

Compacting Cuts: A New Linear Formulation for Minimum Cut

ROBERT D. CARR

Sandia National Laboratories

GORAN KONJEVOD

Arizona State University

GREG LITTLE

Massachusetts Institute of Technology

VENKATESH NATARAJAN

Carnegie Mellon University

AND

OJAS PAREKH

Emory University

27

G. Konjevod's work was supported in part by NSF grant CCR-0209138.

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under contract DE-AC04-94AL85000.

Authors' addresses: R. D. Carr, Discrete Math. Group, Sandia National Labs, Albuquerque, NM 87185, e-mail: rdcarr@sandia.gov; G. Konjevod, School of Computing and Informatics, Arizona State University, Tempe, AZ 85287-8809, e-mail: goran@asu.edu; G. Little, CSAIL, Massachusetts Institute of Technology, Cambridge, MA 02139, e-mail: glittle@gmail.com; V. Natarajan, Department of Mathematical Sciences, Carnegie Mellon University, Pittsburgh, PA 15213, e-mail: vn@andrew.cmu.edu; O. Parekh, Mathematics and Computer Science Department, Emory University, Atlanta, GA 30322, e-mail: ojas@mathcs.emory.edu.

©2009 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the [U.S.] Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org.

© 2009 ACM 1549-6325/2009/07-ART27 \$10.00

DOI 10.1145/1541885.1541888 <http://doi.acm.org/10.1145/1541885.1541888>

Abstract. For a graph (V, E) , existing compact linear formulations for the minimum cut problem require $\Theta(|V||E|)$ variables and constraints and can be interpreted as a composition of $|V| - 1$ polyhedra for minimum s - t cuts in much the same way as early approaches to finding globally minimum cuts relied on $|V| - 1$ calls to a minimum s - t cut algorithm. We present the first formulation to beat this bound, one that uses $O(|V|^2)$ variables and $O(|V|^3)$ constraints. An immediate consequence of our result is a compact linear relaxation with $O(|V|^2)$ constraints and $O(|V|^3)$ variables for enforcing global connectivity constraints. This relaxation is as strong as standard cut-based relaxations and has applications in solving traveling salesman problems by integer programming as well as finding approximate solutions for survivable network design problems using Jain's iterative rounding method. Another application is a polynomial-time verifiable certificate of size n for for the NP-complete problem of l_1 -embeddability of a rational metric on an n -set (as opposed to a certificate of size n^2 known previously).

Categories and Subject Descriptors: G.1.6 [Numerical Analysis]: Optimization—*Linear programming; integer programming*; F.1.1 [Computation by Abstract Devices]: Models of Computation; F.1.3 [Computation by Abstract Devices]: Complexity Measures and Classes; G.2.2 [Discrete Mathematics]: Graph Theory—*Network problems*

General Terms: Algorithms, Theory

Additional Key Words and Phrases: Linear programming formulation complexity, minimum cut problem

ACM Reference Format:

Carr, R. D., Konjevod, G., Little, G., Natarajan, V., and Parekh, O. 2009. Compacting cuts: A new linear formulation for minimum cut. *ACM Trans. Algor.* 5, 3, Article 27 (July 2009), 16 pages. DOI = 10.1145/1541885.1541888 <http://doi.acm.org/10.1145/1541885.1541888>

1. Introduction

Mathematical programming has enjoyed a burgeoning presence in theoretical computer science, both as a framework for developing algorithms and, increasingly, as a bona fide model of computation whose limits are expressed in terms of sizes of formulations and integrality gaps of formulations [Arora et al. 2002; Alekhovich et al. 2005; Khot and Vishnoi 2005].

The impact of mathematical programming is also evident in algorithms for combinatorial optimization problems. In fact, some of the most celebrated algorithmic results of the last decade have been achieved by the reduction of the problem in question to a linear or semidefinite programming model [Goemans and Williamson 1995; Jain 2001]. In many cases, the algorithms are directly based on such mathematical programming models and no combinatorial algorithms are known for such problems. Even those algorithms that do not rely directly on a mathematical program solver often have designs that were explicitly governed by insights gained from a mathematical programming model (for example, primal-dual algorithms; see Vazirani [2001]). It is not difficult to see why linear formulations are an appealing model of computation: Both optimization and decision problems fit naturally into the framework, and both theoretically tractable and efficient practical algorithms exist for solving linear programs.

Smaller formulations for linear relaxations of NP-hard problems are important in the design of efficient approximation algorithms. Perhaps more surprising is the fact that smaller formulations for problems already known to be in P can have an impact on both the exact and approximation solution of NP-hard problems. For instance, state-of-the-art approaches to exactly solving large-scale instances of many NP-hard problems rely on integer programming approaches that require the repeated solution of integer programs representing problems in P [Barnhart et al. 1998].

Polynomial-sized exact linear formulations for these problems can be used in place of the integer programs to potentially improve performance [Barnhart et al. 1998]. In the case of the traveling salesman problem [Applegate et al. 1998], one of the most famous combinatorial optimization problems, minimum cut is one such subroutine, and smaller exact formulations for it are likely to yield faster exact algorithms for large instances traveling salesman problem.

From a theoretical point of view, linear formulation complexity offers a different perspective than that offered by other prominent models of computation. For instance, even though, algorithmically, spanning tree is considered a relatively easy to solve member of P, the smallest known linear formulation requires $\Theta(|V||E|)$ constraints and variables [Wong 1984; Martin 1991; Tamir 1991]. In fact, before the results we present in this article, the same was true of minimum cut [Conforti et al. 2000; Tamir 1994]. The situation of the matching problem is a most peculiar one, since this problem can be solved in polynomial time, yet has resisted all attempts to find polynomial-size linear formulations. Another highly interesting fact about this problem is that the matching problem can be solved in polynomial time by solving an exponential-size linear program. The foundation of this result is the equivalence between separation and optimization problems in linear programming (see also Section 2).

Some of these results can be placed in a better context by observing that Turing machine complexity is perhaps not the ideal model for comparison since the size of a linear program seems to intuitively depend upon an efficient encoding of feasible solutions. Yannakakis observed this and forged a connection between linear formulation complexity and nondeterministic communication complexity [Yannakakis 1991]. He also showed that any symmetric linear programming formulation of matching or of the traveling salesman problem must have exponential size. These results represent some of the earliest work on linear formulation complexity.

The results of Yannakakis are not unconditional; they give lower bounds only for linear programming formulations that satisfy a symmetry condition. However, Yannakakis says “it is not clear what can be gained by treating one node differently than another.” Our result can be seen as an example where additional asymmetry seems to allow a more compact linear formulation (however, we do not have a proof of this, such as would follow from a tight lower bound on the size of a symmetric formulation).

Martin [1991] studied the relationship between optimization and separation problems (see also Section 2), and observed that his results imply a linear formulation for the minimum spanning tree problem with $\Theta(|V|^3)$ variables and $\Theta(|V|^3)$ constraints. This is no more compact than the standard relaxation [Magnanti and Wolsey 1995], and Martin asked whether there exists a linear formulation with $O(|V|^2)$ variables or constraints. While this remains an open problem, in this article we answer the analogous question positively for the related minimum cut problem.

1.1. MINIMUM CUT. The *connectivity* of a weighted undirected graph (V, E) is the minimum total capacity of a set of edges whose removal disconnects the graph. The problem of finding the connectivity of a (weighted) graph is called the (*global*) *minimum cut*, or *min cut*, problem.

Traditional linear programming formulations for the minimum cut problem [Tamir 1994] require $\Theta(|V||E|)$ variables and constraints and can be interpreted as a composition of $(|V| - 1)$ minimum *s-t* cut polyhedra (see Tamir [1994] and

Conforti et al. [2000, Proposition 2]) in much the same way as the early algorithms for globally minimum cuts rely on $|V| - 1$ calls to a minimum s - t cut algorithm. Better algorithms have been found, for example, the $O(|V||E|)$ -time algorithm by Nagamochi and Ibaraki [1992], simplified by Stoer and Wagner [1997], and the almost linear randomized algorithm by Karger [2000], but no linear programming formulation has been known that uses $o(|V||E|)$ variables or constraints.

1.2. OUR RESULTS. We show that the global minimum cut problem has linear formulation complexity $o(|V||E|)$. In particular, we show this problem can indeed be modeled in a way that takes advantage of global connectivity, as opposed to simply composing $|V| - 1$ (s - t)-cut formulations.

Our formulation requires only $O(|V|^2)$ variables and $O(|V|^3)$ constraints, thus reducing the number of variables by a factor of $|V|$ on dense graphs. We give two proofs of our result. The first is based on the powerful graph-theoretic notion of splitting-off [Frank 1992; Lovász 1979] and uses induction on the size of a minimal counterexample. Our formulation can be seen (in a dual sense) as a sequence of splitting-off operations that preserve (local) connectivity while reducing the graph. The second proof is somewhat longer, but it is both elementary and constructive in that it shows how to efficiently reconstruct an integral solution from any optimal fractional solution. This allows us to recover a convex combination of cuts dominated by any given feasible fractional solution to the linear program.

Also of note is that using our formulation, it is possible to obtain an $O(|V|^2) \times O(|V|^3)$ relaxation for the k -edge-connected spanning subgraph problem, which is that of finding a minimum cost k -edge-connected subgraph in a weighted graph. The relaxation obtained is equivalent in strength to the standard cut-based relaxation but has implications for algorithms that explicitly rely on solving LP relaxations, such as Jain's iterative rounding procedure [Jain 2001].

Another consequence of our work is a smaller polynomially verifiable certificate for ℓ_1 -embeddability of metrics, consisting of $\Theta(n)$ instead of $\Theta(n^2)$ rational values (see Section 5). Practical applications of our work are mostly to solving large-scale integer programs, where parts of the formulation may impose global connectivity constraints, and our formulation will help reduce the size of the instance.

1.3. PRELIMINARIES. We use mostly standard notation that is consistent with Schrijver's text [Schrijver 2003]. We let $V_n = \{1, \dots, n\}$, and by E_n we mean the edge set $\{\{i, j\} \mid i \in V_n, j \in V_n\}$. We abbreviate $\{i, j\}$ as ij , or equivalently ji ; when an edge ij is the index of a component of a vector, we write i, j instead. For a graph $G = (V, E)$, if we are given a set $S \subseteq V$, then we define the edge sets $\delta(S) = \{i, j \in E \mid |\{i, j\} \cap S| = 1\}$ and $E(S) = \{i, j \in E \mid |\{i, j\} \cap S| = 2\}$; we abbreviate $\delta(\{v\})$ as $\delta(v)$. The graph $G - v$, for some $v \in V$, is obtained from G by deleting v and all edges in $\delta(v)$. We let $con(G, c)$ refer to the value of a minimal cut in G with respect to the cost vector c ; if the value of c is clear from context we simply use $con(G)$.

Given sets S and I , we use S^I to refer to an $|I|$ -dimensional vector space in which the components of each vector in S^I are in a correspondence with the elements of I . For a vector $x \in S^I$, and a set $J \subseteq I$, by $x(J)$ we mean $\sum_{j \in J} x_j$; the vector $x|_J \in S^J$ is obtained from the vector $x \in S^I$ by dropping all components in $I \setminus J$.

We refer the reader to Yannakakis [1991] for a more detailed discussion of linear formulations of optimization problems; here, by an $M \times N$ formulation we mean

a linear program with M constraints, excluding nonnegativity constraints, and N variables. By a *compact formulation* we mean one in which M and N are polynomial in the representation size of the optimization problem instance.

2. Compact Relaxations for STSP and Network Design Relaxations

The motivation for our compact minimum cut formulation has roots in compact linear relaxations for the Symmetric Traveling Salesman Problem (STSP), which, given a complete graph $G = (V = V_n, E = E_n)$ with edge costs c , consists in finding a minimum cost Hamilton cycle in G . The classic subtour elimination polytope was defined by the linear relaxation for STSP given by Dantzig et al. [1954].

$$\begin{aligned} & \text{Minimize } \sum_{ij \in E} c_{i,j} x_{i,j} && SEP(n) \\ & \text{subject to} \\ & \quad x(\delta(v)) = 2 \quad \forall v \in V && (1) \\ & \quad x(\delta(S)) \geq 2 \quad \forall S \subseteq V : \emptyset \neq S \neq V && (2) \\ & \quad 1 \geq x_{i,j} \geq 0 \quad \forall ij \in E. && (3) \end{aligned}$$

For a more thorough exposition consult, for instance, Schrijver [2003, Chapter 58]. The LP $SEP(n)$ has an exponential number of constraints (2), however an optimal solution can be obtained in polynomial time via the ellipsoid algorithm since the separation problem for Eq. (2) is precisely minimum cut. We refer the reader to Grötschel et al. [1988, Chapter 6] for an in-depth discussion of the relation between separation and optimization. In fact, by an application of linear programming duality, we have the following.

LEMMA 2.1 [MARTIN 1991; CARR AND LANCIA 2002]. *From an $M \times N$ compact formulation of the separation problem associated with an optimization problem, it is possible to generate an $(N + 1) \times M$ compact formulation for the optimization problem.*

Applying this lemma to a compact linear formulation for minimum cut yields a compact set of inequalities which are as strong as and may replace (2) in $SEP(n)$. Thus our results give a compact formulation equivalent to $SEP(n)$ with $O(n^3)$ variables and $O(n^2)$ constraints. Previously the smallest known equivalent formulations had size $\Theta(n^3) \times \Theta(n^3)$. Wong [1980], Claus [1984], Arthanari [1982], Arthanari and Usha [2000], and Carr [1996; 1995] had all proposed such relaxations. The formulation derived from our work, which we present shortly, can be seen as an extension of Carr's cycle-shrink relaxation [Carr 1996]; however, we note that Arthanari and Usha's multistage-insertion relaxation [Arthanari and Usha 2000] is equivalent in strength [Arthanari and Usha 2001].

For any positive integer r , the set of linear inequalities we derive can be used in any linear relaxation in place of the constraints, $x(\delta(S)) \geq r$ for all $S \subseteq V : \emptyset \neq S \neq V$. Let $V' = V \setminus \{1\}$. We introduce $n - 1$ vectors of variables, $y^k \in \mathbb{R}^{E_k}$ for $k \in V'$, where it is possible to think of each y^k as representing the edges in the complete

graph (V_k, E_k) .

$$\sum_{k \in V'} y_e^k = x_e \quad \forall e \in E \quad (4)$$

$$y^k(\delta(k) \cap E_k) \geq r \quad \forall k \in V' \quad (5)$$

$$y^k(\delta(i) \cap E_k) \geq 0 \quad \forall k \in V', i \in V : i < k \quad (6)$$

$$y_{i,k}^k \geq 0 \quad \forall k \in V', i \in V : i < k \quad (7)$$

$$y_{i,j}^k \leq 0 \quad \forall k \in V', ij \in E_{k-1} \quad (8)$$

We observe that there are $O(n^3)$ variables and $O(n^2)$ constraints, excluding the nonnegativity and nonpositivity constraints (7) and (8), respectively. If the variables were integral, we could inductively think of each y^k as adding a new vertex k to the r -connected subgraph of E_{k-1} defined by $y^2 + \dots + y^{k-1}$ in a way that preserves r -connectivity. The inequalities insist that r edges among $\delta(k) \cap E_k$ are added, which is modeled by Eq. (5) and Eq. (7). Additionally some edges in E_{k-1} which are selected in $y^2 + \dots + y^{k-1}$ are allowed to be deleted in a way that preserves r -connectivity; this is modeled by Eq. (6) and Eq. (8). We do not prove this explicitly here, but note that this follows from the results of the next section.

THEOREM 2.2. *The constraints (4)–(8) are equivalent to $x(\delta(S)) \geq r$ for all $S \subseteq V : \emptyset \neq S \neq V$.*

With the results we prove in the next section, this theorem follows through some algebraic manipulation and the constructive proof of Lemma 2.1 [Martin 1991; Carr and Lancia 2002], which essentially follows from linear programming duality.

Our formulation also has applications for other network design problems. The inequalities (4)–(8) can be extended to model generalizations such as Steiner cuts.

3. The Formulation

We start by taking a complete graph $G = (V = V_n, E = E_n)$ with an edge-cost vector $c \in \mathbb{Q}_+^E$. Our formulation has a variable $x_{i,j}$ for each edge ij , and requires only $n - 1$ additional variables and $O(n^3)$ constraints.

$$\text{Minimize } \sum_{ij \in E} c_{i,j} x_{i,j} \quad P(n)$$

subject to

$$\sum_{2 \leq i \leq n} z_i = 1 \quad (9)$$

$$x_{i,k} + x_{j,k} \geq x_{i,j} + 2z_k \quad \forall ij \in E, k \in V : \quad (10)$$

$$i, j < k$$

$$x_{i,k} \geq z_k \quad \forall i \in V, k \in V : \quad (11)$$

$$i < k$$

$$x_{i,j} \geq 0 \quad \forall ij \in E \quad (12)$$

$$z_i \geq 0 \quad \forall i \in V \setminus \{1\}. \quad (13)$$

3.1. PROOF OF CORRECTNESS. To establish the correctness of the previous formulation, we begin by showing that the incidence vectors of cuts in G can indeed be extended to feasible solutions of $P(n)$.

LEMMA 3.1. *If x is the incidence vector of a cut $C \subseteq E$ then there exists a $z \in \{0, 1\}^{V \setminus \{1\}}$ such that $(x, z) \in P(n)$.*

PROOF. Let $S \subset V$ be the set such that $C = \delta(S)$ and $1 \in S$. We construct a $z \in \{0, 1\}^{V \setminus \{1\}}$ such that $(x, z) \in P(n)$. If l is the least index of any vertex in $V \setminus S$, we set $z_l = 1$ and $z_i = 0$ for $i \neq l$. Observe that our choice of z satisfies (9), (13), and the members of (11) and (10) with $k \neq l$, since the latter simply reduce to standard triangle inequalities in this case. It remains to consider (11) and (10) when $k = l$.

By our choice of l , we have $l \in V \setminus S$ while $i \in S$ for all $i < l$, hence $x_{i,l} = 1$ for all $i < l$, and (x, z) satisfies (11). Similarly for a member of (10) with $k = l$ we have $i, j \in S$, hence $x_{i,k} = x_{j,k} = 1$, $x_{i,j} = 0$, and the inequality is satisfied. \square

Next we show that the projections of integral solutions of $P(n)$ onto the x variables are indeed related to cuts in G .

LEMMA 3.2. *If $(x, z) \in P(n)$ is integral then x dominates the incidence vector of a cut $C \subseteq E$.*

PROOF. We assume $n \geq 2$ is the minimal value for which the statement does not hold. If $n = 2$, then (9) and (11) imply $x_{1,2} \geq 1$, so we must have $n > 2$. By (9) and (13) there exists an $l \in V \setminus \{1\}$ such that $z_l = 1$. If $l = n$, then by considering (11) for each $i < n$, we see that x dominates $\delta(\{n\})$, hence we must have $l \neq n$.

If we let $V' = V \setminus \{1, n\}$, $x' = x|_{E_{n-1}}$ and $z' = z|_{V'}$, then observe that $(x', z') \in P(n-1)$. Thus by our initial assumption on the minimality of n , there exists a set $S \subset V'$ such that x' dominates $\delta(S)$; let $T = V' \setminus S$. It suffices for us to show that x dominates either $\delta(S \cup \{n\})$ or $\delta(T \cup \{n\})$. This does not hold only if there are $i \in S$ and $j \in T$ such that $x_{i,n} = x_{j,n} = 0$; however, since $x_{i,j} = 1$, the inequality of (10) with i and j as defined earlier and $k = n$ precludes this. \square

Lemma 3.1 establishes that $P(n)$ is a relaxation for the set of incidence vectors of minimum cuts: The optimal value of $P(n)$ is at most $con(G)$. Our goal in this section is to show that this bound is tight, which by way of a convex decomposition lemma of Carr and Vempala [2002] implies that $P(n)$ has an integral optimum solution. In the next section we give an explicit procedure for obtaining such an integral convex decomposition of a feasible fractional solution.

THEOREM 3.3. *The optimal value of $P(n)$ is equal to $con(G)$.*

We will need the following well-known lemma conjectured by Lovász [1976] and proven by Mader [1978].

LEMMA 3.4. *Let G be an Eulerian multigraph, $x \in V(G)$, and suppose G is k -edge-connected between any two vertices $u, v \neq x$. Then we can find two neighbors y, z of x such that, if we remove two edges xy and xz but join y to z by a new edge, the resulting graph is still k -edge-connected between any two vertices $u, v \neq x$. Moreover, if x has at least two distinct neighbors, then there exist distinct y, z satisfying the preceding conditions.*

The operation outlined by this lemma is known as a *splitting-off* operation (x, y, z) at the vertex x . The distinctness condition is not always stated but can

be guaranteed [Frank 1992]. The key observation is that we may interpret each inequality (10) as corresponding to a splitting-off operation in G . We explore and make this idea precise in the proof of the next lemma.

LEMMA 3.5. *If $G = K_n$ and c is an even integral vector, then for any vector $(x, z) \in P(n)$ there exists a vector $d \in \mathbb{Z}_+^{V \setminus \{1\}}$ such that $c \cdot x \geq d \cdot z$ and $\text{con}(G, c) = \min_{2 \leq i \leq n} d_i$.*

PROOF. We proceed by induction on n . If $n = 2$, then the inequality (11) implies $c_{1,2}x_{1,2} \geq c_{1,2}z_2 = \text{con}(G, c) \cdot z_2$ for any feasible (x, z) . If $n > 2$ we let $G' = (V, E')$ be the multigraph that contains exactly $c_{u,v}$ parallel edges between each pair of vertices u, v , so that

$$\text{con}(G, c) = \text{con}(G', \mathbf{1}). \quad (14)$$

Lemma 3.4 guarantees a sequence of splitting-off operations, $S = ((n, s_1, t_1), \dots, (n, s_k, t_k))$ in G' at vertex n such that: (i) vertex n has a single neighbor $s \in V$ in the resulting multigraph H' , and (ii) $\text{con}(G', \mathbf{1}) = \min\{|\delta_{G'}(n)|, \text{con}(H', \mathbf{1})\}$. Moreover, since no minimal cut in H' may separate n and s , we have

$$\text{con}(G', \mathbf{1}) = \min\{|\delta_{G'}(n)|, \text{con}(H' - n, \mathbf{1})\}. \quad (15)$$

For each pair $u, v \in V$, we let $h_{u,v}$ be the number of edges between u and v in $H' - n$, so that

$$\text{con}(H' - n, \mathbf{1}) = \text{con}(G - n, h). \quad (16)$$

If $(x, z) \in P(n)$ then observe that the restriction $(x, z)|_{E_{n-1} \times V_{n-1} \setminus \{1\}}$ belongs to $P(n-1)$, hence by our inductive hypothesis, there exists a vector $d \in \mathbb{Z}_+^{V_{n-1} \setminus \{1\}}$ such that $h \cdot x|_{E_{n-1}} \geq d \cdot z|_{\{2, \dots, n-1\}}$ and $\text{con}(G - n, h) = \min_{2 \leq i \leq n-1} d_i$. Setting $d_n = |\delta_{G'}(n)|$, our goal is to use the aforesaid bound on $h \cdot x|_{E_{n-1}}$ to show that $c \cdot x \geq d \cdot z + d_n z_n$, which would imply the lemma since by (14), (15), and (16) we have already established $\text{con}(G, c) = \min\{|\delta_{G'}(n)|, \text{con}(G - n, h)\} = \min_{2 \leq i \leq n} d_i$.

We derived h from c by the sequence of splitting-off operations S , and we observe that each triple in S corresponds to an equality of Eq. (10) whose coefficients model the corresponding operation (in this calculation, the addition and equality signs on the left-hand side refer to addition of whole inequalities, not just the left-hand sides).

$$\begin{aligned} & \sum_{i,j \in E(K_{n-1})} h_{i,j} x_{i,j} \geq \sum_{2 \leq i \leq n-1} d_i z_i \\ + & \sum_{1 \leq i \leq k} (x_{s_i, n} + x_{t_i, n} - x_{s_i, t_i} \geq 2z_n) \\ + & |\delta_{H'}(n)| \cdot (x_{s, n} \geq z_n) \\ = & \sum_{i,j \in E(K_n)} c_{i,j} x_{i,j} \geq d_n z_n + \sum_{2 \leq i \leq n-1} d_i z_i \quad \square \end{aligned}$$

PROOF OF THEOREM 3.3. Without loss of generality we may assume that c has been scaled so that it is an integral vector whose components are even. For any $d \in \mathbb{Z}_+^{V \setminus \{1\}}$ and feasible solution $(x, z) \in P(n)$, (9) and (13) imply that $d \cdot z \geq \min_{2 \leq i \leq n} d_i$. Thus by Lemma 3.5 we have $c \cdot x \geq \text{con}(K_n, c)$. \square

3.2. STEINER CUTS. Given an undirected graph $G = (V, E)$ with an edge-cost vector $c \in \mathbb{Q}_+^E$, and a subset $S \subseteq V$ (whose elements are called the *terminals*), the *Steiner edge connectivity* of S is the minimum cost of an edge cut in G that terminals are found in at least two shores of the cut. The cut that realizes the Steiner connectivity of S is called the *minimum Steiner cut*. The problem of finding the minimum Steiner cut was first studied by Dinitz and Vainshtein [1994] (who called it the minimum S -cut), and later by Cole and Hariharan [2003]. Our formulation extends directly to this problem. Indeed, if we remove from the LP formulation $P(n)$ all the variables z_i for $i \in V \setminus S$, the resulting linear program gives an exact formulation for the minimum Steiner cut.

4. Constructive Decomposition

In this section, we give an alternative proof of Theorem 3.3. As opposed to the one we presented in Section 3, this is a constructive proof, and as such provides an algorithm to extract an integral solution, from an optimal solution to the linear program $P(n)$.

It is easy to describe how to extract an integral solution. We first do this and then focus on proving the correctness of our procedure. As a corollary of our result, we find that any feasible solution to the linear program $P(n)$ dominates a convex combination of cuts.

4.1. EXTRACTING AN INTEGRAL SOLUTION. Let (x^*, z^*) be an optimal solution to the LP $P(n)$. We assume that (x^*, z^*) is *minimal*, that is, for any $x < x^*$, the vector (x, z^*) is not feasible for the linear program. This assumption can be made without loss of generality because the cost $c_{i,j}$ is nonnegative for every edge ij . To simplify the exposition, we'll sometimes refer to $x_{j,i}$ as $x_{i,j}$ for $i < j$, and assume for convenience that $x_{i,i} = 0$ for all i .

We use the following notation.

$$\begin{aligned} k^A &= \max \{i \mid z_i^* > 0\} \\ S &= \{\{j, k\} \mid x_{j,k}^* > 0 \text{ and } (x_{k^A,j}^* = 0 \text{ or } x_{k^A,k}^* = 0)\} \\ \lambda &= \min(\{z_{k^A}^*\} \cup \{x_{i,j}^* \mid \{i, j\} \in S\}) \end{aligned}$$

Now we can define the integral vector (x^A, z^A) .

$$\begin{aligned} x_{i,j}^A &= \begin{cases} 1, & \{i, j\} \in S \\ 0, & \{i, j\} \notin S \end{cases} \\ z_i^A &= \begin{cases} 1, & i = k^A \\ 0, & i \neq k^A \end{cases} \end{aligned} \tag{17}$$

We will show that (x^A, z^A) is feasible for the linear program, and therefore, by Lemma 3.1, a cut.

THEOREM 4.1. $\{x^A, z^A\}$ is a feasible solution to the LP.

When we “peel away” A , what remains we call B .

$$\begin{aligned} x_{i,j}^B &= \frac{1}{1-\lambda} \cdot \begin{cases} x_{i,j}^* - \lambda, & \{i, j\} \in S \\ x_{i,j}^*, & \{i, j\} \notin S \end{cases} \\ z_i^B &= \frac{1}{1-\lambda} \cdot \begin{cases} z_i^* - \lambda, & i = k^A \\ z_i^*, & i \neq k^A \end{cases} \end{aligned} \quad (18)$$

In other words, (x^*, z^*) is a convex combination of (x^A, z^A) and (x^B, z^B) .

$$\lambda(x^A, z^A) + (1-\lambda)(x^B, z^B) = (x^*, z^*)$$

Just like (x^A, z^A) , this new pair of vectors forms a feasible solution.

THEOREM 4.2. $\{x^B, z^B\}$ is a feasible solution to the LP.

The main result of this section is summarized in Theorem 4.3.

THEOREM 4.3. *If (x^*, z^*) is a minimal optimal solution to the linear program $P(n)$, then (x^A, z^A) is an optimal integral solution and its cost is equal to the cost of (x^*, z^*) .*

4.2. CONVEX DECOMPOSITION INTO CUTS. Eq. (17) and Theorem 4.3 show us how to round an optimal fractional solution of LP $P(n)$ to an integral solution of equal cost. A stronger statement follows directly.

THEOREM 4.4. *Any minimal feasible solution to the linear program $P(n)$ is a convex combination of cuts.*

PROOF. Let (x^*, z^*) be a minimal feasible solution, and define (x^A, z^A) and (x^B, z^B) by (17) and (18). If (x^B, z^B) is integral, we are done. Otherwise, we would like to treat (x^B, z^B) just like we did (x^*, z^*) and continue by extracting another integral solution. Such a process can go on for at most $n - 1 + \binom{n}{2}$ steps, because each time at least one additional variable is reduced to 0. The resulting cuts then give the claimed convex combination.

This will be possible as long as (x^B, z^B) is guaranteed to be minimal. To see that this holds, suppose that $(x^C, z^C) < (x^B, z^B)$ were a feasible solution. Then $\lambda(x^A, z^A) + (1-\lambda)(x^C, z^C)$ would be both feasible (as a convex combination of two feasible solutions) and strictly dominated by (x^*, z^*) . Since (x^*, z^*) is minimal, it follows that (x^B, z^B) is minimal. \square

4.3. PROOFS.

LEMMA 4.5. *If $j < k$ and $x_{j,k}^* > z_k^*$, then there exists $i < k, i \neq j$ such that $x_{i,k}^* + x_{j,k}^* - 2z_k^* = x_{i,j}^*$.*

PROOF. By minimality, $x_{j,k}^*$ cannot be reduced without violating some constraint. If $x_{j,k}^* > z_k^*$, the only active constraint for $x_{j,k}^*$ can be one of the constraints (10). \square

LEMMA 4.6 (TRIANGLE INEQUALITY). *If $i \neq j \neq k$ then $x_{i,j}^* + x_{i,k}^* \geq x_{j,k}^*$.*

PROOF. Let m be the minimum possible value for which there exists a triple $\{i, j, k\}$ such that

$$x_{i,j}^* + x_{i,k}^* < x_{j,k}^* \quad (19)$$

and $m = \max\{i, j, k\}$. By symmetry, we may assume $j < k$.

In the case where $j < k < i$, the LP constraint $x_{j,i}^* + x_{k,i}^* - 2z_i^* \geq x_{j,k}^*$ contradicts the assumption (19).

In the case where $i < k$, we consider $x_{j,k}^*$. If $x_{j,k}^* = z_k^*$, then

$$\begin{aligned} x_{i,k}^* + x_{i,j}^* - x_{j,k}^* &\geq x_{i,k}^* - x_{i,j}^* - x_{j,k}^* \\ &= x_{i,k}^* + x_{j,k}^* - 2x_{j,k}^* - x_{i,j}^* \\ &= x_{i,k}^* + x_{j,k}^* - 2z_k^* - x_{i,j}^* \\ &\geq 0. \end{aligned}$$

If $x_{j,k}^* > z_k^*$, we have to do a little more work. First, by Lemma 4.5, there exists a p such that $p < k$, $p \neq j$ and

$$x_{j,k}^* + x_{p,k}^* - 2z_k^* = x_{p,j}^*. \quad (20)$$

If $p = i$, this becomes $x_{j,k}^* + x_{i,k}^* - 2z_k^* = x_{i,j}^*$. Since $x_{i,k}^* - z_k^* \geq 0$, we can subtract this quantity twice from the left side to obtain $x_{j,k}^* - x_{i,k}^* \leq x_{i,j}^*$, which again contradicts the assumption (19).

If $p \neq i$, then the fact that $i \neq j \neq p$, and $i, j, p < m$ and the choice of (minimal) m imply that $x_{p,i}^* + x_{i,j}^* \geq x_{p,j}^*$. We substitute the left-hand side for $x_{p,j}^*$ in (20) to get $x_{p,i}^* + x_{i,j}^* \geq x_{j,k}^* + x_{p,k}^* - 2z_k^*$. We also know from an LP constraint that $x_{i,k}^* + x_{p,k}^* - 2z_k^* \geq x_{p,i}^*$, which we can use to make the left side of our equation a little bigger: $x_{i,k}^* + x_{p,k}^* - 2z_k^* + x_{i,j}^* \geq x_{j,k}^* + x_{p,k}^* - 2z_k^*$. After some simple arithmetic, we arrive at $x_{i,k}^* + x_{i,j}^* \geq x_{j,k}^*$, which contradicts the assumption (19). \square

In order to show that both (x^A, z^A) and (x^B, z^B) are feasible, we will examine closely the structure of the graph induced by x^A . The next few lemmas list some useful properties of its edge-set S .

For conciseness, we occasionally write “ A xor B ”, where A and B are equations, to say that exactly one of the equations A or B holds.

LEMMA 4.7. *If $\{j, k\} \in S$, then either $x_{k^A,j}^* = 0$ or $x_{k^A,k}^* = 0$, but not both.*

PROOF. Let $\{j, k\} \in S$. By definition, then $x_{j,k}^* > 0$, and $(x_{k^A,j}^* = 0$ or $x_{k^A,k}^* = 0)$. Not both $x_{k^A,j}^*$ and $x_{k^A,k}^*$ can vanish, however, because then the triangle inequality would dictate that $x_{j,k}^* = 0$. \square

LEMMA 4.8. *If $j < k^A$, then $\{j, k^A\} \in S$.*

PROOF. We need only show that $x_{j,k^A}^* > 0$ and $(x_{k^A,k^A}^* = 0$ or $x_{k^A,j}^* = 0)$. Recall that $z_{k^A}^* > 0$ and $x_{j,k^A}^* \geq z_{k^A}^*$, so $x_{j,k^A}^* > 0$. Also, x_{k^A,k^A}^* is defined to be 0, so we satisfy the “or”-term. \square

LEMMA 4.9. *If $i, j < k^A$, then $\{i, j\} \notin S$.*

PROOF. To show that $\{i, j\} \notin S$, it is enough to show that $x_{i,k^A}^* \neq 0$ and $x_{j,k^A}^* \neq 0$; we use Lemma 4.8 to get $\{i, k^A\} \in S$ and $\{j, k^A\} \in S$, from which it follows that $x_{i,k^A}^* > 0$ and $x_{j,k^A}^* > 0$. \square

LEMMA 4.10. *If $i \neq j \neq k$, then $\{\{i, j\}, \{i, k\}, \{j, k\}\} \not\subset S$.*

PROOF. Assume the opposite and use Lemma 4.7 to get $(x_{k^A,i}^* = 0 \text{ xor } x_{k^A,j}^* = 0)$ and $(x_{k^A,i}^* = 0 \text{ xor } x_{k^A,k}^* = 0)$ and $(x_{k^A,j}^* = 0 \text{ xor } x_{k^A,k}^* = 0)$, which is false. \square

LEMMA 4.11. *If $i \neq j \neq k$ and $\{j, k\} \in S$, then $\{i, j\} \in S$ or $\{i, k\} \in S$.*

PROOF. Let $i \neq j \neq k$ such that $\{j, k\} \in S$, and assume $\{i, j\} \notin S$ and $\{i, k\} \notin S$. This gives

$$\begin{aligned} x_{j,k}^* &> 0 \text{ and } (x_{k^A,j}^* = 0 \text{ or } x_{k^A,k}^* = 0), \\ x_{i,j}^* &= 0 \text{ or } (x_{k^A,i}^* > 0 \text{ and } x_{k^A,j}^* > 0), \text{ and} \\ x_{i,k}^* &= 0 \text{ or } (x_{k^A,i}^* > 0 \text{ and } x_{k^A,k}^* > 0). \end{aligned} \quad (21)$$

Since $x_{j,k}^* > 0$, by triangle inequality we have that $x_{i,j}^* > 0$ or $x_{i,k}^* > 0$. By symmetry, assume $x_{i,j}^* > 0$, which means that $(x_{k^A,i}^* > 0 \text{ and } x_{k^A,j}^* > 0)$. Since $x_{k^A,j}^* > 0$, it follows from $(x_{k^A,j}^* = 0 \text{ or } x_{k^A,k}^* = 0)$ that $x_{k^A,k}^* = 0$. Using the triangle inequality on $x_{k^A,k}^* = 0$ and $x_{k^A,i}^* > 0$ gives us $x_{i,k}^* > 0$, which we can use to conclude that $(x_{k^A,i}^* > 0 \text{ and } x_{k^A,k}^* > 0)$. This contradicts the already derived equality $x_{k^A,k}^* = 0$. \square

The final lemma that we need will be used to argue that the subtraction used to define x^A and x^B in (17) and (18) is valid.

LEMMA 4.12. *If $i, j < k$ and $i \neq j$ and $\{i, k\} \in S$ and $\{j, k\} \in S$, then $x_{i,k}^* + x_{j,k}^* - 2\lambda \geq x_{i,j}^*$.*

PROOF. $\{i, j, k\}$ satisfies the conditions of the lemma, but $x_{i,k}^* + x_{j,k}^* - 2\lambda < x_{i,j}^*$. Take such a triple with minimum possible $m = \max\{i, j\}$. By symmetry, assume $m = j > i$.

First, suppose that $x_{k^A,k}^* \neq 0$. In this case we derive a contradiction, so assume that $x_{i,j}^* + x_{i,k}^* - 2\lambda < x_{j,k}^*$. Since $\{i, k\} \in S$ and $\{j, k\} \in S$, from Lemma 4.7 we know $(x_{k^A,i}^* = 0 \text{ xor } x_{k^A,k}^* = 0)$, and $(x_{k^A,j}^* = 0 \text{ xor } x_{k^A,k}^* = 0)$. Since $x_{k^A,k}^* \neq 0$, it follows that $x_{k^A,i}^* = 0$ and $x_{k^A,j}^* = 0$ and by the triangle inequality $x_{i,j}^* = 0$, which we can plug into our assumption to get $x_{i,k}^* + x_{j,k}^* - 2\lambda < 0$. But now (since $\{i, k\} \in S$ and $\{j, k\} \in S$), we have $x_{i,k}^* \geq \lambda$ and $x_{j,k}^* \geq \lambda$, which contradicts the previous inequality.

If $x_{k^A,k}^* = 0$, but $x_{i,j}^* = 0$, then the logic from the previous paragraph still applies (skipping to the part where we conclude that $x_{i,j}^* = 0$), so let us now consider the case where $x_{k^A,k}^* = 0$ and $x_{i,j}^* > 0$. First, since $\{i, k\} \in S$ and $\{j, k\} \in S$, Lemma 4.10 gives $\{i, j\} \notin S$, which implies that $j \neq k^A$ (or else Lemma 4.8 would dictate that $\{i, j\} \in S$). This gives us two further cases.

Case $j < k^A$. Here we start with the LP constraint $x_{i,k^A}^* + x_{j,k^A}^* - 2z_{k^A}^* \geq x_{i,j}^*$, and since $z_{k^A}^* \geq \lambda$, we get $x_{i,k^A}^* + x_{j,k^A}^* - 2\lambda \geq x_{i,j}^*$. Now if $k = k^A$ we're done (simply substitute k for k^A), so let $k \neq k^A$: first, since $\{j, k\} \in S$, Lemma 4.7 gives us $x_{j,k^A}^* = 0 \text{ xor } x_{k,k^A}^* = 0$. We can argue that $x_{j,k^A}^* > 0$ (since $\{j, k^A\} \in S$ by Lemma 4.8), which means $x_{k,k^A}^* = 0$. A similar argument applies to $\{i, k\} \in S$. Now we can use the triangle inequality to show that $x_{j,k^A}^* = x_{j,k}^*$ and $x_{i,k^A}^* = x_{i,k}^*$, which we can substitute into our previous inequality to get $x_{i,k}^* + x_{j,k}^* - 2\lambda \geq x_{i,j}^*$.

Case $j > k^A$. By the definition of k^A , we have $z_j^* = 0$, and since $x_{i,j}^* > 0$, we may conclude that $x_{i,j}^* > z_j^*$. Now by Lemma 4.5 there exists p such that $p < j$, $p \neq i$, and

$$x_{i,j}^* + x_{p,j}^* - 2z_j^* = x_{i,p}^*. \quad (22)$$

Since $z_j^* = 0$, we actually have $x_{i,j}^* + x_{p,j}^* = x_{i,p}^*$. At this point we have yet another two cases to consider, based on $x_{p,k}^*$.

Case $x_{p,k}^* > 0$. We see that $\{p, k\} \in S$, since we already know $x_{k^A,k}^* = 0$. This, along with the fact that $i, p < j = m$, and $i, p < k$, and $i \neq p$, allows us to use the inductive hypothesis (the maximality of $m = j$) to get $x_{i,k}^* + x_{p,k}^* - 2\lambda \geq x_{i,p}^*$. Substituting for $x_{i,p}^*$ gives $x_{i,k}^* + x_{p,k}^* - 2\lambda \geq x_{i,j}^* + x_{p,j}^*$. Now by the triangle inequality we have $x_{p,j}^* + x_{j,k}^* \geq x_{p,k}^*$, which we can use to increase the value of $x_{p,k}^*$ in our previous equation to get $x_{i,k}^* + x_{p,j}^* + x_{j,k}^* - 2\lambda \geq x_{i,j}^* + x_{p,j}^*$. This can now be simplified to $x_{i,k}^* + x_{j,k}^* - 2\lambda \geq x_{i,j}^*$.

Case $x_{p,k}^* = 0$. From the triangle inequalities we have $x_{i,k}^* = x_{i,p}^*$ and $x_{j,k}^* = x_{p,j}^*$. Substituting this into (22) gives $x_{i,j}^* + x_{j,k}^* = x_{i,k}^*$. Subtracting $x_{j,k}^*$ from both sides gives $x_{i,k}^* - x_{j,k}^* = x_{i,j}^*$. Next we'll note that $\{j, k\} \in S$, which means that $x_{j,k}^* \geq \lambda$. We then add $2(x_{j,k}^* - \lambda)$ to the left side of our previous equation to get $x_{i,k}^* + x_{j,k}^* - 2\lambda \geq x_{i,j}^*$. \square

PROOF OF THEOREM 4.1. The constraint $\sum z_i^A = 1$ is satisfied since only one component of z^A is positive (namely $z_{k^A}^A$), and it equals 1.

The constraint $x_{i,k}^A \geq z_k^A$ is only in question when $z_k^A > 0$, that is, when $k = k^A$, so let $i < k = k^A$. By Lemma 4.8, we have $\{i, k\} \in S$, which in turn means $x_{i,k}^A = 1$. Thus the constraint ends up requiring $1 \geq 1$, which is true.

The final LP constraint is $x_{i,k}^A + x_{j,k}^A - 2z_k^A \geq x_{i,j}^A$ for all $i, j < k, i \neq j$.

Take such a triple $\{i, j, k\}$. Now if $k = k^A$, then we know $z_k^A = 1$. Thus the constraint is equivalent to $x_{i,k^A}^A + x_{j,k^A}^A - 2 \geq x_{i,j}^A$. Since $\{i, k^A\}, \{j, k^A\} \in S$ (by Lemma 4.8), we know $x_{i,k^A}^A = 1 = x_{j,k^A}^A$, so the constraint reduces to $1 + 1 - 2 \geq x_{i,j}^A$. It thus suffices to show that $x_{i,j}^A = 0$, which we can do because $\{i, k^A\}, \{j, k^A\} \in S$, and so (by Lemma 4.10), $\{i, j\} \notin S$.

Now if $k \neq k^A$, then $z_k^A = 0$ by definition, and the constraint becomes $x_{i,k}^A + x_{j,k}^A \geq x_{i,j}^A$. It is possible that none of $\{i, j\}, \{i, k\}$, and $\{j, k\}$ is in S , in which case the constraint becomes $0 + 0 \geq 0$. Otherwise Lemmas 4.11 and 4.10 tell us that exactly two of $\{i, j\}, \{i, k\}$, and $\{j, k\}$ are in S , and the constraint reduces to either $1 + 1 \geq 0$, or $1 + 0 \geq 1$. Both of these are true. \square

PROOF OF THEOREM 4.2. Before looking at the constraints, it is worth mentioning that the implicit nonnegativity constraints are met because we only subtract λ from variables which we've already proven to be at least λ .

Now, the first LP constraint requires that $\sum z_i^B = 1$. This is true because

$$\sum z_i^B = \frac{1}{1-\lambda} \left(\sum_i z_i^* - \lambda \right) = \frac{1}{1-\lambda} (1 - \lambda) = 1.$$

The second LP constraint requires $x_{i,k}^B \geq z_k^B$ for $i < k$. Note that we won't bother talking about dividing by $1 - \lambda$, since we do this to both sides when obtaining this constraint from the $\{x^*, z^*\}$ version.

This constraint is not an issue if $k < k^A$, since in this case by Lemma 4.9 $\{i, k\} \notin S$, so we don't subtract from either side of the solution $\{x^*, z^*\}$. If $k = k^A$, then Lemma 4.8 gives $\{i, k^A\} \in S$, which means that we subtract from both sides of the $\{x^*, z^*\}$ version of this constraint, which is also fine. Finally, if $k > k^A$, then $z_k^B = z_k^* = 0$ (from the definition of k^A), turning the constraint into $x_{i,k}^B \geq 0$, which is satisfied since we already argued that all the variables are nonnegative.

The third LP constraint requires that $x_{i,k}^B + x_{j,k}^B - 2z_k^B \geq x_{i,j}^B$ for all $i, j < k$ where $i \neq j$. Again, we won't bother talking about dividing by $1 - \lambda$, since we do this to both sides. We'll first consider the case where none of $\{i, j\}$, $\{i, k\}$, and $\{j, k\}$ is in S . In this case, we know that $k \neq k^A$, or else Lemma 4.8 would dictate that $\{j, k\} \in S$. This means that our constraint remains the same as the $\{x^*, z^*\}$ version. Now, if at least one of $\{i, j\}$, $\{i, k\}$, and $\{j, k\}$ is in S , then Lemmas 4.11 and 4.10 tell us that exactly 2 of them are. This gives us three cases.

Case 1: $\{i, k\} \in S$ and $\{j, k\} \in S$. In this case we subtract 2λ from the left-hand side of the $\{x^*, z^*\}$ version of this constraint. Now we know $k \geq k^A$ by Lemma 4.9, so if $k = k^A$, then we also end up adding 2λ to the left-hand side of the constraint (in obtaining the $-2z_k^B$). If $k > k^A$, then the definition of k^A tells us that $z_k^B = 0$, and then Lemma 4.12 ensures that we can subtract 2λ from the left-hand side of the $\{x^*, z^*\}$ version, and still meet the constraint.

Cases 2 and 3: $\{i, k\} \in S$ and $\{i, j\} \in S$ or $\{j, k\} \in S$ and $\{i, j\} \in S$. In both these cases $\{i, j\} \in S$, which means that $k \neq k^A$. Both $\{i, j\} \in S$ and $\{i, k\} \in S$, if $\{i, j\} \in S$ by Lemma 4.10. Thus $z_k^B = z_k^*$, and we end up simply subtracting λ from both sides of the $\{x^*, z^*\}$ version of the constraint, which is fine. \square

PROOF OF THEOREM 4.3. First, it is not difficult to see that $\lambda(x^A, z^A) + (1 - \lambda)(x^B, z^B) = (x^*, z^*)$. For example, for each $\{i, j\}$ it follows from the definitions that $\lambda x_{i,j}^A + (1 - \lambda)x_{i,j}^B = x_{i,j}^*$. Indeed, if $\{i, j\} \in S$ we have $\lambda + (x_{i,j}^* - \lambda) = x_{i,j}^*$, and if $\{i, j\} \notin S$ we have $0 + x_{i,j}^* = x_{i,j}^*$. (A similar argument gives $\lambda z^A + (1 - \lambda)z^B = z^*$.)

In Theorems 4.1 and 4.2 we show that (x^A, z^A) and (x^B, z^B) are both valid solutions to the LP. Hence, the original solution (x^*, z^*) is a convex combination of an integral feasible solution (x^A, z^A) and another feasible solution. Since (x^*, z^*) is optimal so are both of the solutions that form the convex combination, and therefore the integral solution (x^A, z^A) is optimal for the linear program, which completes the proof. \square

5. ℓ_1 Embeddability

Every cut in a graph defines a (semi)metric d , where $d(v, w) = 1$ if v and w are on the opposite shores of the cut, and $d(v, w) = 0$ otherwise. Such *cut metrics* are closely related to ℓ_1 metrics, because every ℓ_1 metric on an n -element set V can be written as a nonnegative combination of cut metrics on V [Deza and Laurent 1997, p. 40]. Given a rational metric d , it is NP-complete to determine whether d is ℓ_1 -embeddable [Avis and Deza 1991]. The decomposition of d as $\sum_i \alpha_i \delta(S_i)$, where $\alpha_i > 0$ and $S_i \subseteq V$, is clearly a certificate for the ℓ_1 -embeddability of d .

The certificate consists of the vector α and the list of sets S_i . By Carathéodory's theorem, the number of cut metrics used need not be more than $\binom{n}{2}$, and so the certificate consists of $O(n^2)$ rational numbers and $O(n^2)$ n -bit vectors (to represent the sets).

Our linear programming formulation gives a more compact certificate. Take a decomposition of d into cut metrics as before and let $\lambda = \sum_i \alpha_i$. Then $d' = d/\lambda = \sum_i (\alpha_i/\lambda) \delta(S_i)$ is a convex combination of cut metrics. Let z be an $n-1$ -vector and $d'_{v,w} = d(v, w)/\lambda$. If (d', z) is feasible for our LP, then we can use the procedure of Theorem 4.1 to extract cut metrics that define d' and thus verify that d' is a convex combination of cut metrics. The $(n-1)$ -vector z thus together with λ forms a certificate for ℓ_1 -decomposability of d .

The standard linear programming formulation [Conforti et al. 2000; Tamir 1994] also provides a certificate, but it consists of n rational vectors of length n each, that is, a total of n^2 values.

6. Concluding Remarks and Open Questions

The $O(|V|^3) \times O(|V|^2)$ minimum cut formulation we present can be viewed (in a dual) sense as selecting a collection of splitting-off operations. Can we generalize this to other problems, for instance, spanning tree, by forging a connection between an algorithmic procedure and a formulation? Finally, can the relaxation we obtain for k -edge-connected spanning subgraph be extended to generalized Steiner network problems? The fact that splitting-off can be performed while preserving pairwise connectivity requirements indicates that this might be possible.

ACKNOWLEDGMENTS. The four latter authors thank the first author for the insights outlined in Section 2.

REFERENCES

- ALEKHNovich, M., ARORA, S., AND TOURLAKIS, I. 2005. Towards strong nonapproximability results in the Lovasz-Schrijver hierarchy. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, 294–303.
- APPEGATE, D., BIXBY, R., CHVÁTAL, V., AND COOK, W. 1998. On the solution of traveling salesman problems. *Documenta Mathematica Extra Volume ICM 1998*, 3, 645–656.
- ARORA, S., BOLLOBÁS, B., AND LOVÁSZ, L. 2002. Proving integrality gaps without knowing the linear program. In *Proceedings of the 43rd Symposium on Foundations of Computer Science*, 313–322.
- ARTHANARI, T. S. 1982. On the traveling salesman problem. *XI Symposium on Mathematical Programming*.
- ARTHANARI, T. S., AND USHA, M. 2000. An alternate formulation of the symmetric traveling salesman problem and its properties. *Discrete Appl. Math.* 98, 3, 173–190.
- ARTHANARI, T. S., AND USHA, M. 2001. On the equivalence of the multistage-insertion and cycle-shrink formulations of the symmetric traveling salesman problem. *Oper. Res. Lett.* 29, 3, 129–139.
- AVIS, D., AND DEZA, M. 1991. The cut cone, l^1 embeddability, complexity and multi-commodity flows. *Netw.* 21, 595–617.
- BARNHART, C., JOHNSON, E. L., NEMHAUSER, G. L., SAVELSBERGH, M. W. P., AND VANCE, P. H. 1998. Branch-and-Price: Column generation for solving huge integer programs. *Oper. Res.* 46, 3, 316–329.
- CARR, R. D. 1995. Polynomial separation procedures and facet determination for inequalities of the traveling salesman polytope. Ph.D. thesis, Carnegie Mellon University.
- CARR, R. D. 1996. Separating over classes of TSP inequalities defined by 0 node-lifting in polynomial time. In *Proceedings of the 5th Annual Integer Programming and Combinatorial Optimization Conference*, 460–474.

- CARR, R. D., AND LANCIA, G. 2002. Compact vs. exponential-size LP relaxations. *Oper. Res. Lett.* 30, 57–65.
- CARR, R. D., AND VEMPALA, S. 2002. Randomized metarounding. *Random Struct. Algor.* 20, 3, 343–352.
- CLAUS, A. 1984. A new formulation for the travelling salesman problem. *SIAM J. Algor. Disc. Meth.* 5, 21–25.
- COLE, R., AND HARIHARAN, R. 2003. A fast algorithm for computing Steiner edge connectivity. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing (STOC'03)*, 167–176.
- CONFORTI, M., RINALDI, G., AND WOLSEY, L. 2000. On the cut polyhedron. Tech. rep. 5, CORE, Université Catholique de Louvain.
- DANTZIG, G., FULKERSON, D., AND JOHNSON, S. 1954. Solution of a large-scale traveling salesman problem. *Oper. Res.* 2, 393–410.
- DEZA, M., AND LAURENT, M. 1997. *Geometry of Cuts and Metrics*. Springer-Verlag.
- DINITZ, Y., AND VAINSHTEIN, A. 1994. The connectivity carcass of a vertex subset in a graph and its incremental maintenance. In *Proceedings of the 26th Annual ACM Symposium on Theory of Computing (STOC'94)*, 716–725.
- FRANK, A. 1992. On a theorem of Mader. *Discrete Math.* 101, 1-3, 49–57.
- GOEMANS, M. X., AND WILLIAMSON, D. P. 1995. Improved approximation algorithms for maximum cut and satisfiability problems using semi-definite programming. *J. ACM* 42, 1115–1145.
- GRÖTSCHEL, M., LOVÁSZ, L., AND SCHRIJVER, A. 1988. *Geometric Algorithms and Combinatorial Optimization*. Springer, Berlin.
- JAIN, K. 2001. A factor 2 approximation algorithm for the generalized Steiner network problem. *Combinatorica* 21, 39–60.
- KARGER, D. R. 2000. Minimum cuts in near-linear time. *J. ACM* 47, 46–76.
- KHOT, S., AND VISHNOI, N. 2005. The unique games conjecture, integrality gap for cut problems and embeddability of negative type metrics into ℓ_1 . In *Proceedings of the 46th Symposium on Foundations of Computer Science*.
- LOVÁSZ, L. 1976. Unsolved problems. In *Proceedings of the 5th British Combinatorial Conference*. Utilitas Math., 683–685. Congressus Numerantium, No. XV.
- LOVÁSZ, L. 1979. *Combinatorial Problems and Exercises*. North-Holland, Amsterdam.
- MADER, W. 1978. A reduction method for edge-connectivity in graphs. *Ann. Discrete Math.* 3, 145–164.
- MAGNANTI, T. L., AND WOLSEY, L. A. 1995. Optimal trees. In *Network Models*, M. O. Ball et al., Eds. *Handbooks in Operations Research and Management Science*, vol. 7. Elsevier, 503–615.
- MARTIN, R. K. 1991. Using separation algorithms to generate mixed integer model reformulations. *Oper. Res. Lett.* 10, 119–128.
- NAGAMOCHI, H., AND IBARAKI, T. 1992. Computing edge-connectivity in multi-graphs and capacitated graphs. *SIAM J. Discrete Math.* 5, 54–66.
- SCHRIJVER, A. 2003. *Combinatorial Optimization: Polyhedra and Efficiency*. Springer. Schrijver.
- STOER, M., AND WAGNER, F. 1997. A simple min-cut algorithm. *J. ACM* 44, 4, 585–591.
- TAMIR, A. 1991. On the core of network synthesis games. *Math. Program.* 50, 123–135.
- TAMIR, A. 1994. Polynomial formulations of min-cut problems. Unpublished manuscript. http://www.tau.ac.il/~atamir/min_cut_94.pdf
- VAZIRANI, V. V. 2001. *Approximation Algorithms*. Springer.
- WONG, R. T. 1980. Integer programming formulations of the traveling salesman problem. In *Proceedings of the IEEE Conference on Circuits and Computers*, 149–152.
- WONG, R. T. 1984. A dual ascent approach for Steiner tree problems on a directed graph. *Math. Program.* 28, 271–287.
- YANNAKAKIS, M. 1991. Expressing combinatorial optimization problems by linear programs. *J. Comput. Syst. Sci.* 43, 3, 441–466.

RECEIVED AUGUST 2007; REVISED MARCH 2008; ACCEPTED APRIL 2008