MATHEMATICS OF OPERATIONS RESEARCH
Vol. 31, No. 4, November 2006, pp. 696–713
ISSN 0364-765X | EISSN 1526-5471 | 06 | 3104 | 0696
DOI 10.1287/moor.1060.0214

Polynomial-Time Separation of a Superclass of Simple Comb Inequalities

Lisa K. Fleischer
Department of Computer Science, Dartmouth College, 6211 Sudikoff Laboratory, Hanover, NH 03755-3510,
lkf@cs.dartmouth.edu

Adam N. Letchford
Department of Management Science, Lancaster University, Lancaster LA1 4YW, Great Britain
{A.N.Letchford@lancaster.ac.uk, http://www.lancs.ac.uk/staff/letchfoa/}

Andrea Lodi
DEIS, University of Bologna, Viale Risorgimento 2, 40136 Bologna, Italy; currently at IBM,
T. J. Watson Research Center, Yorktown Heights, NY 10598, USA,
alodi@deis.unibo.it

The *comb* inequalities are a well-known class of facet-inducing inequalities for the traveling salesman problem, defined in terms of certain vertex sets called the *handle* and the *teeth*. We say that a comb inequality is *simple* if the following holds for each tooth: Either the intersection of the tooth with the handle has cardinality one, or the part of the tooth outside the handle has cardinality one, or both. The simple comb inequalities generalize the classical *2-matching* inequalities of Edmonds [Edmonds, J. 1965. Maximum matching and a polyhedron with 0–1 vertices. *J. Res. Nat. Bur. Standards* **69B** 125–130] and also the so-called Chvátal comb inequalities. In 1982, Padberg and Rao [Padberg, M. W., M. R. Rao. 1982. Odd minimum cut-sets and b-matchings. *Math. Oper. Res.* **7** 67–80] gave a polynomial-time *separation algorithm* for the 2-matching inequalities, i.e., an algorithm for testing if a given fractional solution to an LP relaxation violates a 2-matching inequality. We extend this significantly by giving a polynomial-time separation algorithm for a class of valid inequalities which includes all simple comb inequalities.

Key words: traveling salesman problem; cutting planes; separation
MSC2000 subject classification: Primary: 90C57; secondary: 68W40, 90C27
OR/MS subject classification: Primary: Programming, networks/graphs; secondary: traveling salesman, cutting planes/facets
History: Received May 21, 2002; revised June 9, 2004 and October 18, 2005.

1. Introduction. The famous *symmetric traveling salesman problem* (STSP) is the $\mathcal{NP}$-hard problem of finding a minimum cost Hamiltonian cycle (or *tour*) in a complete undirected graph. The most successful optimization algorithms at present (e.g., Padberg and Rinaldi [32] and Applegate et al. [1]) are based on an integer programming formulation of the STSP due to Dantzig et al. [9], which we now describe.

Let G be a complete graph with vertex set V and edge set E. For each $e \in E$, let c_e be the cost of traversing edge e. For any $S \subset V$, let $\delta(S)$ (respectively, $E(S)$) denote the set of edges in G with exactly one end vertex (respectively, both end vertices) in S. Then, for each $e \in E$ define the 0-1 variable x_e taking the value one if e is to be in the tour, zero otherwise. Finally, let $x(F)$ for any $F \subset E$ denote $\sum_{e\in F} x_e$. Then, the formulation is:

$$\text{minimise} \quad \sum_{e\in E} c_e x_e$$

subject to:

$$x(\delta(\{i\})) = 2 \quad \forall\, i \in V, \tag{1}$$

$$x(E(S)) \le |S| - 1 \quad \forall\, S \subset V\colon 2 \le |S| \le |V| - 2, \tag{2}$$

$$x_e \ge 0 \quad \forall\, e \in E, \tag{3}$$

$$x \in \mathbb{Z}^{|E|}. \tag{4}$$

The equations (1) are called *degree equations*. The inequalities (2) are called *subtour elimination constraints* (SECs) and the inequalities (Equation 3) are simple non negativity conditions. Note that an SEC with $|S| = 2$ is a mere upper bound of the form $x_e \le 1$ for some edge e.

The convex hull in $\mathbb{R}^{|E|}$ of vectors satisfying Equations (1)–(4) is called a *symmetric traveling salesman polytope*. The polytope defined by Equations (1)–(3) is called a *subtour elimination polytope*. These polytopes are denoted by STSP(n) and SEP(n), respectively, where $n := |V|$. Clearly, STSP(n) $\subseteq$ SEP(n) and containment is strict for $n \ge 6$.

The polytopes STSP(n) have been studied in great depth and many classes of valid and facet-inducing inequalities are known; see the surveys by Jünger et al. [18] and Naddef [23]. Here, we are primarily interested in the *comb* inequalities of Grötschel and Padberg [14, 15], which are defined as follows. Let $t \geq 3$ be an odd integer. Let $H \subset V$ and $T_j \subset V$ for $j = 1, \ldots, t$ be such that $T_j \cap H \neq \emptyset$ and $T_j \backslash H \neq \emptyset$ for $j = 1, \ldots, t$, and also let the T_j be vertex disjoint. (See Figure 1 for an illustration.) The comb inequality is:

$$x(E(H)) + \sum_{j=1}^{t} x(E(T_j)) \leq |H| + \sum_{j=1}^{t} |T_j| - \lceil 3t/2 \rceil. \tag{5}$$

The set H is called the *handle* of the comb and the T_j are called *teeth*.

Comb inequalities induce facets of STSP(n) for $n \geq 6$ (Grötschel and Padberg [14, 15]). The validity of comb inequalities in the special case where $|T_j \cap H| = 1$ for all j was proved by Chvátal [6]. For this reason, inequalities of this type are sometimes referred to as *Chvátal comb* inequalities. If, in addition, $|T_j \backslash H| = 1$ for all j, then the inequalities reduce to the classical *2-matching* inequalities of Edmonds [10].

In this paper we are concerned with a class of inequalities which is intermediate in generality between the class of comb inequalities and the class of Chvátal comb inequalities. For want of a better term, we call them *simple* comb inequalities, although the reader should be aware that the term *simple* is used with a different meaning in Padberg and Rinaldi [31] and with yet another meaning in Naddef and Rinaldi [25].

Definition 1.1. A comb (and its associated comb inequality) will be said to be *simple* if, for all j, either $|T_j \cap H| = 1$ or $|T_j \backslash H| = 1$ (or both).

So, for example, the comb shown in Figure 1 is simple because $|T_1 \cap H|$, $|T_2 \backslash H|$, and $|T_3 \cap H|$ are all equal to one. Note, however, that it is not a Chvátal comb because $|T_2 \cap H|$ equals two.

For a given class of inequalities, a *separation algorithm* is a procedure which, given a vector $x^* \in \mathbb{R}^{|E|}$ as input, either finds an inequality in the class which is violated by x^* or proves that none exists (see Grötschel et al. [16]). In the context of separation for the STSP, it is helpful to define the edge set $E^* := \{e \in E: x_e^* > 0\}$ and the associated *support graph* $G^* = (V, E^*)$. It is also useful to define $m = |E^*|$, the number of variables which are positive at x^*.

A desirable property of a separation algorithm is that it runs in polynomial time. In 1982, Padberg and Rao [30] gave the first polynomial-time separation algorithm for the 2-matching inequalities. The algorithm is rather complicated and time-consuming, involving the computation of a minimum-weight odd cut in an expanded graph. Recently, Letchford et al. [22] gave a faster and simpler algorithm which, using the *preflow push* maximum flow algorithm of Goldberg and Tarjan [13] as a subroutine, runs in $\mathcal{O}(nm^2 \log(n^2/m))$ time.

In Padberg and Grötschel [29, p. 341], it is conjectured that there also exists a polynomial-time separation algorithm for the more general *comb* inequalities. This conjecture is still unsettled, and in practice many researchers resort to *heuristics* for comb separation (see, for example, Padberg and Rinaldi [31], Applegate et al. [1], and Naddef and Thienel [26]). Nevertheless, some progress has recently been made on the theoretical side. In chronological order:

- Carr [5] showed that for a *fixed* value of t, the separation problem for comb inequalities with t teeth reduces to solving $\mathcal{O}(n^{2t})$ maximum flow problems, which takes $\mathcal{O}(n^{2t+1} m \log(n^2/m))$ time using the preflow push algorithm.
- Fleischer and Tardos [11] gave an $\mathcal{O}(n^2 \log n)$ algorithm for detecting *maximally violated* comb inequalities. (A comb inequality is maximally violated if it is violated by $\frac{1}{2}$, which is the largest violation possible if $x^* \in \mathrm{SEP}(n)$.) However, this algorithm only works when G^* is planar.
- Caprara et al. [4] showed that the comb inequalities are contained in a more general class of inequalities called $\{0, \frac{1}{2}\}$-cuts, and showed how to detect maximally violated $\{0, \frac{1}{2}\}$-cuts in $\mathcal{O}(n^2 m)$ time.
- Letchford [19] defined a different generalization of the comb inequalities called *domino parity* inequalities, and showed that the associated separation problem can be solved in $\mathcal{O}(n^3)$ time when G^* is planar.

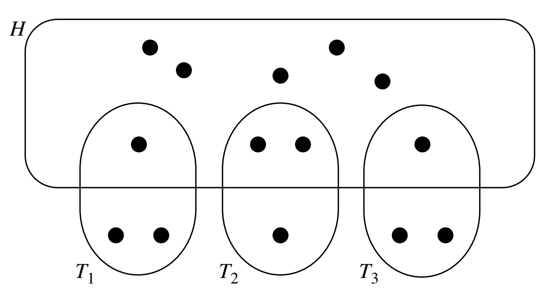

Figure 1. A comb with three teeth.

- Caprara and Letchford [3] showed that if the *handle* H is fixed, then the separation problem for a class of inequalities including all $\{0, \frac{1}{2}\}$-cuts, called *split cuts*, can be solved in polynomial time. They did not analyse the running time, but the order of the polynomial is likely to be very high.

In this paper, we make another step forward in this line of research by proving the following theorem:

THEOREM 1.1. *There is a polynomial time separation algorithm for a class of valid inequalities containing all simple comb inequalities* (*provided that* $x^* \in \text{SEP}(n)$).

This is a significant extension of the Padberg-Rao [30] result. As in Letchford [19], the proof is based on some results of Caprara and Fischetti [2] concerning $\{0, \frac{1}{2}\}$-cuts, together with arguments which enable one to restrict attention to a small (polynomial-sized) collection of *candidate teeth*.

The structure of the paper is as follows. In §2, we summarize the results given in Caprara and Fischetti [2] about $\{0, \frac{1}{2}\}$-cuts and show how they relate to the simple comb inequalities. In §3, we analyse the structure of candidate teeth. In §4, we describe a simple version of the separation algorithm and analyse its running time, which turns out to be very high at $\mathcal{O}(n^9 \log n)$. In §5, we show that the running time can be reduced to $\mathcal{O}(n^2 m^2 \log(n^2/m))$. Conclusions are given in §6.

NOTE. An extended abstract of this paper appeared in the 2002 IPCO proceedings (Letchford and Lodi [21]). However, the abstract gives a running time of $\mathcal{O}(n^3 m^3 \log n)$. Here, we describe an improved run time of $\mathcal{O}(n^2 m^2 \log(n^2/m))$. Moreover, we correct a minor error in Letchford and Lodi [21] by showing that our separation algorithm detects not only violated simple comb inequalities, but also inequalities in a slightly extended class.

2. Simple comb inequalities as $\{0, \frac{1}{2}\}$-cuts. As mentioned above, we will need some definitions and results from Caprara and Fischetti [2]. We begin with the definition of $\{0, \frac{1}{2}\}$-cuts:

DEFINITION 2.1. Given an integer polyhedron $P_I := \text{conv}\{x \in Z_+^q : Ax \leq b\}$, where A is a $p \times q$ integer matrix and b is a column vector with p integer entries, a $\{0, \frac{1}{2}\}$-*cut* is a valid inequality for P_I of the form

$$\lfloor \lambda A \rfloor x \leq \lfloor \lambda b \rfloor, \tag{6}$$

where the multiplier vector $\lambda \in \{0, \frac{1}{2}\}^p$ is chosen so that λb is not integral.

(Actually, Caprara and Fischetti [2] give a more general definition, applicable when variables are not necessarily required to be nonnegative, but the definition given here is more appropriate for our purposes. Also, note that an equation can easily be represented by two inequalities.)

Define the ith inequality in the system $Ax \leq b$ to be *used* if $\lambda_i = \frac{1}{2}$. The nonnegativity inequality for a given variable x_j is *used* if the jth coefficient of the vector λA is fractional. (Rounding down the coefficient of x_j on the left-hand side of Equation (6) is equivalent to adding one half of the nonnegativity inequality $-x_j \leq 0$.) Using this terminology, we have:

PROPOSITION 2.1. *Let $x^* \in \mathbb{R}_+^q$ be a point to be separated. Then, a given $\{0, \frac{1}{2}\}$-cut is violated by x^* if and only if the sum of the slacks of the inequalities used, computed with respect to x^*, is less than one.*

Under the (reasonable) assumption that $Ax^* \leq b$, all slacks are nonnegative and Proposition 2.1 also implies that the slack of each inequality used must be less than one.

Using this, Caprara and Fischetti [2] show that the separation problem for $\{0, \frac{1}{2}\}$-cuts is strongly $\mathcal{NP}$-hard in general but polynomially solvable in certain special cases. One of these special cases is of interest for this paper and to present it, we need two more definitions:

DEFINITION 2.2. The *mod-2 support* of an integer matrix A, denoted by $\bar{A}$, is the matrix obtained by replacing each entry in A by its parity (0 if even, 1 if odd).

The mod-2 support of a single inequality is defined analogously.

DEFINITION 2.3. A $p \times q$ binary matrix A is called an *edge-path incidence matrix of a tree* (EPT for short) if there is a tree T on $p+1$ vertices such that each row of A corresponds to an edge of T and each column c of A is the incidence vector of edges of a path P_c in T.

The result in Caprara and Fischetti [2] which we need is then:

THEOREM 2.1 (CAPRARA AND FISCHETTI [2]). *The separation problem for $\{0, \frac{1}{2}\}$-cuts for a system $Ax \leq b$, $x \geq 0$ can be solved in polynomial time if $\bar{A}$ is EPT.*

The separation algorithm is essentially an extension of the Padberg-Rao [30] algorithm, which as mentioned above is based on the computation of a minimum weight odd cut in a suitable weighted labelled graph. Given the EPT matrix A and corresponding tree T, one defines a graph $G[A]$ on $p+1$ vertices, containing the edges of T together with an edge e_c for each column of A (with the same end points as P_c). The edges of T are in one-to-one correspondence with the inequalities in the system $Ax \le b$; the edges of $G[A]\backslash T$ are in one-to-one correspondence with the nonnegativity inequalities. Each edge of $G[A]$ is labelled *odd* or *even* according to whether the right-hand side of the associated inequality is odd or even, and is given a weight equal to the slack of the associated inequality (computed with respect to x^*). Then, there is a one-to-one correspondence between odd cuts in $G[A]$ and $\{0, \frac{1}{2}\}$-cuts for the original problem, and every odd cut of weight less than one yields a violated $\{0, \frac{1}{2}\}$-cut.

We will call the pair $(G[A], T)$ the *witness* for A.

The reason that these results are of relevance is that the comb inequalities (and certain more general inequalities such as the *extended* comb inequalities of Naddef and Rinaldi [24]) can be derived as $\{0, \frac{1}{2}\}$-cuts from the degree equations and SECs; see Caprara et al. [4] for details. In fact, as pointed out in Letchford [19], to derive the comb inequalities as $\{0, \frac{1}{2}\}$-cuts it suffices to use, together with the degree equations, a certain *weakened version* of the SECs as expressed in the following propositions and definition:

Proposition 2.2 (Letchford [19]). *Let $S \subset V$ and $T \subset V$ be disjoint, nonempty vertex sets such that $S \cup T \neq V$. Summing together the SECs on S, T, and $S \cup T$ yields the following inequality*:

$$2x(E(S)) + 2x(E(T)) + x(E(S:T)) \le 2|S| + 2|T| - 3, \tag{7}$$

where $E(S:T)$ denotes the set of edges with one endnode in S and the other endnode in T.

Inequalities of the form in Equation (7) are called *tooth* inequalities in Letchford [20].

Definition 2.4 (Letchford [19]). A *domino parity* (DP) inequality is a valid inequality for the TSP which can be derived as a $\{0, \frac{1}{2}\}$-cut from the degree equations (Equation 1) and the tooth inequalities (Equation 7).

Proposition 2.3 (Letchford [19]). *The DP inequalities are a proper generalization of the comb inequalities in the sense that every comb inequality is a DP inequality, yet there are facet-inducing DP inequalities which are not comb inequalities.*

For more results on domino parity inequalities see, for example, Naddef and Wild [27] and Cook et al. [7].

In this paper, we restrict our attention to *simple* comb inequalities. To derive these as $\{0, \frac{1}{2}\}$-cuts, it suffices to use a subclass of the tooth inequalities as expressed in the following definition and proposition:

Definition 2.5. A tooth inequality is *simple* if $|T| = 1$. It takes the form

$$2x(E(S)) + x(E(i:S)) \le 2|S| - 1, \tag{8}$$

where $S \subset V$ satisfies $1 \le |S| \le |V| - 2$, $i \in V\backslash S$, and where $E(i:S)$ denotes the set of edges with i as one endnode and the other endnode in S. The vertex i is the *root* of the tooth and the vertex set S the *body*.

Proposition 2.4. *Simple comb inequalities can be derived as $\{0, \frac{1}{2}\}$-cuts from the degree equations* (*Equation* 1) *and the simple tooth inequalities* (*Equation* 8).

Proof. First, sum together the degree equations for all $i \in H$ to obtain:

$$2x(E(H)) + x(\delta(H)) \le 2|H|. \tag{9}$$

Now suppose, without loss of generality, that there is some $1 \le k \le t$ such that $|T_j \cap H| = 1$ for $j = 1, \ldots, k$, and $|T_j \backslash H| = 1$ for $k+1, \ldots, t$. For $j = 1, \ldots, k$, associate a simple tooth inequality of the form (Equation 8) with tooth T_j by setting $\{i\} := T_j \cap H$ and $S := T_j \backslash H$. Similarly, for $j = k+1, \ldots, t$, associate a simple tooth inequality with tooth T_j by setting $\{i\} := T_j \backslash H$ and $S := T_j \cap H$. Add all of these simple tooth inequalities to Equation (9) to obtain:

$$\begin{aligned} 2x(E(H)) + x(\delta(H)) + \sum_{j=1}^{k}(2x(E(T_j\backslash H)) + x(E(T_j \cap H : T_j\backslash H))) \\ + \sum_{j=k+1}^{t}(2x(E(T_j \cap H)) + x(E(T_j \cap H : T_j \backslash H))) \le 2|H| + 2\sum_{j=1}^{t}|T_j| - 3t. \end{aligned}$$

This can be rearranged to give:

$$2x(E(H)) + 2\sum_{j=1}^{t} x(E(T_j)) + x\left(\delta(H)\backslash \bigcup_{j=1}^{t} E(T_j \cap H : T_j\backslash H)\right) \le 2|H| + 2\sum_{j=1}^{t}|T_j| - 3t.$$

Dividing by two and rounding down yields Equation (5). □

Let us call the inequalities which can be derived as $\{0, \frac{1}{2}\}$-cuts from the degree equations and simple tooth inequalities *simple* DP inequalities. Then, the above proposition states that every simple comb inequality is a simple DP inequality. It is interesting to note that the SECs (Equation 2) can themselves be regarded as simple DP inequalities, obtained by dividing a single simple tooth inequality by two and rounding down. In the conference version of this paper (Letchford and Lodi [21]), it was conjectured that every simple DP inequality is equivalent to or dominated by SECs and simple comb inequalities. However, this is false. A counterexample with nine vertices is displayed in Figures 2 and 3.

The violated simple DP inequality is derived using the degree equations for $\{1, 2, 3, 4, 5\}$ and five tooth inequalities: the first four teeth $\{1, 8\}, \{4, 6\}, \{5, 7\}, \{3, 9\}$ are just edges, while the fifth tooth $\{2, 4, 5, 6, 7\}$ has vertex 2 as root and the set $\{4, 5, 6, 7\}$ as body. This structure differs from simple comb inequalities in two respects: some teeth are nested inside other teeth and the body of one tooth crosses the handle, which is defined as the set of vertices for which the degree constraints are included in the derivation.

The resulting simple DP inequality is

$$x(E(H)) + \sum_i x(E(T_i)) - x_{24} - x_{25} \leq 10.$$

The point depicted in Figure 2 has left-hand-side value of 10.33. It is a vertex of the polytope that is described by $x_{18} = x_{39} = x_{46} = x_{57} = 1$, $x_e = 0$ if e is not displayed in Figure 2, all degree constraints, the subtour constraint on $\{2, 4, 5, 6, 7\}$, and the two comb inequalities implied by the following sets of handles and teeth: $H1 = \{1, 2, 3\}$, $T1a = \{1, 8\}$, $T1b = \{3, 9\}$, $T1c = \{2, 4, 5, 6, 7\}$, and $H2 = \{2, 6, 7\}$, $T2a = \{4, 6\}$, $T2b = \{5, 7\}$, $T2c = \{1, 2, 3, 8, 9\}$.

Interestingly, this inequality is not facet inducing although it induces a face of high dimension. (It can be made into a facet by increasing the left-hand side coefficients of x_{48} and x_{58} from zero to one, but the resulting inequality is not a $\{0, \frac{1}{2}\}$ cut.) Indeed, results of Naddef and Wild [27] imply that the simple comb inequalities are the only *facet-inducing* simple DP inequalities.

In this paper, then, we actually give a separation algorithm for simple DP inequalities, which include the simple comb inequalities as a special case. To aid the reader, we display in Figure 4 the relationships between all of the inequalities discussed so far. An arrow from one class to another means that the former is a proper generalization of the latter.

3. The structure of candidate teeth. Our goal in this paper is to apply the results of Caprara and Fischetti [2] to yield a polynomial-time separation algorithm for simple DP inequalities. However, a problem which immediately presents itself is that there is an exponential number of simple tooth inequalities and, therefore, the system $Ax \leq b$ defined by the degree and simple tooth inequalities is of exponential size.

Fortunately, Proposition 2.1 tells us that we can restrict our attention to simple tooth inequalities whose slack is less than one without losing any violated $\{0, \frac{1}{2}\}$-cuts. Such tooth inequalities are polynomial in number, as shown in the following two lemmas.

LEMMA 3.1. *Suppose that $x^* \in \mathrm{SEP}(n)$. Then, the number of sets whose SECs have slack less than $\frac{1}{2}$ is $\mathcal{O}(n^2)$ and these sets can be found in $\mathcal{O}(nm(m + n \log n))$ time.*

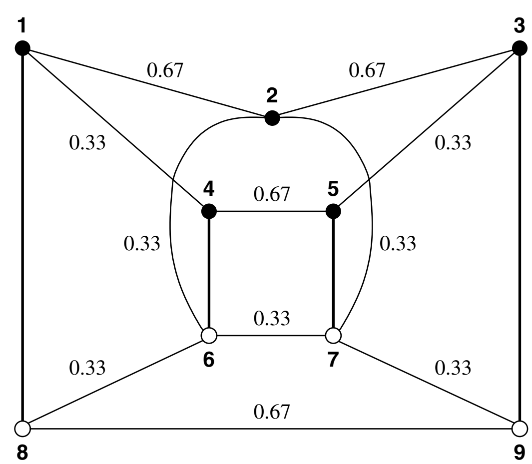

FIGURE 2. A point inside the subtour polytope on nine vertices for which there is a violated simple DP inequality but no violated comb inequalities. The unmarked dark lines have weight one.

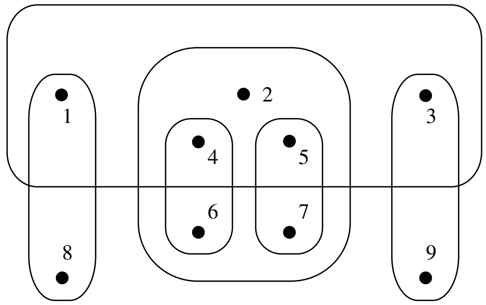

FIGURE 3. A schematic diagram of a simple DP inequality which is neither an SEC nor a simple comb inequality.

PROOF. The degree equations can be used to show that the slack of the SEC on a set S is less than $\frac{1}{2}$ if and only if $x^*(\delta(S)) < 3$. Since the minimum cut in G^* has weight two, we require that the cut-set $\delta(S)$ has a weight strictly less than $\frac{3}{2}$ times the weight of the minimum cut. It is known (Henzinger and Williamson [17]) that there are $\mathcal{O}(n^2)$ such sets and that the algorithm of Nagamochi et al. [28] finds them in $\mathcal{O}(nm(m+n\log n))$ time. □

LEMMA 3.2. *Suppose that $x^* \in \mathrm{SEP}(n)$. Then, the number of distinct simple tooth inequalities with slack less than one is $\mathcal{O}(n^3)$, and these teeth can be found in $\mathcal{O}(nm(m+n\log n))$ time.*

PROOF. The slack of the tooth inequality is equal to the slack of the SEC for S plus the slack of the SEC for $\{i\}\cup S$. For the tooth inequality to have slack less than one, the slack for at least one of these SECs must be less than $\frac{1}{2}$. So, we can take each of the $\mathcal{O}(n^2)$ sets mentioned in Lemma 3.1 and consider them as candidates for either S or $\{i\}\cup S$. For each candidate, there are only n possibilities for the vertex i. The time bottleneck is easily seen to be the Nagamochi et al. [28] algorithm. □

Now, consider the system of inequalities $Ax \le b$ formed by the degree equations and the $\mathcal{O}(n^3)$ simple tooth inequalities mentioned in Lemma 3.2. If we could show that the mod-2 support of A is always an EPT matrix, then we would be done. Unfortunately, this is *not* the case. (It is easy to produce counterexamples even for $n=6$.)

Therefore, we must use a more involved argument if we wish to solve the separation problem for simple DP inequalities via $\{0,\frac{1}{2}\}$-cut arguments. It turns out that the key is to pay special attention to simple tooth inequalities whose slack is strictly less than $\frac{1}{2}$. This leads us to the following definitions and lemma:

DEFINITION 3.1. A tooth is said to be *light* if the slack of the associated tooth inequality (computed with respect to x^*) is less than $\frac{1}{2}$. If the slack is at least $\frac{1}{2}$ but less than one, it is said to be *heavy*. For a given root $i\in V$, a vertex set $S\subset V\backslash\{i\}$ is said to be *i-light* if the slack of the tooth inequality with root i and body S has slack strictly less than $\frac{1}{2}$. If the slack is at least $\frac{1}{2}$, but less than one, it is said to be *i-heavy*.

LEMMA 3.3. *If a simple DP inequality is violated by a given $x^* \in \mathrm{SEP}(n)$, then at most one of its teeth can be heavy and the others are light.*

PROOF. If two of the teeth are heavy, the slacks of the associated tooth inequalities sum to at least $\frac{1}{2}+\frac{1}{2}=1$. Then, by Proposition 2.1 the DP inequality is not violated. □

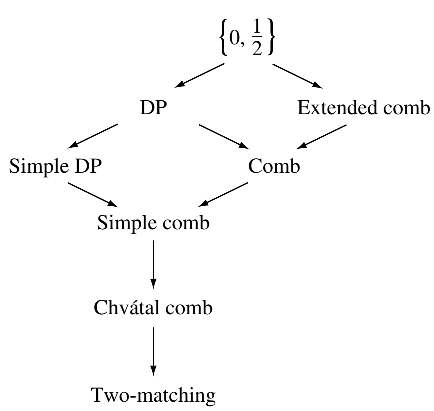

FIGURE 4. Relationships between various valid inequalities.

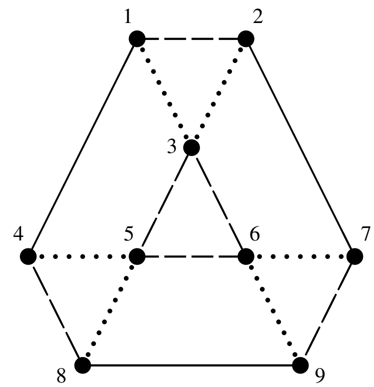

FIGURE 5. A fractional point contained in SEP(9).

We illustrate these ideas on a small example.

EXAMPLE. Figure 5 shows the support graph G^* for a vector x^* which lies in SEP(9). The solid lines, dashed lines, and dotted lines show edges with $x^*_e = 1$, 2/3, and 1/3, respectively. The 1-light sets are $\{2\}, \{4\}, \{2, 7\}, \{3, 4, 5, 6, 8, 9\}, \{3, \ldots, 9\}$, and $\{2, 3, 5, 6, 7, 8, 9\}$; the 3-light sets are $\{5\}, \{6\}, \{5, 6\}, \{1, 2, 4, 7, 8, 9\}, \{1, 2, 4, 5, 7, 8, 9\}$, and $\{1, 2, 4, 6, 7, 8, 9\}$. The 1-heavy sets are $\{3\}, \{4, 8\}, \{3, 5, 6\}, \{2, 4, 7, 8, 9\}, \{2, 3, 5, 6, 7, 9\}$, and $\{2, 4, 5, 6, 7, 8, 9\}$, while the 3-heavy sets are $\{1\}, \{2\}, \{1, 4\}, \{2, 7\}, \{2, 5, 6, 7, 8, 9\}, \{1, 4, 5, 6, 8, 9\}, \{2, 4, 5, 6, 7, 8, 9\}$, and $\{1, 4, 5, 6, 7, 8, 9\}$. The reader can easily identify light and heavy sets for other roots by exploiting the symmetry of the fractional point.

The light sets have an interesting structure, as expressed in the following definition and theorem:

DEFINITION 3.2. Let $i \in V$ be a fixed root. Two vertex sets $S_1, S_2 \subset V \backslash \{i\}$ are said to *i-cross* if each of the four sets $S_1 \cap S_2$, $S_1 \backslash S_2$, $S_2 \backslash S_1$, and $V \backslash (S_1 \cup S_2 \cup \{i\})$ is nonempty.

THEOREM 3.1. *Let $i \in V$ be a fixed root. If $x^* \in \mathrm{SEP}(n)$, it is impossible for two i-light sets to i-cross.*

PROOF. If we sum together the degree equations (Equation 1) for all $j \in S_1 \cap S_2$ along with the SECs on the four vertex sets $i \cup S_1 \cup S_2$, $S_1 \backslash S_2$, $S_2 \backslash S_1$, and $S_1 \cap S_2$, then (after some rearranging) we obtain the inequality:

$$x^*(E(i:S_1)) + 2x^*(E(S_1)) + x^*(E(i:S_2)) + 2x^*(E(S_2)) \le 2|S_1| + 2|S_2| - 3. \tag{10}$$

On the other hand, the sum of the tooth inequality with root i and body S_1 and the tooth inequality with root i and body S_2 is:

$$x^*(E(i:S_1)) + 2x^*(E(S_1)) + x^*(E(i:S_2)) + 2x^*(E(S_2)) \le 2|S_1| + 2|S_2| - 2. \tag{11}$$

Comparing inequalities (11) and (10), we see that the sum of the slacks of these two tooth inequalities is at least one. Since $x^* \in \mathrm{SEP}(n)$, each of the individual slacks is nonnegative. Hence, at least one of the slacks must be $\ge \frac{1}{2}$. That is, at least one of S_1 and S_2 is i-heavy. $\square$

The following lemma shows that we can eliminate half of the i-light sets from consideration.

LEMMA 3.4. *A tooth inequality with root i and body S is equivalent to the tooth inequality with root i and body $V \backslash (S \cup \{i\})$.*

PROOF. The latter inequality can be obtained from the former by subtracting the degree equations for the vertices in S and adding the degree equations for the vertices in $V \backslash (S \cup \{i\})$. $\square$

A set ψ of subsets of V is *laminar* if for all $S, T \in \psi$, at least one of the following three sets is empty: $S \cap T$, $S \backslash T$, $T \backslash S$. Together, Theorem 3.1 and Lemma 3.4 imply the following corollaries.

COROLLARY 3.1. *For a given root i, the bodies of i-light sets may be chosen to be laminar. Thus, there are only $\mathcal{O}(n)$ i-light vertex sets.*

COROLLARY 3.2. *There are only $\mathcal{O}(n^2)$ light teeth.*

4. Separation. Our separation algorithm has two stages. In the first stage, we search for a violated simple DP inequality in which all of the teeth are light. If this fails, then we proceed to the second stage, where we search for a violated simple DP inequality in which one of the teeth is heavy. Lemma 3.3 in the previous section shows that this approach is valid.

We will need the following lemma:

LEMMA 4.1. *Let i be an arbitrary root and let $Ax \leq b$ be the inequality system formed by the degree equation on i (written in less-than-or-equal-to form) and the tooth inequalities whose bodies form a laminar set ψ. Then, the mod-two support of the matrix A is an EPT matrix.*

PROOF. We show how to construct a tree T such that the mod-2 support of A is an edge-path incidence matrix of T. Suppose $|\psi| = r$. The tree will have $r+2$ nodes, numbered $v_1, \ldots, v_{r+2}$, and $r+1$ edges (one for each body in ψ plus an extra one for the degree equation). For $S_1, S_2 \in \psi$, S_2 is the *parent* of body S_1 if $S_1 \subset S_2$ but there is no third set $S_3 \in \psi$ with $S_1 \subset S_3 \subset S_2$. We construct our tree as follows: If the pth body in the family has the qth body as parent, we connect vertex v_p to vertex v_q by an edge. If the pth body has no parent, we connect vertex v_p to vertex v_{r+1} by an edge. In either case, the edge added represents the tooth inequality with root i and body p. Finally, we connect vertex v_{r+1} to vertex v_{r+2} by an edge, which represents the degree equation. To see that the mod-2 support of A is an edge-path incidence matrix of T, note that if the variable x_e receives an odd coefficient in the tooth inequality with root i and body S, and S' is the parent of S, then x_e also receives an odd coefficient in the tooth inequality with root i and body S' and also in the degree equation for i. Hence, a column of A either consists of zeroes and twos (when the associated edge $e \in E \setminus \delta(\{i\})$), or is the characteristic vector of a path in T ending at vertex $r+2$. □

EXAMPLE (CONTINUED). The 1-light sets which do not include vertex 9 are $\{2\}$, $\{4\}$, and $\{2, 7\}$. The third set $\{2, 7\}$ is the parent of the first $\{2\}$. The associated tooth inequalities are $x_{12} \leq 1$, $x_{14} \leq 1$, and $2x_{27} + x_{12} + x_{17} \leq 1$. The corresponding tree is shown in Figure 6. It can be seen, for example, that the column of A associated with variable x_{12} is the incidence vector of the path from vertex v_1 to vertex v_5 in the tree.

We are now in a position to state an important theorem, which is at the heart of our separation algorithm:

THEOREM 4.1. *Let $A'x \leq b'$ be the inequality system formed by the degree equations (written in less-than-or-equal-to form) and a set of tooth inequalities such that, for each root i, the corresponding set of bodies form a laminar set. Then, the mod-2 support of the matrix A' is an EPT matrix.*

PROOF. The inequality system $A'x \leq b'$ is the union of n inequality systems of the form given in Lemma 4.1, one for each root i. We already know that each of these inequality systems can be represented by a tree. Moreover, in each of these trees, the edge representing the degree equation is incident on a leaf vertex (called v_{r+2} in Lemma 4.1). Take each of the n trees and form a single larger tree by identifying each of these leaf vertices to form a single vertex v^*. Note that a variable x_{ij} has an odd coefficient in exactly *two* of the n smaller inequality systems, namely the ones associated with the roots i and j. This means that the mod-2 support of the associated column of A' is the incidence vector of a path in exactly two subtrees. However, each of the paths ends at v^* because x_{ij} has an odd coefficient in the degree equation for i and j. Hence, these two paths form a single larger path in the large tree, passing through v^*. So, each column of A' is the incidence vector of a path in the larger tree. □

EXAMPLE (CONTINUED). There are six i-light sets for each root. Applying Lemma 3.4, we can eliminate half of these from consideration. So, suppose we choose:
- 1-light sets: $\{2\}, \{4\}, \{2, 7\}$;
- 2-light sets: $\{1\}, \{7\}, \{1, 4\}$;
- 3-light sets: $\{5\}, \{6\}, \{5, 6\}$;
- 4-light sets: $\{1\}, \{8\}, \{8, 9\}$;

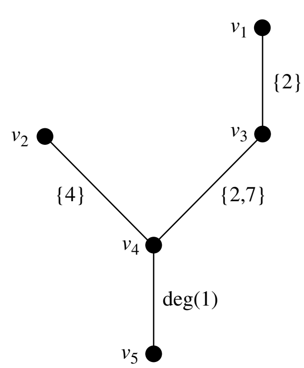

FIGURE 6. Tree in illustration of Lemma 4.1.

- 5-light sets: $\{3\}$, $\{6\}$, $\{3, 6\}$;
- 6-light sets: $\{3\}$, $\{5\}$, $\{3, 5\}$;
- 7-light sets: $\{2\}$, $\{9\}$, $\{8, 9\}$;
- 8-light sets: $\{4\}$, $\{9\}$, $\{1, 4\}$;
- 9-light sets: $\{7\}$, $\{8\}$, $\{2, 7\}$.

This leads to 27 light tooth inequalities in total. However, there are some duplicates: A tooth inequality with root i and body $\{j\}$ is identical to a tooth inequality with root j and body $\{i\}$ (in both cases, the inequality is a simple upper bound $x_{ij} \le 1$). In fact, there are only 18 distinct inequalities, namely:

$$\begin{array}{lll} 2x_{27}+x_{12}+x_{17}\le 3 & 2x_{14}+x_{12}+x_{24}\le 3 & 2x_{56}+x_{35}+x_{36}\le 3 \\ 2x_{89}+x_{48}+x_{49}\le 3 & 2x_{36}+x_{35}+x_{56}\le 3 & 2x_{35}+x_{36}+x_{56}\le 3 \\ 2x_{89}+x_{78}+x_{79}\le 3 & 2x_{14}+x_{18}+x_{48}\le 3 & 2x_{27}+x_{29}+x_{79}\le 3, \end{array}$$

plus the upper bounds on $x_{12}, x_{14}, x_{27}, x_{35}, x_{36}, x_{48}, x_{56}, x_{79}$, and x_{89}. Therefore, the matrix A' has 36 columns (one for each variable) and 27 rows (18 tooth inequalities plus 9 degree equations). The single large tree is shown in Figure 7. The edge associated with the ith degree equation is labelled d_i. The edge associated with the upper bound $x_{ij} \le 1$ is labelled u_{ij}. The edges associated with the remaining nine light tooth inequalities are labelled with the root and body. The vertex at the centre of the tree, incident on the edges labelled $d_1, \ldots, d_9$, is the vertex v^* mentioned in the proof of Theorem 4.1. Note that many other compatible trees can be formed by moving the edges representing the upper bounds. For example, the edge marked u_{12} could be moved to the right, making it adjacent to the edge marked $2, \{1, 4\}$.

Let $\mathcal{S}$ be the set of teeth such that the corresponding tooth inequalities (taken modulo two) together with the degree constraints form an EPT matrix $A_{\mathcal{S}}$. Let $(G_{\mathcal{S}}, T_{\mathcal{S}})$ be the witness for $A_{\mathcal{S}}$. Theorem 4.1 has an important corollary:

Corollary 4.1. *A simple DP inequality derived from tooth inequalities in the set $\mathcal{S}$, degree constraints, and nonnegativity inequalities is violated by a point in the subtour polytope if and only if the corresponding edges form an odd cutset in $G_{\mathcal{S}}$ with weight less than one.*

A core component of our separation algorithm is the subroutine $\mathsf{buildT}(\mathcal{S})$ that builds the witness $(G_{\mathcal{S}}, T_{\mathcal{S}})$. This is described in Figure 8.

Lemma 4.2. *Given $\mathcal{S}$, the subroutine* $\mathsf{buildT}(\mathcal{S})$ *runs in $O(n|\mathcal{S}| + m)$ time.*

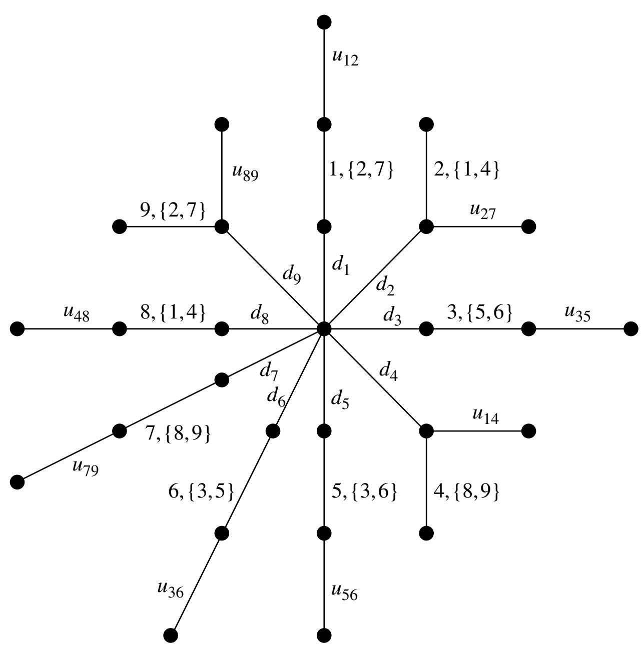

Figure 7. Tree in demonstration of Theorem 4.1.

buildT($\mathcal{S}$)

1 $T_{\mathcal{S}} \leftarrow \{r_0\}$.
2 Sort $\mathcal{S}$ according to root node, creating partition $\{\mathcal{S}_i\}_{i=1}^n$.
3 For each i, build subtree corresponding to $\mathcal{S}_i$:
4   Add edge (r_i, r_0) to $T_{\mathcal{S}}$.
5   Sort bodies of teeth in $\mathcal{S}_i$ by decreasing size: $|S_1| \geq |S_2| \geq \cdots$
6   For each set S_l,
7   If $\exists$ smallest set S_k with $S_l \subset S_k$, add (v_l, v_k) to $T_{\mathcal{S}}$.
8   If no such S_k, add edge (v_l, r_i) to $T_{\mathcal{S}}$.
9 $G_{\mathcal{S}} \leftarrow T_{\mathcal{S}}$.
10 For each $(i, j) \in E^*$,
11     $\min_j^i \leftarrow$ vertex in subtree of r_i for smallest set S containing j.
12   Add $(\min_j^i, \min_i^j)$ to $G_{\mathcal{S}}$

FIGURE 8. Subroutine that builds the witness $(G_{\mathcal{S}}, T_{\mathcal{S}})$.

PROOF. All steps in this proof refer to Figure 8. Step 2 takes $O(n|\mathcal{S}|)$ time. For each root i, Step 5 takes $O(n|\mathcal{S}_i|)$ time. If updates to $\min_j^i$ are done as sets S_l are added to $T_{\mathcal{S}}$, Steps 7 and 11 take $O(n|\mathcal{S}_i|)$ time over all bodies of root i. Over all roots, this takes $O(n|\mathcal{S}|)$ time. Adding edges corresponding to E^* takes additional $O(m)$ time. □

COROLLARY 4.2. *If $x^* \in \mathrm{SEP}(n)$, then a violated simple DP inequality which uses only light teeth can be found in polynomial time, if any exists.*

PROOF. This follows from Lemma 3.1, Corollary 4.1, and Lemma 4.2. □

EXAMPLE (CONTINUED). Applying the first stage of the separation algorithm to the fractional point shown in Figure 3, we find a violated simple comb inequality with $H = \{1, 2, 3\}$, $T_1 = \{1, 4\}$, $T_2 = \{2, 7\}$, and $T_3 = \{3, 5, 6\}$. The inequality is

$$x_{12} + x_{13} + x_{23} + x_{14} + x_{27} + x_{35} + x_{36} + x_{56} \leq 5,$$

and it is violated by one-third.

We now proceed to describe stage 2 of the algorithm, in which we search for a violated simple DP inequality in which one used tooth is heavy. The key to this stage is the following lemma:

LEMMA 4.3. *Let x^* be a given fractional point lying in the subtour polytope. If there is a violated simple DP inequality, then there is a simple DP inequality, violated by at least as much such that no used teeth share the same root.*

PROOF. Suppose that two *used* teeth have the same root i and bodies S_1 and S_2. Without loss of generality we can assume that $S_1 \backslash S_2$ and $S_2 \backslash S_1$ are nonempty. (If $S_1 \subset S_2$, then we can replace S_2 by its complement $V \backslash (\{i\} \cup S_2$ and similarly if $S_2 \subset S_1$.)

Our claim is that a $\{0, \frac{1}{2}\}$-cut with at least the same amount of violation can be obtained by reducing the multipliers for the two teeth from $\frac{1}{2}$ to zero, and increasing the multipliers for the degree equation on i (in less-than-or-equal-to form) and, the nonnegativity inequalities on $E(i : V \backslash (\{i\} \cup (S_1 \backslash S_2) \cup (S_2 \backslash S_1))$, by $\frac{1}{2}$.

Note that this change causes the left-hand side of the $\{0, \frac{1}{2}\}$-cut to reduce by $x(E(S_1)) + x(E(S_2)) + x(E(i\colon S_1 \cap S_2))$ and the right-hand side to reduce by $|S_1| + |S_2| - 2$. The change in the amount of violation, computed with respect to x^*, is therefore $|S_1| + |S_2| - 2 - x^*(E(S_1)) - x^*(E(S_2)) - x^*(E(i\colon S_1 \cap S_2))$. This can be rewritten as the sum of three terms:

(a) $|S_1 \backslash S_2| - x^*(E(S_1 \backslash S_2)) - 1$,
(b) $|S_2 \backslash S_1| - x^*(E(S_2 \backslash S_1)) - 1$,
(c) $2|S_1 \cap S_2| - 2x^*(E(S_1 \cap S_2)) - x^*(E(S_1 \cap S_2\colon (S_1 \backslash S_2) \cup (S_2 \backslash S_1) \cup \{i\}))$.

The first of these terms is nonnegative because by assumption x^* satisfies the SEC on $S_1 \backslash S_2$. Similarly, the second term is nonnegative because of the SEC on $S_2 \backslash S_1$. Finally, the third term is nonnegative because of the degree equations on $S_1 \cap S_2$. Thus, the total amount of violation is either unchanged or increased.

The resulting $\{0, \frac{1}{2}\}$-cut may not be a simple DP inequality because some of the multipliers may have increased from $\frac{1}{2}$ to 1. However, we can obtain a still stronger $\{0, \frac{1}{2}\}$-cut by changing any such multipliers to zero. The resulting $\{0, \frac{1}{2}\}$-cut is now a simple DP inequality.

This procedure can be repeated until no two teeth share the same root. □

We can now prove Theorem 1.1. The algorithm is summarized in Figure 9.

SimpleDPSep(G, x^*)

1 Find all simple tooth inequalities with x-slack less than 1.
2 Create subset $\mathcal{L}$ of light simple teeth.
3 $(G_{\mathcal{L}}, T_{\mathcal{L}}) \leftarrow \mathsf{buildT}(\mathcal{L})$.
4 Find a minimum odd cut in $G_{\mathcal{L}}$.
5 If weight of cut is < 1, output inequality.
6 For each root i,
7     $\mathcal{L}_i \leftarrow$ the set of light teeth with root i.
8   For each heavy tooth S with root i,
9         $\mathcal{H} \leftarrow (\mathcal{L} \backslash \mathcal{L}_i) \cup \{S\}$.
10         $(G_{\mathcal{H}}, T_{\mathcal{H}}) \leftarrow \mathsf{buildT}(\mathcal{H})$.
11   Find the minimum odd cut in $G_{\mathcal{H}}$.
12   If weight of cut is < 1, output inequality.

FIGURE 9. Separation for simple domino parity inequalities: Basic algorithm.

PROOF OF THEOREM 1.1. All steps in the proof refer to Figure 9. Corollary 4.2 establishes that Steps 1–5, the separation of simple DP inequalities in which all teeth are light, can be accomplished in polynomial time. It remains to prove the theorem when one of the teeth involved is heavy. From Lemma 3.2, we know that there are $\mathcal{O}(n^3)$ candidates for this heavy tooth. We do the following for each of these candidates: We take the collection of $\mathcal{O}(n^2)$ light tooth inequalities and eliminate the ones whose teeth have the same root as the heavy tooth under consideration. (Lemma 4.3 shows that this is a valid operation.) It is easy to show that the resulting modified matrix is still an EPT matrix: One of the subtrees is removed from the larger tree and replaced by a single edge representing the heavy tooth inequality. The minimum odd cut procedure can then be repeated on this modified graph. □

Now let us analyse the running time of this separation algorithm. Stage 1 involves computing a minimum-weight odd cut in a labelled weighted graph with $\mathcal{O}(n^2)$ vertices and $\mathcal{O}(n^2)$ edges (because there are $\mathcal{O}(n^2)$ light tooth inequalities). Lemmas 3.1 and 4.2 shows this graph can be built in $\mathcal{O}(nm(m + n\log n))$ time. Using the Padberg–Rao algorithm [30], this means solving $\mathcal{O}(n^2)$ max-flow problems in this graph. Using the preflow push algorithm (Goldberg and Tarjan [13]) to solve the max-flow problems, stage 1 takes $\mathcal{O}(n^6 \log n)$ time. This is bad enough, but an even bigger running time is needed for stage 2, which involves $\mathcal{O}(n^3)$ minimum odd cut computations on graphs of a similar size. This leads to a running time of $\mathcal{O}(n^9 \log n)$ for stage 2 which, though polynomial, is totally impractical.

In the next section, we show that this running time can be reduced to $\mathcal{O}(n^2 m^2 \log(n^2/m))$.

5. Improving the running time.

In this section, we prove two theorems that allow us to reduce the complexity of our separation algorithm. The first theorem implies that it is sufficient to consider a set of light teeth of size $\mathcal{O}(n)$. The second theorem implies that it is sufficient to consider a set of heavy teeth of size $\mathcal{O}(nm)$ that has a special structure. The proofs are contained respectively in §§5.1 and 5.2.

THEOREM 5.1. *There exists a set of light teeth $\mathcal{L}$ of size $\mathcal{O}(n)$ such that if there exists a violated simple DP inequality, then there exists one with light teeth from the set $\mathcal{L}$ only. The set $\mathcal{L}$ can be found in $\mathcal{O}(n^3 m)$ time.*

Let $|\delta(i)|$ denote the degree of node i in G^*.

THEOREM 5.2. *There is a set of heavy teeth of size $\mathcal{O}(nm)$ such that the bodies of the heavy teeth with root i can be partitioned into $|\delta(i)|$ laminar subsets; and if there exists a violated simple DP inequality derived using a heavy tooth, then there exists one with its heavy tooth in this set. This set can be found in $\mathcal{O}(n^3 m)$ time.*

We use these two theorems to modify SimpleDPSep to obtain a faster separation algorithm described in Figure 10. As before, the algorithm first looks for simple DP inequalities that are derived using light teeth only. Then, it looks for simple DP inequalities that use one heavy tooth. By Theorem 5.2, Corollary 4.1, and the following Lemma 4.3, this can be done as follows: for each $i \in V$, consider at one time all light teeth with roots in $V\backslash\{i\}$ and a subset of heavy teeth with root i, and check the corresponding graph obtained using the subroutine buildT for a minimum odd cut.

THEOREM 5.3. FastSimpleDPSep *is a separation algorithm for simple domino parity inequalities that runs in $\mathcal{O}(n^2 m^2 \log(n^2/m))$ time.*

FastSimpleDPSep(G, x^*)

1 Find all simple tooth inequalities with x-slack less than one.
2 Reduce the subset of light teeth to a set $\mathscr{L}$ of size $\mathcal{O}(n)$ by uncrossing (Theorem 5.1).
3 $(G_{\mathscr{L}}, T_{\mathscr{L}}) \leftarrow \mathsf{buildT}(\mathscr{L})$.
4 Find a minimum odd cut in $G_{\mathscr{L}}$.
5 If weight of cut is < 1, output inequality.
6 Reduce the subset of heavy teeth to $\mathcal{O}(m)$ laminar sets (Theorem 5.2).
7 For each root i,
8   $\mathscr{L}_i \leftarrow$ the set of light teeth with root i.
9   Partition the set of heavy teeth with root i into $|\delta(i)|$ laminar subsets, $\{\mathscr{H}_1^i, \ldots, \mathscr{H}_{|\delta(i)|}^i\}$ (Theorem 5.2).
10   For each such subset $\mathscr{H}_j^i$,
11     $\mathscr{H} \leftarrow (\mathscr{L} \backslash \mathscr{L}_i) \cup \mathscr{H}_j^i$.
12     $(G_{\mathscr{H}}, T_{\mathscr{H}}) \leftarrow \mathsf{buildT}(\mathscr{H})$.
13   Find the minimum odd cut in $G_{\mathscr{H}}$.
14   If weight of cut is < 1, output inequality.

FIGURE 10. Separation for simple domino parity inequalities: Fast algorithm.

PROOF. All steps in the proof refer to Figure 10. If there is a violated simple DP inequality using only teeth in $\mathscr{L}$, then by Corollary 4.1 and Theorem 5.1, FastSimpleDPSep finds it in Step 4. Otherwise, using Corollary 4.1, Lemma 4.3, and Theorems 5.1 and 5.2, if there is a violated simple DP inequality using a heavy tooth with root i, FastSimpleDPSep finds it in Step 13. This establishes correctness. Now we focus on run time.

By Lemmas 3.1 and 3.2, we know that the time required for Step 1 is $\mathcal{O}(n^4 + nm(m + n\log n))$, where the first term is for sorting according to root the $\mathcal{O}(n^3)$ teeth found by the Nagamochi et al. [28] (second term).

Theorem 5.1 implies that time required for Step 2 is $\mathcal{O}(n^3 m)$. Lemma 4.2 implies that the total time spent in subroutine buildT over the course of the algorithm (Steps 3 and 12) is at most $\mathcal{O}(nm^2)$.

For the light teeth, this results in a graph $G_{\mathscr{L}}$ containing a tree $T_{\mathscr{L}}$ with a root with n branches and a total number of nodes of $\mathcal{O}(n)$. There is an edge for every edge in the support graph, so the number of edges is $\mathcal{O}(m)$. On this graph, it takes $\mathcal{O}(n^2 m \log(n^2/m))$ time to find a minimum odd cut (Step 4).

By Theorem 5.2, Steps 7–9 take total time at most $\mathcal{O}(n^3 m)$. Steps 10–11 are not a bottleneck. To find violated inequalities that use a heavy tooth with root i, the graph $G_{\mathscr{H}}$ still contains $\mathcal{O}(n)$ nodes (nodes from the original $\mathcal{O}(n)$ light teeth plus $\mathcal{O}(n)$ nodes for the laminar heavy teeth with root i) and $\mathcal{O}(m)$ edges. By Theorem 5.2, it suffices to consider $\mathcal{O}(m)$ of these graphs. Thus, the time spent on these graphs is $\mathcal{O}(n^2 m^2 \log(n^2/m))$ (Step 13).

Given an odd cut in $G_{\mathscr{H}}$, the corresponding inequality can be recovered in time $\mathcal{O}(n)$ times the size of the cut, i.e., in $\mathcal{O}(nm)$ time (Steps 5 and 14). $\square$

5.1. Proof of Theorem 5.1. The proof of Theorem 5.1 works in two phases. In the first phase, we show that the number of roots i that form a light tooth with any fixed body S is at most three. In the second phase, we show that it is possible to obtain a laminar set of bodies such that all light teeth we consider have a body in this set. This implies that the number of bodies we consider is $\mathcal{O}(n)$.

LEMMA 5.1. *At most three distinct light teeth share the same body.*

PROOF. If a body S is light with respect to a vertex i, then the slack of the SEC on $S \cup \{i\}$ must be less than $1/2$, i.e., $x^*(E(S)) + x^*(E(i:S)) > |S| - 1/2$. So, if S were light with respect to four roots we would have:

$$4x^*(E(S)) + x^*(\delta(S)) > 4|S| - 2.$$

The SEC on S implies $-2x^*(E(S)) \geq 2 - 2|S|$. Adding these two inequalities together gives $2x^*(E(S)) + x^*(\delta(S)) > 2|S|$. However, this contradicts the degree equations on the vertices in S. $\square$

Instead of multiplying inequalities in the derivation of a $\{0, \frac{1}{2}\}$ inequality by $\frac{1}{2}$, we can simply add inequalities together and consider the derived inequality modulo two. The $\{0, \frac{1}{2}\}$ inequalities are then inequalities with odd coefficients on the right-hand side and even coefficients on the left-hand side. To obtain even coefficients on the left, for a fixed set of tooth inequalities and degree constraints it may be necessary to add nonnegativity inequalities.

If the tooth inequality for (i, S) is used in the derivation of some $\{0, \frac{1}{2}\}$ inequality, it contributes an odd amount to the right-hand side, and an odd amount to the coefficients of all edges in $E(i:S)$. Thus, replacing (i, S) with (i, T) for some $T \subset S$ changes the parity only of coefficients of edges in $E(i: S \backslash T)$. If we then add (or remove) the nonnegativity constraints for these edges, no parities are changed and the inequality remains valid. Thus, if the slack of the tooth inequality on (i, S) is at least the slack of the tooth inequality on (i, T) plus the slack of the nonnegativity constraints for $E(i\colon S\backslash T)$, then the tooth inequality on (i, S) can be replaced with the tooth inequality on (i, T). In this situation, we say that (i, T) *improves* (i, S).

Now recall that sets $S, T \subset V$ are said to *cross* if $S \cap T$, $S\backslash T$, $T\backslash S$, and $V\backslash(S \cup T)$ are all nonempty. The next lemma implies that if S and T are both bodies of teeth to be considered for domino parity inequalities and they cross, then it is possible to uncross them without decreasing the violation of any affected simple DP constraint. We use the notation $S'_i = \{V\backslash(S \cup \{i\})\}$ for any (i, S) belonging to a set of light teeth $\mathcal{L}$. When root i can not be confused, we write S' instead of S'_i.

Lemma 5.2. *If (i, S) and (j, T) are two teeth and S and T cross, then either one of the following four conditions holds with strict improvement, or two conditions hold exactly*:

(i) $(i, S\backslash T)$ *or* $(i, S \cap T)$ *improves* (i, S).
(ii) $(i, S'\backslash T)$ *or* $(i, S' \cap T)$ *improves* (i, S').
(iii) $(j, T\backslash S)$ *or* $(j, T \cap S)$ *improves* (j, T).
(iv) $(j, T'\backslash S)$ *or* $(j, T' \cap S)$ *improves* (j, T').

Proof. Suppose X and Y are such that (i, X) and (j, Y) are teeth and X and Y cross. Suppose $i \notin Y$ and $j \notin X$. The slack for (i, X) is

$$\begin{aligned}
&2|X| - 1 - 2x^*(E(X)) - x^*(E(i:X)) \\
&\quad = 2|X\backslash Y| - 1 - 2x^*(E(X\backslash Y)) - x^*(E(i:X\backslash Y)) \\
&\qquad + 2|X \cap Y| - 2x^*(E(X\backslash Y : X \cap Y)) - 2x^*(E(X \cap Y)) - x^*(E(i:X \cap Y)) \\
&\quad = [2|X\backslash Y| - 1 - 2x^*(E(X\backslash Y)) - x^*(E(i:X\backslash Y)) + x^*(E(i:X \cap Y))] \\
&\qquad + [2|X \cap Y| - 2x^*(E(\{i\} \cup (X\backslash Y) : X \cap Y)) - 2x^*(E(X \cap Y))],
\end{aligned}$$

where the last expression is obtained by adding and subtracting the term $x^*(E(i : X \cap Y))$. Note that the first bracketed term in this last expression is the sum of the slack on tooth inequality $(i, X\backslash Y)$ plus the slack on nonnegativity constraints for $E(i : X \cap Y)$. The slack for (j, Y) is

$$\begin{aligned}
&2|Y| - 1 - 2x^*(E(Y)) - x^*(E(j:Y)) \\
&\quad = 2|Y\backslash X| - 1 - 2x^*(E(Y\backslash X)) - x^*(E(j:Y\backslash X)) \\
&\qquad + 2|X \cap Y| - 2x^*(E(Y\backslash X : X \cap Y)) - 2x^*(E(X \cap Y)) - x^*(E(j:X \cap Y)) \\
&\quad = [2|Y\backslash X| - 1 - 2x^*(E(Y\backslash X)) - x^*(E(j:Y\backslash X)) + x^*(E(j:X \cap Y))] \\
&\qquad + [2|X \cap Y| - 2x^*(E(\{j\} \cup (Y\backslash X) : X \cap Y)) - 2x^*(E(X \cap Y))].
\end{aligned}$$

By summing degree constraints on $X \cap Y$, we have that $x^*(E(\{i\} \cup (X\backslash Y) : X \cap Y)) + x^*(E(\{j\} \cup (Y\backslash X) : X \cap Y)) + 2x^*(E(X \cap Y)) \leq 2\ |X \cap Y|$. Thus, either at least one of $[2\ |X \cap Y| - 2x^*(E(\{i\} \cup (X\backslash Y) : X \cap Y)) - 2x^*(E(X \cap Y))]$ and $[2\ |X \cap Y| - 2x^*(E(\{j\} \cup (Y\backslash X) : X \cap Y)) - 2x^*(E(X \cap Y))]$ must be positive or both terms equal zero. In conjunction with the above expressions for the slack of inequalities for (i, X) and (j, Y), if we let $X = S$ and $Y = T$, this implies that either (i) or (iii) holds strictly or both hold exactly.

If $i \notin T$, $j \in S$, then let $X = S'$ and $Y = T$ to get that either (ii) or (iii) holds strictly or both hold exactly.

If $i \in T$, $j \notin S$, then let $X = S$ and $Y = T'$ to get that either (i) or (iv) holds strictly or both hold exactly.

If $i \in T$, $j \in S$, then let $X = S'$ and $Y = T'$ to get that either (ii) or (iv) holds strictly or both hold exactly. □

The next lemma describes why uncrossing teeth is useful in bounding $\mathcal{L}$.

Lemma 5.3. *Let $\mathcal{L}$ be a set of light teeth that satisfies the following property. For all pairs (i, S) and (j, T) in $\mathcal{L}$, at most one of the following pairs of bodies cross*: $(S, T), (S, T'_j), (S'_i, T), (S'_i, T'_j)$. *Then, the size of $\mathcal{L}$ is $O(n)$.*

Proof. Construct the following graph: There are $|\mathcal{L}|$ pairs of vertices. Each pair corresponds to a tooth in $\mathcal{L}$. For tooth (i, S) the first vertex corresponds to S and the second corresponds to S'_i. There is an edge joining vertices for S and S'_i. There is also an edge joining each pair of vertices that correspond to bodies that cross. An independent set I in this graph corresponds to a set of laminar bodies. Thus, any such set has size $O(n)$.

Starting with $\mathcal{K} = \varnothing$, we select a maximal set $\mathcal{K}$ as follows. For each root i in turn, let $\mathcal{T}_i$ be the set of teeth with root i. Select a set of bodies $\mathcal{B}_i$ such that (i) each tooth in $\mathcal{T}_i$ has either its body or complement body in $\mathcal{B}_i$; (ii) if a body or complement body of a tooth in $\mathcal{T}_i$ is already in $\mathcal{K}$, then it is in $\mathcal{B}_i$; and (iii) $\mathcal{B}_i$ is a laminar set (Corollary 3.1). Let $\overline{B_i}$ be the set of complement bodies. Then, add to $\mathcal{K}$ the bodies in $\mathcal{B}_i$ that do not cross bodies already in $\mathcal{K}$ (ignoring duplicates).

Claim. For each tooth (i, S) in $\mathcal{L}$, either S or S' is in $\mathcal{K}$, or $\mathcal{K}$ contains some body T with T' equal to S or S'.

Proof of Claim. For tooth $(i, S) \in \mathcal{L}$, suppose neither the vertex for S nor S' are in $\mathcal{K}$. Without loss of generality, assume $S \in \mathcal{B}_i$. Then, there is a tooth $(j, T) \in \mathcal{L}$ that is added to $\mathcal{K}$ before (i, S) is considered, with $j \neq i$ such that T crosses S.

Since S crosses T, either $S' \subset T$ or $i \in T$ but not both (otherwise $S \cap T = V$). If $i \in T$, then since S and T cross but S' and T do not, $T \backslash S = \{i\}$. In turn, this implies that $S \backslash T = \{j\}$, since all three of the following conditions also hold: S and T' do not cross, $S' \neq \{j\}$, and $S \cup T \neq V$. However, $T \backslash S = \{i\}$ and $S \backslash T = \{j\}$ together imply that $S' = T'$.

If $S' \subset T$, then the argument in the previous paragraph implies that $i \notin T$ and $j \notin S$. However, $j \notin S'$ then implies that $i = j$—contradicting the laminarity of bodies in $\mathcal{B}_i$. This ends the proof of the claim.

Let $\mathcal{K}'$ be the union of $\mathcal{K}$ and the complements of all the bodies in $\mathcal{K}$. Note that $|\mathcal{K}'| \leq 2|\mathcal{K}|$. Now the claim implies that every tooth in $\mathcal{L}$ has a body in $\mathcal{K}$. Then, from Lemma 5.1 it follows that $|\mathcal{L}| = O(n)$. □

We say that tooth (i, S) *t-crosses* tooth (j, T) if either S or S'_i crosses both T and T'_j **or** either T or T'_j crosses both S and S'_i. Since i can be in at most one of T and T', this implies that (i, S) and (j, T) do not t-cross if and only if at most one of the following pairs of bodies cross: (S, T), (S, T'_j), (S'_i, T), and (S'_i, T'_j). If two teeth t-cross, we can apply Lemma 5.2 to uncross them.

Lemma 5.4. *There is a set of light teeth $\mathcal{L}$ with $|\mathcal{L}| = \mathcal{O}(n)$ such that if there is a violated simple DP inequality derived using tooth inequalities of light teeth only, then there is a simple DP inequality derived using tooth inequalities from the set $\mathcal{L}$ only. Given laminar sets of i-teeth for all i, the set $\mathcal{L}$ can be found in $\mathcal{O}(n^3 m)$ time.*

Proof. We begin with the set of teeth $\mathcal{L}'$ given by Lemma 3.2 and Corollary 3.1. We will create a new set $\mathcal{L}$ such that each tooth in $\mathcal{L}'$ is improved by some tooth in $\mathcal{L}$. This is done on an incremental basis until all teeth in $\mathcal{L}'$ are improved by some tooth in $\mathcal{L}$. Throughout, we maintain that no two teeth in $\mathcal{L}$ t-cross. Then, by Lemma 5.3, the size of $\mathcal{L}$ is $\mathcal{O}(n)$ throughout the algorithm.

We begin with $\mathcal{L} = \{(1, S) | (1, S) \in \mathcal{L}'\}$. By Theorem 3.1, no two teeth in $\mathcal{L}$ t-cross. Then, for roots $i = 2$ through n, we orient all teeth $(j, T) \in \mathcal{L}$ (by perhaps replacing (j, T) with (j, T'_j)) so that T does not contain i. We consider one by one teeth in $\{(i, S) | S \subset V\} \cap \mathcal{L}'$; and for each such (i, S) we apply Lemma 5.2 to (i, S) and the teeth in $\mathcal{L}$ starting with the teeth with the smallest bodies. With such a procedure, each uncrossing produces a new tooth that does not t-cross any previously uncrossed tooth: We claim that if the bodies of $(j, T) \in \mathcal{L}$ and $(k, U) \in \mathcal{L}$ are nested before uncrossing one of them with (i, S), then after uncrossing, their respective bodies are either still nested or completely disjoint. This is true for the following reason. Without loss of generality, T is contained in U and S does not contain j. After uncrossing (j, T) with (i, S), either $S = S \backslash T$ or $T = T \backslash S$. In both cases, T is still contained in U. Also, in both cases S and T are disjoint. Thus, uncrossing S and U will not affect the relation of U and T. Hence, we end up with a final set of teeth that do not t-cross.

The time to uncross one pair of teeth is $\mathcal{O}(m)$. Since the size of $\mathcal{L}$ is at most $\mathcal{O}(n)$, the number of uncrossings per new tooth in $\mathcal{L}'$ is at most $\mathcal{O}(n)$. The initial size of $\mathcal{L}'$ may be $\mathcal{O}(n^2)$. Thus, the total time taken by this routine is $\mathcal{O}(n^3 m)$. □

The final piece of the proof of Theorem 5.1 involves establishing the time it takes to go from the list of light teeth sorted by root obtained in step (1) of the algorithm to an organized laminar set of bodies for each root. For the proof of Lemma 5.4, all that is needed is that the bodies be sorted according to size. Naively, this takes at most $\mathcal{O}(n^2)$ time per root, or $\mathcal{O}(n^3)$ overall.

5.2. Proof of Theorem 5.2. We begin the proof with a theorem about the structure of *heavy* teeth. Although the i-heavy sets need not be nested, they satisfy a certain 'circular' property.

Theorem 5.4. *Let $i \in V$ be a given root. There is a cyclic ordering of the vertices in $V \backslash \{i\}$ such that each i-heavy set is the union of consecutive vertices in the ordering.*

The full proof is given in the appendix. It is based on the following: Let $j \in V \backslash \{i\}$ be an arbitrary vertex, and let M be a 0-1 matrix whose columns correspond to the vertices in $V \backslash \{i, j\}$ and whose rows are the incidence vectors of the i-heavy sets which do not include j. Due to Lemma 3.4, the theorem is true if and only if the columns of M can be permuted so that in every row of the resulting matrix the 1s occur consecutively. Then, from Theorem 9 of Tucker [33] on matrices with the consecutive 1s property, it suffices to prove five claims that disallow certain arrangements of i-heavy teeth.

Example (Continued). The 1-light sets are $\{2\}$, $\{4\}$, $\{2,7\}$, $\{3,4,5,6,8,9\}$, $\{3,\ldots,9\}$, and $\{2,3,5,6,7,8,9\}$; the 1-heavy sets are $\{3\}$, $\{4,8\}$, $\{3,5,6\}$, $\{2,4,7,8,9\}$, $\{2,3,5,6,7,9\}$, and $\{2,4,5,6,7,8,9\}$. A suitable ordering of $V\backslash\{1\}$ is $3,5,9,8,4,2,7,6$. The 3-light sets are $\{5\}$, $\{6\}$, $\{5,6\}$, $\{1,2,4,7,8,9\}$, $\{1,2,4,5,7,8,9\}$, and $\{1,2,4,6,7,8,9\}$. The 3-heavy sets are $\{1\}$, $\{2\}$, $\{1,4\}$, $\{2,7\}$, $\{2,5,6,7,8,9\}$, $\{1,4,5,6,8,9\}$, $\{2,4,5,6,7,8,9\}$, and $\{1,4,5,6,7,8,9\}$. A suitable ordering of $V\backslash\{3\}$ is $5,6,1,4,2,7,8,9$.

An important corollary of Theorem 5.4 is the following.

Corollary 5.1. *For a fixed root i, the i-heavy sets can be partitioned into $\mathcal{O}(n)$ nested families.*

Proof. Without loss of generality, assume that $i=1$ and that a suitable ordering of $V\backslash\{1\}$ is $2,\ldots,n$. Then, the first nested family includes all sets which contain two but not n, the second includes all those which contain three but not two, and so on. □

This immediately enables us to save a factor of $\mathcal{O}(n)$ in stage 2 of the separation algorithm:

Corollary 5.2. *Only $\mathcal{O}(n^2)$ minimum odd cut computations suffice in stage 2 of the separation algorithm.*

Proof. Instead of performing one minimum odd cut calculation for each heavy tooth, we need only perform one minimum odd cut calculation for each of the $\mathcal{O}(n^2)$ nested families. □

To improve the running time further, we need to exploit the sparsity of the support graph G^*. To this end, we now describe a simple lemma which enables us to eliminate teeth from consideration. We will then prove that after applying the lemma the number of light and heavy teeth is significantly reduced.

Lemma 5.5. *Suppose a violated $\{0,\frac{1}{2}\}$-cut can be derived using the tooth inequality with root i and body S. If there exists a set $S'\subset V\backslash\{i\}$ such that*

- $E(i:S)\cap E^* = E(i:S')\cap E^*$ *and*
- $2|S'|-2x^*(E(S'))-x^*(E(i:S'))\le 2|S|-2x^*(E(S))-x^*(E(i:S))$,

then we can obtain a $\{0,\frac{1}{2}\}$-cut violated by at least as much by replacing the body S with the body S' (and adjusting the set of used nonnegativity inequalities accordingly).

Proof. By Proposition 2.1, we have to consider the net change in the sum of the slacks of the *used* inequalities. The second condition in the lemma simply says that the slack of the tooth inequality with root i and body S' is not greater than the slack of the tooth inequality with root i and body S. Therefore, replacing S with S' causes the sum of the slacks to either remain the same or decrease. Now we consider the *used* nonnegativity inequalities. The only variables to receive an *odd* coefficient in a tooth inequality with root i and body S are those which correspond to edges in $E(i:S)$, and a similar statement holds for S'. So, for the edges in $E(i:(S\backslash S')\cup(S'\backslash S))$, the nonused nonnegativity inequalities must now be used and vice versa. This has no effect on the sum of the slacks, because $E(i:(S\backslash S')\cup(S'\backslash S))\subset E\backslash E^*$ by assumption and the slack of a nonnegativity inequality for an edge in $E\backslash E^*$ is zero. Hence, the total sum of slacks is either unchanged or decreased and the new $\{0,\frac{1}{2}\}$-cut is violated by at least as much as the original. □

The next theorem shows that after Lemma 5.5 is applied, relatively few heavy teeth remain.

Theorem 5.5. *Only $\mathcal{O}(nm)$ heavy tooth inequalities remain after applying the elimination criterion of Lemma 5.5, and these can be partitioned into $\mathcal{O}(m)$ nested families.*

Proof. By the circular property of i-heavy teeth (Theorem 5.4), the sets $E(i:S)\cap m$ also have a circular property. After applying the elimination criterion, there can be at most $|\delta(i)|^2$ i-heavy teeth. So, the total number of heavy tooth inequalities is at most $\sum_{i\in V}|\delta(i)|^2\le n\sum_{i\in V}|\delta(i)|=2nm$. Moreover, the i-heavy sets partition naturally into $|\delta(i)|$ nested families, giving $\sum_{i\in V}|\delta(i)|=2m$ nested families in total. □

To complete the proof of Theorem 5.2, it remains to show that the reduction and reorganization of heavy teeth from $\mathcal{O}(n^3)$ teeth to $\mathcal{O}(m)$ nested families can be accomplished in $\mathcal{O}(n^3m)$ time.

Completion of proof of Theorem 5.2. For each root i, it takes at most $O(n^3)$ time to obtain the circular ordering of $V\backslash\{i\}$. (It takes at most $O(n)$ time per tooth to place each tooth in this circular order.) This also gives a circular ordering of $\delta(i)$. For each tooth, it takes at most $O(m)$ time to calculate the value on the right side of the expression of the second criteria of Lemma 5.5. Then, in $O(|\delta(i)|^2+n^2)$ time we can eliminate heavy teeth with root i according to Lemma 5.5.

6. Concluding remarks. We have given a polynomial-time separation algorithm for the simple DP inequalities, which include the simple comb inequalities as a special case. This is a significant extension of the results of Padberg and Rao [30] and forms the latest in a series of positive results concerned with comb separation (Padberg and Rao [30], Carr [5], Fleischer and Tardos [11], Caprara et al. [4], Letchford [19], and Caprara and Letchford [3]).

A number of open questions immediately spring to mind. The main one is, of course, whether there exists a polynomial-time separation algorithm for *general* comb inequalities, or perhaps a generalization of them such as the *domino parity* inequalities (Letchford [19]). For some further discussion of this issue see Letchford [20].

We can also consider special classes of graphs. For a given graph G, let us denote by $\mathrm{S}(G)$ the polytope defined by the degree equations, the SECs, and the nonnegativity and simple DP inequalities. (Now we only define variables for the edges in G.) Let us say that a graph G is *S-perfect* if $\mathrm{S}(G)$ is an integral polytope. Clearly, the TSP is polynomially solvable on S-perfect graphs. It would be desirable to know which graphs are S-perfect. Similarly, let us say that a graph is *S-Hamiltonian* if $\mathrm{S}(G)$ is nonempty. Obviously, every Hamiltonian graph is S-Hamiltonian but the reverse does not hold. (The famous *Peterson graph* is S-Hamiltonian but not Hamiltonian.) It would be desirable to establish structural properties for the S-Hamiltonian graphs just as Chvátal [6] did for the so-called *weakly Hamiltonian* graphs.

Finally, we would like to make an observation about *lower bounds*. The lower bound obtained by optimizing over $\mathrm{SEP}(n)$ is good in practice, and it is conjectured (e.g., Goemans [12]) that it is always at least $3/4$ of the optimal value when the costs satisfy the triangle inequality. We would expect the addition of the simple DP inequalities to lead to even stronger bounds in practice. However, consider the family of fractional extreme points of $\mathrm{SEP}(4k)$ with $k \geq 2$, shown in Figure 11. Points of this type violate many comb inequalities but no simple DP inequalities. The following *path inequality* is valid for $\mathrm{STSP}(4k)$ (see Cornuéjols et al. [8]):

$$\sum_{i=1}^{k-1} x(\delta(H_i)) + \sum_{i=1}^{3} x(\delta(T_i)) \geq 4k+2,$$

where $H_i := \{1, \ldots, 4i\}$, $T_1 := \{1, 5, \ldots, 4k-3\}$, $T_2 := \{2, 3, 6, 7, \ldots, 4k-2, 4k-1\}$, and $T_3 := \{4, 8, \ldots, 4k\}$. Moreover, the left-hand side coefficients of the path inequality are easily seen to satisfy the triangle inequality. Now, the left-hand side of this inequality, computed with respect to the fractional point, is only $3k+3$. Thus, even when simple DP inequalities are used the ratio between lower bound and optimum can be as bad as $(3k+3)/(4k+2)$, which approaches $3/4$ as k approaches infinity.

Appendix.

Proof of Theorem 5.4. Let $j \in V \backslash \{i\}$ be an arbitrary vertex, and let M be a 0-1 matrix whose columns correspond to the vertices in $V \backslash \{i, j\}$ and whose rows are the incidence vectors of the i-heavy sets which do not include j. Due to Lemma 3.4, the theorem is true if and only if the columns of M can be permuted so that in every row of the resulting matrix the 1s occur consecutively. Then, from Theorem 9 of Tucker [33] on matrices with the consecutive 1s property, it suffices to prove the following five claims:

Claim 1. There can not exist i-heavy sets $S_1, \ldots, S_m \subset V \backslash \{i, j\}$ with $m \geq 3$ and a set $R \subseteq V \backslash \{i, j\}$ consisting of distinct vertices $v_1, \ldots, v_m$ such that, for all k, $S_k \cap R = \{v_k, v_{k+1}\}$ (with subscripts taken modulo m).

Claim 2. There can not exist i-heavy sets $S_1, \ldots, S_m \subset V \backslash \{i, j\}$ with $m \geq 4$ and a set $R \subseteq V \backslash \{i, j\}$ consisting of distinct vertices $v_1, \ldots, v_m$ such that $S_k \cap R = \{v_k, v_{k+1}\}$ for $k = 1, \ldots, m-2$, $S_{m-1} \cap R = \{v_1, \ldots, v_{m-2}, v_m\}$, and $S_m \cap R = \{v_2, \ldots, v_m\}$.

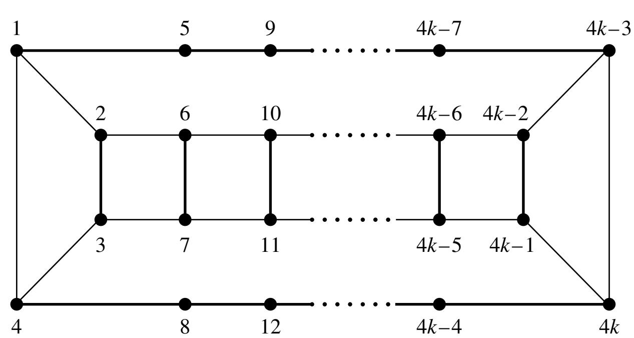

Figure 11. Fractional vertex of subtour polytope. Bold lines have $x_e^* = 1$, narrow lines have $x_e^* = 1/2$.

Claim 3. There can not exist i-heavy sets $S_1, \ldots, S_m \subset V \backslash \{i, j\}$ with $m \geq 3$ and a set $R \subseteq V \backslash \{i, j\}$ consisting of distinct vertices $v_1, \ldots, v_{m+1}$ such that $S_k \cap R = \{v_k, v_{k+1}\}$ for $k = 1, \ldots, m-1$, and $S_m \cap R = \{v_2, \ldots, v_{m-1}, v_{m+1}\}$.

Claim 4. There can not exist four i-heavy sets $S_1, \ldots, S_4 \subset V \backslash \{i, j\}$ and a set $R \subseteq V \backslash \{i, j\}$ consisting of distinct vertices $v_1, \ldots, v_6$ such that $S_1 \cap R = \{v_1, v_3, v_5\}$, $S_2 \cap R = \{v_1, v_2\}$, $S_3 \cap R = \{v_3, v_4\}$, and $S_4 \cap R = \{v_5, v_6\}$.

Claim 5. There can not exist four i-heavy sets $S_1, \ldots, S_4 \subset V \backslash \{i, j\}$ and a set $R \subseteq V \backslash \{i, j\}$ consisting of distinct vertices $v_1, \ldots, v_5$ such that $S_1 \cap R = \{v_1, v_3, v_5\}$, $S_2 \cap R = \{v_1, v_2\}$, $S_3 \cap R = \{v_3, v_4\}$, and $S_4 \cap R = \{v_1, \ldots, v_4\}$.

Proof of Claim 1. Here we assume that indices are taken modulo m. We sum together the following subtour elimination constraints:
- the SEC on $\{i\} \cup S_1 \cup \cdots \cup S_m$ (two times);
- the SECs on $S_k \cap S_{k+1}$ for $k = 1, \ldots, m$;
- the SECs on $(S_k \cap S_{k+1}) \backslash (S_{k+2} \cup \cdots \cup S_{k+m-1})$ for $k = 1, \ldots, m$;
- the SEC on $\{i\} \cup (S_1 \cap \cdots \cap S_m)$ ($m-2$ times).

Simple but tedious checking shows that the left-hand side of the resulting inequality is greater than or equal to $2\sum_{k=1}^m x(E(S_k)) + \sum_{k=1}^m x(E(i : S_k))$ and that the right-hand side is $2\sum_{k=1}^m |S_k| - 2m$. Hence, we have:

$$2\sum_{k=1}^{m} x(E(S_k)) + \sum_{k=1}^{m} x(E(i : S_k)) \leq \sum_{k=1}^{m} 2|S_k| - 2m. \tag{12}$$

However, the sum from $k = 1, \ldots, m$ of the tooth inequality with root i and body S_k is:

$$2\sum_{k=1}^{m} x(E(S_k)) + \sum_{k=1}^{m} x(E(i : S_k)) \leq \sum_{k=1}^{m} 2|S_k| - m. \tag{13}$$

Comparing Equations (12) and (13), we conclude that when $x^* \in \mathrm{SEP}(n)$, at least one of the m tooth inequalities has slack ≥ 1 at x^*. Hence, at least one of the sets $S_1, \ldots, S_m$ is not i-heavy.

Proof of Claim 2. Lemma 3.4 shows that S_m is i-heavy if and only if $V \backslash (\{i\} \cup S_m)$ is i-heavy, and a similar statement holds for S_{m-1}. If we replace S_m with $V \backslash (\{i\} \cup S_m)$ and replace S_{m-1} with $V \backslash (\{i\} \cup S_{m-1})$, we obtain the configuration described in Claim 1, which we have already proved can not exist.

Proof of Claim 3. If we replace S_m with $V \backslash (\{i\} \cup S_m)$, we again obtain the configuration described in Claim 1.

Proof of Claim 4. If S_2, S_3, and S_4 are i-heavy, then the sum of the slacks of the three associated tooth inequalities must be less than three. Equivalently,

$$\sum_{j=2}^{4} (2x(E(S_j)) + x(E(i : S_j))) > 2\sum_{j=2}^{4} |S_j| - 6.$$

Subtracting from this the degree equation on i and the degree equations for the vertices in $S_2 \cap S_3 \cap S_4$ gives

$$\begin{aligned} &2\left(\sum_{j=2}^{4} x(E(S_j)) - x(E(S_2 \cap S_3 \cap S_4))\right) + \sum_{j=2}^{4} x(E(i : S_j)) \\ &\qquad - x(\delta(i)) - x(\delta(S_2 \cap S_3 \cap S_4)) > 2\left(\sum_{j=2}^{4} |S_j| - |S_2 \cap S_3 \cap S_4|\right) - 8. \end{aligned}$$

Dividing by two and weakening gives

$$\sum_{j=2}^{4} x(E(S_j)) + x\big(E(i : (S_2 \cap S_3) \cup (S_2 \cap S_4) \cup (S_3 \cap S_4))\big) - x(E(S_2 \cap S_3 \cap S_4)) > \sum_{j=2}^{4} |S_j| - |S_2 \cap S_3 \cap S_4| - 4.$$

Subtracting the degree equations for the vertices in $(S_2 \cap S_3) \cup (S_2 \cap S_4) \cup (S_3 \cap S_4)$, adding the SECs for $(S_1 \cap S_2) \backslash (S_3 \cup S_4)$, $(S_1 \cap S_3) \backslash (S_2 \cup S_4)$, $(S_1 \cap S_4) \backslash (S_2 \cup S_3)$, $S_2 \backslash (S_1 \cup S_3 \cup S_4)$, $S_3 \backslash (S_1 \cup S_2 \cup S_4)$, and $S_4 \backslash (S_1 \cup S_2 \cup S_3)$, and weakening gives

$$x(E(S_1 : V \backslash (S_1 \cup \{i\}))) > 2.$$

This together with the degree equations for the vertices in S_1 yields

$$2x(E(S_1)) + x(E(i : S_1)) < 2|S_1| - 2,$$

thus showing that S_1 cannot be i-heavy.

Proof of Claim 5. Lemma 3.4 shows that S_4 is i-heavy if and only if $V\backslash(\{i\}\cup S_4)$ is i-heavy. If we replace S_4 with $V\backslash(\{i\}\cup S_4)$ and interchange the roles of j and v_5, we obtain the configuration described in Claim 4, which we have already proved can not exist. $\square$

Acknowledgments. The first author was supported in part by NSF grant CCR-0049071. The authors are grateful to an anonymous referee for a very careful reading and insightful comments.

References

[1] Applegate, D., R. E. Bixby, V. Chvátal, W. Cook. 1995. Finding cuts in the TSP (a preliminary report). Technical report 95–05, Centre for Discrete Mathematics and Theoretical Computer Science (DIMACS), Rutgers University, New Brunswick, NJ.
[2] Caprara, A., M. Fischetti. 1996. $\{0, \frac{1}{2}\}$-Chvátal-Gomory cuts. *Math. Programming* **74** 221–235.
[3] Caprara, A., A. N. Letchford. 2003. On the separation of split cuts and related inequalities. *Math. Programming* **94** 279–294.
[4] Caprara, A., M. Fischetti, A. N. Letchford. 2000. On the separation of maximally violated mod-k cuts. *Math. Programming* **87** 37–56.
[5] Carr, R. 1997. Separating clique trees and bipartition inequalities having a fixed number of handles and teeth in polynomial time. *Math. Oper. Res.* **22** 257–265.
[6] Chvátal, V. 1973. Edmonds polytopes and weakly Hamiltonian graphs. *Math. Programming* **5** 29–40.
[7] Cook, W., D. Espinoza, M. Goycoolea. 2005. A study of domino-parity and k-parity constraints for the TSP. M. Jünger, V. Kaibel, eds. *Integer Programming and Combinatorial Optimization XI, Lecture Notes in Computer Science*, Vol. 3509. Springer, Berlin, Germany, pp. 452–467.
[8] Cornuéjols, G., J. Fonlupt, D. Naddef. 1985. The traveling salesman problem on a graph and some related integer polyhedra. *Math. Programming* **33** 1–27.
[9] Dantzig, G. B., D. R. Fulkerson, S. M. Johnson. 1954. Solution of a large-scale traveling salesman problem. *Oper. Res.* **2** 393–410.
[10] Edmonds, J. 1965. Maximum matching and a polyhedron with 0–1 vertices. *J. Res. Nat. Bur. Standards* **69B** 125–130.
[11] Fleischer, L., É. Tardos. 1999. Separating maximally violated comb inequalities in planar graphs. *Math. Oper. Res.* **24** 130–148.
[12] Goemans, M. X. 1995. Worst-case comparison of valid inequalities for the TSP. *Math. Programming* **69** 335–349.
[13] Goldberg, A. V., R. E. Tarjan. 1988. A new approach to the maximum flow problem. *J. ACM* **35** 921–940.
[14] Grötschel, M., M. W. Padberg. 1979. On the symmetric travelling salesman problem I: Inequalities. *Math. Programming* **16** 265–280.
[15] Grötschel, M., M. W. Padberg. 1979. On the symmetric travelling salesman problem II: Lifting theorems and facets. *Math. Programming* **16** 281–302.
[16] Grötschel, M., L. Lovász, A. J. Schrijver. 1988. *Geometric Algorithms and Combinatorial Optimization*. Wiley, New York.
[17] Henzinger, M. R., D. P. Williamson. 1996. On the number of small cuts in a graph. *Inform. Proc. Lett.* **59** 41–44.
[18] Jünger, M., G. Reinelt, G. Rinaldi. 1995. The traveling salesman problem. M. Ball, T. Magnanti, C. Monma, G. Nemhauser, eds. *Network Models, Handbooks in Operations Research and Management Science*, Vol. 7. Elsevier Publisher B.V., Amsterdam, The Netherlands, 225–330.
[19] Letchford, A. N. 2000. Separating a superclass of comb inequalities in planar graphs. *Math. Oper. Res.* **25** 443–454.
[20] Letchford, A. N. 2003. Binary clutter inequalities for integer programs. *Math. Programming* **98** 201–221.
[21] Letchford, A. N., A. Lodi. 2002. Polynomial-time separation of simple comb inequalities (extended abstract). W. J. Cook, A. S. Schulz, eds. *Integer Programming and Combinatorial Optimization 9, Lecture Notes in Computer Science*, Vol. 2337. Springer-Verlag, Berlin/Heidelberg, Germany, 96–111.
[22] Letchford, A. N., G. Reinelt, D. O. Theis. 2004. A faster exact separation algorithm for blossom inequalities. G. Nemhauser, D. Bienstock, eds. *Integer Programming and Combinatorial Optimization 10, Lecture Notes in Computer Science*, Vol. 3064. Springer-Verlag, Berlin/Heidelberg, Germany, 196–205.
[23] Naddef, D. 2002. Polyhedral theory and branch-and-cut algorithms for the symmetric TSP. G. Gutin, A. Punnen, eds. *The Traveling Salesman Problem and its Variations*. Kluwer Academic Publishers, Boston, MA, 29–116.
[24] Naddef, D., G. Rinaldi. 1988. The symmetric traveling salesman polytope: New facets from the graphical relaxation. Technical report 248, Istituto di Analisi dei Sistemi e Informatica (IASI-CNR), Rome, Italy.
[25] Naddef, D., G. Rinaldi. 1993. The graphical relaxation: A new framework for the symmetric travelling salesman polytope. *Math. Programming* **58** 53–88.
[26] Naddef, D., S. Thienel. 2002. Efficient separation routines for the symmetric traveling salesman problem I: General tools and comb separation. *Math. Programming* **92** 237–255.
[27] Naddef, D., E. Wild. 2003. The domino inequalities: new facets for the symmetric traveling salesman polytope. *Math. Programming* **98** 223–251.
[28] Nagamochi, H., K. Nishimura, T. Ibaraki. 1997. Computing all small cuts in undirected networks. *SIAM Discrete Math.* **10** 469–481.
[29] Padberg, M. W., M. Grötschel. 1985. Polyhedral computations. E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, D. B. Schmoys, eds. *The Traveling Salesman Problem*. John Wiley & Sons, Chichester, UK, 307–360.
[30] Padberg, M. W., M. R. Rao. 1982. Odd minimum cut-sets and b-matchings. *Math. Oper. Res.* **7** 67–80.
[31] Padberg, M. W., G. Rinaldi. 1990. Facet identification for the symmetric traveling salesman polytope. *Math. Programming* **47** 219–257.
[32] Padberg, M. W., G. Rinaldi. 1991. A branch-and-cut algorithm for the resolution of large-scale symmetric travelling salesman problems. *SIAM Rev.* **33** 60–100.
[33] Tucker, A. 1972. A structure theorem for the consecutive 1s property. *J. Combin. Theory Ser. B* **12** 153–162.