233	198	59086166	72746006	1.23	53398490.5	1.36	7.82%
ES100FST04	188	235	198	61560110	74342392	1.21	56913866	1.31	6.25%
ES100FST05	188	238	198	67349573.5	75670198	1.12	63747077	1.19	4.76%
ES100FST06	301	452	198	54793391	74414990	1.36	49844261	1.49	6.65%
ES100FST07	276	401	198	57259103.5	77740576	1.36	55248165	1.41	2.59%
ES100FST08	210	276	198	57450211.5	73033178	1.27	49429636.5	1.48	10.98%
ES100FST09	248	342	198	None Found	77952027	N/A	55263427.5	1.41	N/A
ES100FST10	229	312	198	59071191	75952202	1.29	51494351.5	1.47	9.98%
ES100FST11	253	362	198	61602444	78674859	1.28	56071849.5	1.40	7.03%
ES100FST12	266	385	198	58856798	76131099	1.29	52548592	1.45	8.29%
ES100FST13	254	361	198	None Found	74604990	N/A	53491107	1.39	N/A
ES100FST14	198	253	198	63732138.5	78632795	1.23	56248728	1.40	9.52%
ES100FST15	231	319	198	54255776	70446493	1.30	46804585	1.51	10.58%

Table 3.10: Test set ES100FST

Instance	$\|V\|$	$\|E\|$	$\|R\|$	1/2-int. Obj.	Optimal	Gap	Undir. Obj.	(P) Gap	Improvement
MSM0580	338	541	20	390	467	1.20	337	1.39	11.35%
MSM0654	1290	2270	18	769	823	1.07	756	1.09	1.58%
MSM0709	1442	2403	30	801	884	1.10	721	1.23	9.05%
MSM0920	752	1264	50	686	806	1.17	594	1.36	11.41%
MSM1008	402	695	20	454	494	1.09	384	1.29	14.17%
MSM1234	933	1632	24	537	550	1.02	508.5	1.08	5.18%
MSM1477	1199	2078	60	854	1068	1.25	786	1.36	6.37%
MSM1707	278	478	20	554	564	1.02	534	1.06	3.55%
MSM1844	90	135	18	168	188	1.12	145	1.30	12.23%
MSM1931	875	1522	18	571	604	1.06	545	1.11	4.30%
MSM2000	898	1562	18	527	594	1.13	527	1.13	0.00%
MSM2326	418	723	26	378	399	1.06	312	1.28	16.54%
MSM3676	957	1554	18	569	607	1.07	530	1.15	6.43%
MSM4038	237	390	20	316	353	1.12	290	1.22	7.37%
MSM4114	402	690	30	373	393	1.05	343	1.15	7.63%
MSM4190	391	666	30	341	381	1.12	321	1.19	5.25%
MSM4224	191	302	20	276	311	1.13	250	1.24	8.36%
MSM4414	317	476	20	362	408	1.13	347	1.18	3.68%
MSM4515	777	1358	24	555	630	1.14	458	1.38	15.40%

Table 3.11: Test set MSM

Instance	$\lvert V\rvert$	$\lvert E\rvert$	$\lvert R\rvert$	1/2-int. Obj.	Optimal	Gap	Undir. Obj.	(P) Gap	Improvement
P455	100	4950	8	989.5	1138	1.15	872.5	1.30	10.28%
P456	100	4950	8	1058	1228	1.16	932	1.32	10.26%
P457	100	4950	18	1293	1609	1.24	1113.5	1.44	11.16%
P458	100	4950	18	1448.5	1868	1.29	1256.5	1.49	10.28%
P459	100	4950	38	1785	2345	1.31	1543.5	1.52	10.30%
P460	100	4950	38	2049	2959	1.44	1783.5	1.66	8.97%

Table 3.12: Test set P4E

Instance	$\lvert V\rvert$	$\lvert E\rvert$	$\lvert R\rvert$	1/2-int. Obj.	Optimal	Gap	Undir. Obj.	(P) Gap	Improvement
P601	100	180	8	7181	8083	1.13	6444	1.25	9.12%
P602	100	180	8	4600.5	5022	1.09	4233.5	1.19	7.31%
P603	100	180	18	10517.5	11397	1.08	9164	1.24	11.88%
P604	100	180	18	10355	10355	1.00	9083	1.14	12.28%
P605	100	180	18	12518.5	13048	1.04	11251	1.16	9.71%
P606	100	180	38	12558	15358	1.22	10882.5	1.41	10.91%
P607	100	180	38	12293.5	14439	1.17	10868.5	1.33	9.87%
P608	100	180	38	13889	18263	1.31	11757	1.55	11.67%
P609	100	180	98	22951.5	30161	1.31	20465.25	1.47	8.24%
P610	100	180	98	20264	26903	1.33	17841.5	1.51	9.00%
P611	100	180	98	22344	30258	1.35	20344	1.49	6.61%
P612	200	370	18	14320	18429	1.29	13805.5	1.33	2.79%
P613	200	370	38	22721.5	27276	1.20	20735	1.32	7.28%
P614	200	370	78	31499.5	42474	1.35	28052.75	1.51	8.11%
P615	200	370	198	None Found	62263	N/A	40615.0625	1.53	N/A

Table 3.13: Test set P6Z

Instance	$\lvert V\rvert$	$\lvert E\rvert$	$\lvert R\rvert$	1/2-int. Obj.	Optimal	Gap	Undir. Obj.	(P) Gap	Improvement
P619	100	180	8	7378	7485	1.01	7378	1.01	0.00%
P620	100	180	8	8746	8746	1.00	7015	1.25	19.79%
P621	100	180	8	8346	8688	1.04	8346	1.04	0.00%
P622	100	180	18	14814	15972	1.08	12941	1.23	11.73%
P623	100	180	18	17016	19496	1.15	16059	1.21	4.91%
P624	100	180	38	16845	20246	1.20	13751	1.47	15.28%
P625	100	180	38	19208.5	23078	1.20	14616	1.58	19.90%
P626	100	180	38	18099.5	22346	1.23	15508	1.44	11.60%
P627	100	180	98	28477	40647	1.43	23982.5	1.69	11.06%
P628	100	180	98	29287	40008	1.37	24237	1.65	12.62%
P629	100	180	98	29798	43287	1.45	25259	1.71	10.49%
P630	200	370	18	25316	26125	1.03	20148	1.30	19.78%
P631	200	370	38	33936	39067	1.15	26510	1.47	19.01%
P632	200	370	78	44665	56217	1.26	37202	1.51	13.28%
P633	200	370	198	None Found	86268	N/A	52174	1.65	N/A

Table 3.14: Test set P6E

Instance	$\lvert V\rvert$	$\lvert E\rvert$	$\lvert R\rvert$	1/2-int. Obj.	Optimal	Gap	Undir. Obj.	(P) Gap	Improvement
Berlin52	52	1326	30	805	1044	1.30	684.5	1.53	11.54%
Brasil58	58	1653	48	11421	13655	1.20	10099	1.35	9.68%

Table 3.15: Test set X

Instance	$\|V\|$	$\|E\|$	$\|R\|$	1/2-int. Obj.	Optimal	Gap	Undir. Obj.	(P) Gap	Improvement
B01	50	63	16	75.5	82	1.09	72	1.14	4.27%
B02	50	63	24	75.5	83	1.10	72.5	1.14	3.61%
B03	50	63	48	127	138	1.09	124.5	1.11	1.81%
B04	50	100	16	52	59	1.13	49.5	1.19	4.24%
B05	50	100	24	52	61	1.17	49.5	1.23	4.10%
B06	50	100	48	95	122	1.28	89	1.37	4.92%
B07	75	94	24	99	111	1.12	96.5	1.15	2.25%
B08	75	94	36	95	104	1.09	82.5	1.26	12.02%
B09	75	94	74	196.5	220	1.12	194.5	1.13	0.91%
B10	75	150	24	77	86	1.12	71.5	1.20	6.40%
B11	75	150	36	72.5	88	1.21	70	1.26	2.84%
B12	75	150	74	131	174	1.33	128	1.36	1.72%
B13	100	125	32	142	165	1.16	137	1.20	3.03%
B14	100	125	48	199	235	1.18	198	1.19	0.43%
B15	100	125	98	264.5	318	1.20	249.25	1.28	4.80%
B16	100	200	32	106	127	1.20	103.5	1.23	1.97%
B17	100	200	48	108	131	1.21	102.5	1.28	4.20%
B18	100	200	98	170.5	218	1.28	167.5	1.30	1.38%

Table 3.16: Test set B

Instance	$\|V\|$	$\|E\|$	$\|R\|$	1/2-int. Obj.	Optimal	Gap	Undir. Obj.	(P) Gap	Improvement
C01	500	625	8	74.5	85	1.14	71	1.20	4.12%
C02	500	625	18	113	144	1.27	108.5	1.33	3.13%
C03	500	625	164	598.5	754	1.26	579.5	1.30	2.52%
C04	500	625	248	864.5	1079	1.25	854.75	1.26	0.90%
C05	500	625	498	1285	1579	1.23	1270.5	1.24	0.92%
C06	500	1000	8	47	55	1.17	45.5	1.21	2.73%
C07	500	1000	18	83.5	102	1.22	83	1.23	0.49%
C08	500	1000	164	387.5	509	1.31	379.25	1.34	1.62%
C09	500	1000	248	516	707	1.37	507	1.39	1.27%

Table 3.17: Test set C

Instance	$\|V\|$	$\|E\|$	$\|R\|$	1/4-int. Obj.	Optimal	Gap	Undir. Obj.	(P) Gap	Improvement
ES50FST12	110	149	98	43476772	53409291	1.23	35224299.5	1.52	15.45%
ES80FST02	183	249	158	51174464	65273810	1.28	45239625	1.44	9.09%
ES80FST14	209	297	158	55069706.5	70077902	1.27	48800427	1.44	8.95%
ES90FST05	190	254	178	53579279.5	71831224	1.34	58221148.5	1.23	6.46%
ES90FST08	234	332	178	47156067.5	72341668	1.53	55722313	1.30	11.84%
ES100FST02	339	522	198	57727090.5	75176630	1.30	51318478.5	1.46	8.53%
ES100FST09	248	342	198	60272648.5	77952027	1.29	55263427.5	1.41	6.43%
ES100FST13	254	361	198	57726076	74604990	1.29	53491107	1.39	5.68%
P633	200	370	198	61978.5	86268	1.39	52174	1.65	11.37%
P615	200	370	198	44292.25	62263	1.41	40615.0625	1.53	5.91%

Table 3.18: Quarter-Integral Optimal Solutions

Chapter 4

Conclusions and Further Directions

In this report we looked at the lifted-cut linear programming relaxation for the Steiner forest problems and its performance, particularly with regard to the existence of half-integral optimal solutions. We constructed some classes of graphs on which the unweighted minimum spanning tree instance has a half-integral optimal solution for a given choice of root and ordering of the terminal pairs. It is an interesting question to identify instances of the problem where we are guaranteed half-integral optimal solutions, even if we restrict ourselves to the simple case unweighted minimum spanning tree instances.

We then solved the LP for test instances from the Steinlib suite using a compact flow formulation. The unweighted minimum spanning tree instances on all connected graphs of order at most 8 had half-integral optimal solutions. Out of 234 Steiner tree instances we successfully solved on larger graphs, we were unable to find half-integral optimal solutions for only 10, and in doing so we verified (computationally) that for every possible choice of root and ordering on these instances, there exists at least one non-half-integral extreme point of the polytope. In addition, comparing the IP/LP gap with that of the undirected-cut relaxation suggests that the lifted-cut relaxation has a consistent significant improvement in the objective value.

Bibliography

[1] A. Agarwal, P. Klein, and R. Ravi, *When trees collide: An approximation algorithm for the generalized Steiner problem in networks*, SIAM Journal on Computing, 24 (1995), pp. 445-456.

[2] J. Könemann, S. Leonardi, G. Schäfer, *A group-strategyproof mechanism for Steiner forests*, In Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), ACM Press, 2005, pp. 612619.

[3] J. Könemann, S. Leonardi, G. Schäfer, and S. van Zwam, *From primal-dual to cost shares and back: A stronger LP relaxation for the Steiner forest problem*, In Proceedings of the 32nd International Colloquium on Automata, Languages and Programming (ICALP), 2005.

[4] M. Goemans, *The Steiner Tree Polytope and Related Polyhedra*, Mathematical Programming, 63 (1994), pp. 157-182.

[5] M. Goemans, Y. Myung, *A Catalog of Steiner Tree Formulations*, Networks, 23 (1992), pp. 19-28.

[6] K. Jain, V. Vazirani, *Applications of Approximation Algorithms to Cooperative Games*, In Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing (STOC), ACM Press, 2001, pp. 364-372.

[7] V. Vazirani, *Approximation Algorithms*, Springer, 2001, New York.

[8] T. Koch, A. Martin, and S. Voss, *SteinLib: An Updated Library on Steiner Tree Problems in Graphs*, 2000. Online document at `http://elib.zib.de/steinlib`, last visited on July 10, 2007.

[9] ILOG CPLEX, `http://www.ilog.com/products/cplex/`.

[10] B. McKay, *nauty User's Guide (Version 2.2)*, Computer Science Department, Australian National University, 2004.

[11] K. Mehlhorn , S. Näher, *LEDA: a platform for combinatorial and geometric computing*, Communications of the ACM, v.38 n.1, pp. 96-102, Jan. 1995