4OR 1: 209–224 (2003)

DOI: 10.1007/s10288-003-0012-8

Primal separation algorithms

Adam N. Letchford[1] **and Andrea Lodi**[2]

[1] Department of Management Science, Lancaster University, Lancaster LA1 4YW, England (e-mail: A.N.Letchford@lancaster.ac.uk)
[2] D.E.I.S., University of Bologna, Viale Risorgimento 2, 40136 Bologna, Italy (e-mail: alodi@deis.unibo.it)

Received: November 2002 / Revised version: March 2003

Abstract. Given an integer polyhedron $P_I \subset \mathbb{R}^n$, an integer point $\bar{x} \in P_I$, and a point $x^* \in \mathbb{R}^n \setminus P_I$, the *primal separation problem* is the problem of finding a linear inequality which is valid for P_I, violated by x^*, and satisfied at equality by $\bar{x}$. The primal separation problem plays a key role in the primal approach to integer programming.

In this paper we examine the complexity of primal separation for several well-known classes of inequalities for various important combinatorial optimization problems, including the *knapsack*, *stable set* and *travelling salesman* problems.

Key words: Integer programming, separation, primal algorithms, knapsack problem, stable set problem, travelling salesman problem.

1 Introduction

In the early 1980s, several authors independently realised that Khachian's ellipsoid algorithm for linear programming had important implications for combinatorial optimization and integer programming (see Grötschel et al. 1988). The basic idea is as follows. Suppose we have an integer polyhedron of the form:

$$P_I = \text{conv}\{x \in Z^n_+ : Ax \leq b\}, \tag{1}$$

where $A \in Z^{m \times n}$ and $b \in Z^m$. Then, under certain reasonable assumptions, one can optimize a linear function over P_I efficiently if and only if the following *separation problem* can be solved efficiently:

The separation problem: *Given some $x^* \in \mathbb{R}^n_+$, find an inequality which is valid for P_I and violated by x^*, or prove that none exists.*

4OR Quarterly Journal of the Belgian, French and Italian Operations Research Societies

This idea enabled Grötschel et al. (1988) to obtain the first known polynomial-time algorithms for *submodular function minimization* and for finding a maximum weight *stable set* in perfect graphs.

Of equal importance is the fact that, using a slightly modified version of the separation problem, it is possible to obtain (exponential, but still useful) solution procedures for $\mathcal{NP}$-hard optimization problems too. For many $\mathcal{NP}$-hard problems (such as the *stable set*, *knapsack* and *travelling salesman* problems), various classes of valid inequalities are known which give a *partial* description of the associated P_I. One can define a separation problem for each class of inequalities:

> **The separation problem for a class $\mathcal{F}$ of inequalities:** *Given a vector $x^* \in \mathbb{R}^n_+$, find a member of $\mathcal{F}$ which is violated by x^*, or prove that none exists.*

Frequently, this modified separation problem is polynomial for some classes of inequalities and $\mathcal{NP}$-hard for others. Exact and heuristic separation routines for specific classes of inequalities are at the core of modern *cutting plane* algorithms, and have led to the highly successful *branch-and-cut* approach to integer and combinatorial optimization (see Nemhauser and Wolsey 1988; Padberg and Rinaldi 1991; Caprara and Fischetti 1997).

In a recent paper (Letchford and Lodi 2002a) we pointed out that all of the cutting plane algorithms in the modern literature are so-called *dual fractional* algorithms, in which the main purpose of the cutting planes is to improve an upper bound (for maximization problems) or a lower bound (for minimization problems). We argued that more attention should be given to *primal* cutting plane algorithms, in which the main purpose of the cutting planes is to improve a feasible integer solution, i.e., a *lower* bound (for maximization problems) or an *upper* bound (for minimization problems).

As explained in that paper, the separation problem has to be modified in the primal context. Suppose that the best known feasible solution to the problem is denoted by $\bar{x}$, and that x^* is a fractional point with better objective value. When trying to separate x^* from P_I, it is desirable to add the extra condition that the violated inequality be *tight* (satisfied at equality) at $\bar{x}$. Otherwise, in the next iteration our algorithm would merely generate a new fractional point which is a convex combination of x^* and $\bar{x}$. (A detailed discussion of primal vs. standard separation is given in Letchford and Lodi 2002a). This consideration led to the following two definitions:

> **The primal separation problem:** *Given some $x^* \in \mathbb{R}^n_+$, and some integer vector $\bar{x} \in P_I$, find an inequality which is valid for P_I, violated by x^*, and tight for $\bar{x}$, or prove that none exists.*

> **The primal separation problem for a class $\mathcal{F}$ of inequalities:** *Given a vector $x^* \in \mathbb{R}^n_+$, and some integer vector $\bar{x} \in P_I$, find a member of $\mathcal{F}$ which is violated by x^*, and tight for $\bar{x}$, or prove that none exists.*

In the same paper (see also Padberg and Grötschel 1985) it was noted that the primal separation problem (either for a class of polyhedra or a class of inequalities) can be *transformed* into the standard separation problem in a simple way. The idea is that an inequality solves the primal separation problem if and only if the same inequality solves the standard separation problem for the point $\tilde{x} := \epsilon x^* + (1-\epsilon)\bar{x}$, where ϵ is a small positive quantity whose encoding length is polynomial in the encoding length of the inequalities defining P_I.

Therefore, in terms of computational complexity, the primal version of a separation problem is *no harder* than the standard version. But, interestingly, the reverse does not hold in general. As we show in the present paper, for some inequalities primal separation is *easier* than standard separation, either in terms of implementation or in terms of asymptotic running time.

The remainder of the paper is structured as follows. In Sect. 2 we consider the complexity of primal separation for P_I itself, for various integer programming problems. In Sect. 3 we consider the primal separation problem for various classes of valid inequalities and facets. Conclusions are given in Sect. 4.

Throughout the paper we assume that the reader is familiar with basic concepts of complexity theory, such as the classes $\mathcal{NP}$ and co-$\mathcal{NP}$, $\mathcal{NP}$-completeness and $\mathcal{NP}$-hardness (see Garey and Johnson 1979). We also distinguish between *Turing-reducibility* and *Karp-reducibility*. Problem A is Turing-reducible to problem B if there exists an algorithm for A which works by solving a polynomial number of instances of problem B. Karp-reducibility, on the other hand, is a stronger notion: we require that the algorithm for A calls the algorithm for problem B only *once*. A Karp reduction is commonly called a *transformation*.

To finish this introduction, we mention the recent paper by Eisenbrand et al. (2003). They show that, in the special case of 0-1 integer programs, the optimization problem and the primal separation problem are Turing-reducible to each other. Thus, if a class of 0-1 integer programming problems is $\mathcal{NP}$-hard, primal separation is too. They also explore separation for various classes of inequalities, such as the *odd cut* inequalities for b-matching problems, and show that primal separation for these inequalities can be performed more quickly than standard separation. This is in accordance with our results.

2 Primal separation for P_I

In this section we will show that, for various families of integer polyhedra, the primal separation problem can be easily *transformed* into the standard separation problem. The approach is much more direct than that given in Eisenbrand et al. (2003), which is based on a sequence of Turing reductions. Moreover, our results will not be limited to the 0-1 case.

We begin by defining various families of integer polyhedra by imposing conditions on A and b in (1). In particular we define:

- The *set packing* polyhedra, obtained when $A \in \{0, 1\}^{m \times n}$ and $b = \{1\}^m$;

- The *set covering* polyhedra, obtained when $A \in \{0, -1\}^{m \times n}$ and $b = \{-1\}^m$;
- The *set partitioning* polyhedra, obtained when (A, b) can be written as $\binom{A' \quad b'}{-A' \ -b'}$, where $A' \in \{0, 1\}^{(m/2) \times n}$ and $b' = \{1\}^{m/2}$;
- The (general integer) *knapsack* polyhedra, obtained when $m = 1$, $b > 0$, and A has non-negative entries.

We have the following theorem:

Theorem 1. *Suppose that a family of integer polyhedra has the following property: If the polyhedron*

$$P_I = conv\{x \in Z_+^n : \sum_{j=1}^{n} a_{ij} x_j \le b_i \qquad (i = 1, \ldots, m)\} \tag{2}$$

lies in the family, then so does the polyhedron

$$P_I' = conv\{x \in Z_+^{n+1} : \sum_{j=1}^{n} a_{ij} x_j + b_i x_{n+1} \le b_i \qquad (i = 1, \ldots, m)\}. \tag{3}$$

Then the standard separation problem for that family can be transformed (Karp-reduced) to the primal separation problem for the same family.

Proof. Let P_I be defined as in (2) and let the vector x^* be the input for the standard separation problem. Also let P_I' be as in (3). By assumption, P_I' lies in the given family. Now let $\bar{x} \in P_I'$ be defined as follows. Set $\bar{x}_j = 0$ for $j = 1, \ldots, n$, and set $\bar{x}_{n+1} = 1$. Then the inequality $\sum_{j=1}^{n} \alpha_j x_j \le \beta$ is valid for P_I and violated by x^* if and only if the inequality $\sum_{j=1}^{n} \alpha_j x_j + \beta x_{n+1} \le \beta$ is valid for P_I', violated by $(x^*, 0)$, and tight for $\bar{x}$. □

This immediately implies the following corollary:

Corollary 1. *The standard and primal separation problems are equivalent (in the Karp sense) for set packing, set covering, set partitioning and knapsack polyhedra, and also for general integer programs.*

Proof. These families of polyhedra meet the condition of Theorem 1. □

Thus, we obtain a simple proof that these separation problems are all $\mathcal{NP}$-hard under Turing reductions. Moreover, unlike the result of Eisenbrand et al. (2003), Theorem 1 is not limited to 0-1 integer programs. However, an analogous result does indeed hold for problems in which variables are explicitly constrained to be binary:

Theorem 2. *Suppose that a family of 0-1 integer polyhedra has the following property: If the polyhedron*

$$P_I = conv\{x \in \{0,1\}^n : \sum_{j=1}^{n} a_{ij}x_j \le b_i \qquad (i = 1, \dots, m)\}$$

lies in the family, then so does the polyhedron

$$P_I' = conv\{x \in \{0,1\}^{n+1} : \sum_{j=1}^{n} a_{ij}x_j + b_i x_{n+1} \le b_i \qquad (i = 1, \dots, m)\}.$$

Then the standard separation problem for that family can be transformed to the primal separation problem for the same family.

Proof. Similar to the proof of Theorem 1. □

This enables us to prove the following two results about primal separation for *0-1 knapsack polytopes*, i.e., polytopes of the form:

$$P_I := conv\left\{x \in \{0,1\}^n : \sum_{i=1}^{n} a_i x_i \le a_0\right\}. \tag{4}$$

Corollary 2. *The standard and primal separation problems are equivalent (in the Karp sense) for 0-1 knapsack polytopes.*

Proof. 0-1 knapsack polytopes meet the condition of Theorem 2. □

Corollary 3. *The following problem is co-$\mathcal{NP}$-complete in the Karp sense: given a 0-1 knapsack polytope of the form (4), a point $x^* \in [0,1]^n$, and an integer vector $\bar{x} \in P_I$, decide whether x^* violates a valid inequality which is tight for $\bar{x}$.*

Proof. Schrijver (1986), pp. 253–254 showed that the problem of deciding whether a given $x^* \in [0,1]^n$ lies in a given 0-1 knapsack polytope is $\mathcal{NP}$-complete in the Karp sense. The result then follows from Corollary 2. □

We would like to point out that there are several important families of polyhedra which are not covered by Theorems 1 and 2. For example, there is no obvious way to transform the standard separation problem for *travelling salesman polytopes* (see Padberg and Grötschel 1985, and also Subsection 3.5) to the primal version. Thus, although testing if a given x^* lies in a given travelling salesman polytope is strongly $\mathcal{NP}$-complete in the Karp sense (Papadimitriou and Yannakakis 1984), primal separation for travelling salesman polytopes may only be $\mathcal{NP}$-hard under *Turing* reductions.

To present the last two results of this section, we will need the definition of the following strongly $\mathcal{NP}$-complete decision problem, taken from Garey and Johnson (1979):

3-PARTITION: *Given positive integers* $q_1, \ldots, q_r$ *such that* $Q := (\sum_{i=1}^{r} q_i)/3$ *is integer, is there a partition of* $\{1, \ldots, r\}$ *into three sets* S_1, S_2, S_3 *such that* $\sum_{i \in S_1} q_i = \sum_{i \in S_2} q_i = \sum_{i \in S_3} q_i = Q$ *?*

Theorem 3. *The following problem is strongly* $\mathcal{NP}$*-complete in the Karp sense: given a polytope* P_I *defined as in (1) and a point* $x^* \in \mathbb{R}^n_+$*, decide whether* $x^* \in P_I$.

Proof. Our proof follows Schrijver (1986), pp. 253–254. Let $q_1, \ldots, q_r$ and Q be an instance of 3-PARTITION. We define the integer polyhedron:

$$P_I := \text{conv}\{x \in Z^{3r}_+ : \ \sum_{j=1}^{r} q_j x_{ij} \le Q \quad (i = 1, 2, 3), \\ x_{1j} + x_{2j} + x_{3j} = 1 \ (j = 1, \ldots, r)\}.$$

Now consider the fractional point $x^*_{ij} = 1/3$ for $i = 1, 2, 3$ and $j = 1, \ldots, r$. The answer to 3-PARTITION is 'yes' if and only if $x^* \in P_I$. (The answer is 'no' if and only if $P_I = \emptyset$.) All that remains is to show that the problem is in $\mathcal{NP}$. To see this, note that a given x^* lies in a given P_I if and only if it can be written as a convex combination of extreme points of P_I. By Caratheodory's theorem, a polynomial number of extreme points suffices. □

Corollary 4. *The following problem is strongly co-*$\mathcal{NP}$*-complete in the Karp sense: given a polytope* P_I *defined as in (1), a point* $x^* \in \mathbb{R}^n_+$*, and an integer point* $\bar{x} \in P_I$*, decide whether* x^* *violates a valid inequality which is tight at* $\bar{x}$.

Proof. Follows from Theorem 3 and the proof of Theorem 1. □

3 Primal separation for specific inequalities

3.1 Chvátal-Gomory cuts

In this section we turn our attention to the complexity of primal separation for various specific classes of inequalities. We begin by considering the well-known *Chvátal-Gomory* (CG) *cuts*, which are defined for any polyhedron of the form given in (1). A CG cut is (see for example Nemhauser and Wolsey 1988), a valid inequality of the form $\lfloor \lambda A \rfloor x \le \lfloor \lambda b \rfloor$, where $\lambda \in \mathbb{R}^m_+$ is an arbitrary multiplier vector chosen so that $\lambda b \notin Z$.

Recently, Eisenbrand (1999) showed that standard separation of CG cuts is strongly $\mathcal{NP}$-hard. In fact his proof uses a direct transformation, so separation is hard in the Karp sense. Unfortunately, the same holds for the primal version.

Theorem 4. *For CG cuts, the standard separation problem can be transformed into the primal separation problem.*

Proof. We use the same transformation as in the proof of Theorem 1. Let P_I be defined as in (2) and let the vector x^* be the input for the standard CG separation problem. We define the polyhedron P_I' and the vector $\bar{x} \in Z_+^{n+1}$ as in the proof of Theorem 1. Note that any CG cut for P_I' is tight at $\bar{x}$. Then, a CG cut for P_I is violated by x^* if and only if a CG cut for P_I' is violated by $(x^*, 0)$ and tight at $\bar{x}$. □

Corollary 5. *Primal CG separation is strongly $\mathcal{NP}$-hard in the Karp sense.*

3.2 Clique inequalities

The well-known *clique* inequalities are facet-inducing inequalities for the *stable set problem*. Let G be an undirected graph with vertex set V and edge set E. A set $S \subset V$ is said to be *stable* if no two vertices in S are adjacent in G. Given such a graph, along with a vector $w \in \mathbb{R}_+^{|V|}$ of vertex weights, the well-known (and strongly $\mathcal{NP}$-hard) *maximum weight stable set problem* is that of finding a stable set of maximum weight (see Grötschel et al. 1988, for a survey). The standard 0-1 integer programming formulation of this problem is (see Padberg 1973):

$$\max \left\{ w^T x : x \in \{0,1\}^{|V|} : x_i + x_j \leq 1 \ \forall \{i,j\} \in E \right\}.$$

Note that the associated P_I is a set packing polytope. This can be used to show that the primal separation for P_I is equivalent (in the Karp sense) to the standard separation problem. Here, however, we are concerned with a specific class of valid inequalities, the so-called *clique inequalities* of Padberg (1973). The clique inequalities, which are facet-inducing, are of the form $\sum_{i \in C} x_i \leq 1$, where $C \subseteq V$ is a maximal clique in G. The associated (standard) separation problem is to find a clique in G whose x^*-weight exceeds 1, or prove that none exists. Nemhauser and Wolsey (1988), pp. 163–164 show that this problem is strongly $\mathcal{NP}$-hard under Turing reductions. In fact, the following stronger result holds:

Theorem 5. *The problem 'Does x^* violate a clique inequality?' is strongly $\mathcal{NP}$-complete in the Karp sense.*

Proof. For any integer $k > 1$, consider the fractional point defined by $x_i^* = 1/k$ for all $i \in V$. Then a clique inequality is violated if and only if G contains a clique of cardinality greater than k. The latter problem is strongly $\mathcal{NP}$-complete in the Karp sense. □

Now, if $\bar{x}$ represents a stable set, then the *primal* separation problem amounts to finding a clique in G whose x^*-weight exceeds 1, under the additional restriction that this clique contains exactly one vertex i such that $\bar{x}_i = 1$. We now show that, in terms of computational complexity, this extra restriction does not help:

Theorem 6. *For clique inequalities, the standard separation problem can be transformed into the primal separation problem.*

Proof. As in the proof of Theorem 1, this can be done by adding a dummy variable. So let G and x^* be the input to the standard separation problem. We construct a new graph G' by appending a new node u which is adjacent to every vertex in V. The polytopes P_I and P_I' are defined accordingly. We then define $\bar{x} \in P_I'$ by setting $\bar{x}_i = 0$ for $i \in V$, and setting $\bar{x}_u = 1$. Then x^* violates a clique inequality for P_I if and only if the point $(x^*, 0)$ violates a clique inequality which is valid for P_I' and tight for $\bar{x}$. This is so because any maximal clique in G can be enlarged to a maximal clique in G' by appending the dummy vertex u. □

Corollary 6. *Primal clique separation is strongly $\mathcal{NP}$-hard in the Karp sense.*

It is worth noting however that the point $\bar{x}$ in the proof of Theorem 6 is 'pathological': there is only one vertex with $\bar{x}_i = 1$ and this vertex has high degree. In practice, a stable set with high weight tends to contain *many* vertices (i.e., there will be many indices $i \in V$ such that $\bar{x}_i = 1$); moreover these vertices tend to have *small* degree. In such a situation, we can decompose primal clique separation into smaller independent sub-problems: for each vertex i with $\bar{x}_i = 1$, look for a maximum weight clique on the subgraph of G induced by the vertices which are adjacent to i.

Indeed, consider a random graph where every edge is present with probability p. It is known (Bollobás and Erdos 1976) that in such a graph, the maximum stable set is almost surely of size $\mathcal{O}(\log |V|)$. Hence, if $\bar{x}$ represents a nearly-optimal stable set, primal clique separation is likely to reduce to $\mathcal{O}(\log |V|)$ smaller maximum clique problems on graphs with fewer than $p|V|$ nodes.

So in practice we would expect primal separation to be easier than standard separation.

3.3 Odd cycle inequalities

Another well-known class of valid inequalities for the stable set polytope are the *odd cycle* inequalities, also due to Padberg (1973). These take the form $\sum_{i \in C} x_i \leq (|C| - 1)/2$, where C is a set of vertices inducing an odd cycle in G. The standard separation problem for odd cycle inequalities can be solved in polynomial time, see Gerards and Schrijver (1986), by solving $|V|$ shortest path problems in an auxiliary graph.

It turns out that the corresponding primal separation problem is in some sense easier. For an odd cycle inequality to be tight for $\bar{x}$, exactly $(|C| - 1)/2 > 0$ vertices in the cycle must have value 1 at $\bar{x}$. Thus, we know that the odd cycle must include at least one vertex currently packed. Using this fact, we can reduce the number of shortest path problems from $|V|$ to $\sum_{i \in V} \bar{x}_i$. If the graph is dense, and the current best stable set small, this will represent a considerable saving. Indeed, in the random graph model mentioned in Subsection 3.2, only $\mathcal{O}(\log |V|)$ shortest path computations will be needed.

This reduction in itself may not appear too impressive. However, it has important implications. Many valid and even facet-inducing inequalities for other combinatorial problems can also be separated by computing a minimum weight odd cycle in a suitably-defined graph (see, e.g., Caprara and Fischetti 1996 or Borndörfer and Weismantel 2000). Using similar arguments to those given here, it can be shown that primal separation can be solved more quickly than standard separation for many of these inequalities.

Here is a specific example. An important class of facet-inducing inequalities for the Asymmetric Travelling Salesman Problem (ATSP) are the *odd CAT* inequalities of Balas (1989). As shown in Caprara and Fischetti (1996) and Fischetti and Toth (1997), a slightly weakened version of these inequalities can be separated in polynomial time by computing a minimum weight odd cycle in a so-called *conflict graph*. A careful reading of these papers and some elementary analysis shows that the conflict graph has $O(m)$ vertices and $O(nm)$ edges, where n is the number of vertices and m is the number of variables which are positive at x^*. Computing the minimum weight odd cycle in the standard way involves solving a sequence of m shortest path problems, which, if we use a good implementation of Dijkstra's algorithm, leads to an $O(nm^2)$ time algorithm for the separation problem, which is $O(n^5)$ in the worst case.

Now, from the above argument, to solve the *primal* weak odd CAT separation problem it suffices to solve only n shortest path problems. This is because there are only n variables of value 1 in any ATSP tour. Thus, the primal weak odd CAT separation problem can be solved in $O(n^2m)$ time, which is only $O(n^4)$ in the worst case.

3.4 Cover inequalities

The *cover* inequalities are well-known valid inequalities for the 0-1 knapsack polytope, see e.g., Balas (1975) or Nemhauser and Wolsey (1988). A *cover* is a set $C \subseteq \{1, \dots n\}$ such that $\sum_{i \in C} a_i > a_0$. Each cover C yields a cover inequality of the form $\sum_{i \in C} x_i \leq |C| - 1$.

As shown in Crowder et al. (1983), the standard separation problem for cover inequalities can itself be formulated as a kind of knapsack problem, as follows. Define for $i = 1, \dots, n$ the zero-one variable y_i, which takes the value 1 if item i is to be placed into the cover, and 0 otherwise. Then the cover separation problem can be formulated as:

$$\min \left\{ \sum_{i=1}^{n} (1 - x_i^*) y_i \, : \, y \in \{0, 1\}^n, \ \sum_{i=1}^{n} a_i y_i > a_0 \right\}. \tag{5}$$

It was proved by Ferreira (1994) that this problem is $\mathcal{NP}$-hard in the Karp sense, even when x^* is non-negative and satisfies $\sum_{i=1}^{n} a_i x_i^* \leq a_0$.

It is instructive to note that the primal separation problem for cover inequalities can be formulated as a kind of 'multiple-choice' knapsack problem. If we define the index set $N_0 := \{i \in \{1, \ldots, n\} : \bar{x}_i = 0\}$, then the primal cover separation problem can be formulated as:

$$\min \left\{ \sum_{i=1}^{n} (1 - x_i^*) y_i : y \in \{0, 1\}^n, \sum_{i=1}^{n} a_i y_i > a_0, \sum_{i \in N_0} y_i = 1 \right\}. \quad (6)$$

We now show that, in terms of computational complexity, this extra restriction does not help:

Theorem 7. *For cover inequalities, the standard separation problem can be transformed into the primal separation problem.*

Proof. This follows similar lines to the proofs of Theorems 1 and 6, but the polytope P_I' has to be a little different and the construction of $\bar{x}$ has to be modified. We change the definition of P_I' to:

$$P_I' := \text{conv}\{x \in \{0, 1\}^{n+1} : \sum_{i=1}^{n} a_i x_i + M x_{n+1} \leq a_0 + M\}, \quad (7)$$

where M is a large integer satisfying $M \geq \sum_{i=1}^{n} a_i$; and we set $\bar{x}_i = 1$ for $i = 1, \ldots, n$ and $\bar{x}_{i+1} = 0$. Now, $C \subseteq \{1, \ldots, n\}$ is a cover for P_I if and only if $C \cup \{n + 1\}$ is a cover for P_I'. Also, x^* violates a cover inequality for P_I if and only if the point $(x^*, 1)$ violates a cover inequality which is both valid for P_I' and tight for $\bar{x}$. □

Corollary 7. *Primal cover separation is $\mathcal{NP}$-hard in the Karp sense.*

As for the clique inequalities (Subsection 3.2), we would like to point out that the vector $\bar{x}$ used in the proof of Theorem 7 is 'pathological': all variables but one have $\bar{x}_i = 1$, so that the index set N_0 (defined above) only has one member. In practice, N_0 is likely to be much larger. This is important because the primal cover separation problem (6) becomes easier when N_0 is large (because only one of the variables y_i with $i \in N_0$ can be set to one).

Of course, in practice, one uses *lifted* cover inequalities rather than ordinary cover inequalities as cutting planes. (See, e.g., Crowder et al. 1983, Nemhauser and Wolsey 1988, Gu et al. 1998). It can be shown without much difficulty that primal separation of lifted cover inequalities is also $\mathcal{NP}$-hard. Yet, some simple primal separation heuristics work well in practice, see Letchford and Lodi (2003).

3.5 Subtour elimination constraints

We now turn our attention to some valid inequalities for the well-known *Symmetric Travelling Salesman Problem* (STSP). The standard 0-1 programming formulation for the STSP is as follows (see, e.g., Padberg and Grötschel 1985):

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

Subject to:

$$x(\delta(i)) = 2 \quad \forall i \in V, \tag{8}$$

$$x(\delta(S)) \geq 2 \quad \forall S \subset V : |S| \geq 2, \tag{9}$$

$$x \in \{0, 1\}^{|E|}, \tag{10}$$

where V is the vertex set, E is the edge set, c is the cost vector and $\delta(S)$ denotes the set of edges with exactly one end-vertex in S.

The associated P_I is called the *Symmetric Travelling Salesman Polytope*. The equations (8), known as *degree equations*, describe the affine hull of this polytope. The inequalities (9), known as *subtour elimination constraints* (SECs), are facet-defining. It is well-known that, using the degree equations, the SECs can be written in the alternative form

$$x(E(S)) \leq |S| - 1 \quad \forall S \subset V : |S| \geq 2, \tag{11}$$

where $E(S)$ denotes the set of edges with both end-vertices in S. This will be useful in Subsection 3.7.

It is well-known (see, e.g., Padberg and Grötschel 1985) that the (standard) separation problem for the SECs (9) can be solved in polynomial time. Given some $x^* \in [0, 1]^{|E|}$ to be separated, we define $E^* := \{e \in E : x^*_e > 0\}$ and the *support graph* $G^* = (V, E^*)$. The SEC separation problem calls for a set $S \subset V$ which minimizes $x^*(\delta(S))$, which amounts to finding a minimum weight cut in G^*. The current best (deterministic) minimum cut algorithm is that of Nagamochi et al. (1994), which runs in $\mathcal{O}(|V||E^*| + |V|^2 \log |V|)$ time.

Now let $\bar{x}$ be the incidence vector of a tour. The primal SEC separation problem calls for a set $S \subset V$ which:

– a) minimizes $x^*(\delta(S))$,
– b) satisfies $\bar{x}(\delta(S)) = 2$.

The interesting thing is that, although there are an exponential number of SECs, the number of them which satisfy condition (b) is polynomial. (To be precise, the number of distinct SECs satisfying the condition is $|V|(|V| - 3)/2$.) This suggests that primal separation should be easier than the standard version. This is indeed true: Padberg and Hong (1980) devised an $\mathcal{O}(|V||E^*|)$ time algorithm for the primal version and, with a little more work, an $\mathcal{O}(|E^*| \log |V|)$ algorithm can be devised.

The $O(|E^*| \log |V|)$ algorithm is based on a 'divide-and-conquer' argument. Assume for simplicity that the vertices in the tour $\bar{x}$ are numbered in the order

$1, \ldots, |V|$. For $i = 1, \ldots, |V| - 1$ and $j = i + 1, \ldots, |V|$, let $S(i, j) := \{i, i + 1, \ldots, j\}$. For a given i, suppose $S(i, q_i)$ is the vertex set whose SEC has smallest slack among all sets $S(i, j)$. Using a simple 'uncrossing' type argument, it can be shown that the sets $S(i, q_i)$ form a 'nested' (or laminar) family. Thus, we can find $S(1, q_1)$, which takes $O(|E^*|)$ time, and then divide the problem into two subproblems: one in which vertices $\{1, \ldots, q_1\}$ are shrunk into a single vertex, and the other in which the vertices $\{q_1 + 1, \ldots, |V|\}$ are shrunk into a single vertex. Repeating this idea, and choosing the order of computations appropriately, one can ensure that no edge is examined more than $O(\log |V|)$ times.

We have since found out that this idea was discovered independently by Applegate et al. (1998) a number of years ago. (Indeed, there is a brief reference to it on page 650 of Applegate et al. 1998, and the idea is used in their 'Concorde' code, in the file 'segments.c'.) The interesting thing is that (as we understand it) Applegate et al. (1998) did not have primal separation in mind. Rather, they ran this algorithm, using the best tour $\bar{x}$, as a fast heuristic for *standard* separation.

3.6 2-matching inequalities

Another well-known class of facet-inducing inequalities for the STSP are the so-called *2-matching* inequalities (again, see Padberg and Grötschel 1985). Using the degree equations (8), these can be written in various forms, one of which is:

$$x(\delta(H) \setminus F) + \sum_{e \in F} (1 - x_e) \geq 1. \tag{12}$$

where H, the so-called *handle*, satisfies $\emptyset \neq H \subset V$, and F is a set of edges (the so-called *teeth*) satisfying:

- $F \subset \delta(H)$,
- $|F|$ odd.

Padberg and Rao (1982) showed that the standard 2-matching separation problem can be solved in polynomial time. The Padberg-Rao algorithm involves computing a minimum weight odd cut in a certain graph. (This graph has $\mathcal{O}(|E^*|)$ vertices and $\mathcal{O}(|E^*|)$ edges, and its nodes are labelled odd or even.) They then showed that such an odd cut can be found in polynomial time by solving a series of $\mathcal{O}(|E^*|)$ maximum flow problems in that graph. It is important to note, however, that these maximum flow computations are not 'independent' of each other, in the sense that the result of each maximum flow computation influences the choice of source and sink nodes in the subsequent one.

Now let us consider the primal version of 2-matching separation. In order for (12) to be tight for a given tour $\bar{x}$, one of two conditions must hold. Either

- a) $\bar{x}_e = 1$ for all $e \in F$ and $\bar{x}(\delta(H) \setminus F) = 1$, or
- b) $\bar{x}(F) = |F| - 1$ and $\bar{x}_e = 0$ for all $e \in \delta(H) \setminus F$.

Given this fact, it is not difficult to see that the primal 2-matching separation problem can be solved by the following algorithm:

- Set $w = 1$.
- For $i = 0, 1$, let $E^i := \{e \in E^* : \bar{x}_e = i\}$.
- Create a copy of G^* and assign a weight of x_e^* to each edge in E^0 and a weight of $1 - x_e^*$ to each edge in E^1.
- For each $e = \{u, v\} \in E^0$, do the following:
  - Change the weight of e from x_e^* to $1 - x_e^*$.
  - Compute a maximum flow from u to v. Let $w(e)$ be the value of the maximum flow and let $H(u)$ be the shore of the minimum $\{u, v\} - cut$ which contains e.
  - If $w(e) < w$, set $H := H(u)$, $w := w(e)$ and $F := \{e\} \cup (\delta(H(u)) \cap E^1)$.
  - Change the weight of e from $1 - x_e^*$ back to x_e^*.
- For each $e = \{u, v\} \in E^1$, do the following:
  - Change the weight of e from $1 - x_e^*$ to x_e^*.
  - Compute a maximum flow from u to v. Let $w(e)$ be the value of the maximum flow and let $H(u)$ be the shore of the minimum $\{u, v\} - cut$ which contains e.
  - If $w(e) < w$, set $H := H(u)$, $w := w(e)$ and $F := (\delta(H(u)) \cap E^1) \setminus \{e\}$.
  - Change the weight of e from x_e^* back to $1 - x_e^*$.
- If $w < 1$, then H and F give a 2-matching inequality which is violated by the largest amount.

This is faster than the algorithm for standard separation because all maximum flow computations are carried out directly on G^*. Moreover, it is simpler, because:

- Each maximum flow computation can be carried out independently,
- There is no need to label nodes odd or even.

3.7 Simple comb inequalities

The 2-matching inequalities are a special case of the well-known (and facet-inducing) *comb* inequalities for the STSP (see again Padberg and Grötschel 1985). Again, these can be written in various ways. For our purposes, it will be useful to write them in the classical form:

$$x(E(H)) + \sum_{j=1}^{p} x(E(T_j)) \ \leq \ |H| + \sum_{j=1}^{p} |T_j| - (3p + 1)/2. \qquad (13)$$

Here, H and T_j for $j = 1, \ldots, p$ must satisfy:

- $p \geq 3$ and odd,
- $\emptyset \neq H \subset V$,
- $T_j \subset V$, $T_j \cap H \neq \emptyset$ and $T_j \setminus H \neq \emptyset$ for $j = 1, \ldots, p$,

– $T_i \cap T_j = \emptyset$ for $1 \le i < j \le p$.

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

When $|T_j \cap H| = |T_j \setminus H| = 1$ for all j, the comb inequalities reduce to the 2-matching inequalities.

In a recent paper (Letchford and Lodi 2002b), we devised a polynomial-time (standard) separation algorithm for the so-called *simple* comb inequalities. A comb is *simple* if, for all j, either $|T_j \cap H| = 1$ or $|T_j \setminus H| = 1$ (or both).

The algorithm essentially consists of two steps. In the first step, which takes $\mathcal{O}(n^3|E^*|)$ time, a collection of $\mathcal{O}(n^3)$ 'candidate' simple teeth is constructed. In the second step, the separation problem is converted into a series of minimum weight odd cut computations, each of which is solved by the Padberg-Rao algorithm (Padberg and Rao 1982). This second step takes $|E^*|^4 \log n$ time.

Although we have not been able to reduce the asymptotic running time of this algorithm in the primal case, we have found that the first step can be simplified considerably. To explain how, we need to explain that a node i and a set S form a candidate simple tooth if and only if the sum of the slacks on the SECs on S and $S \cup \{i\}$ is less than 1 (equivalently, if $2x^*(E(S)) + x^*(E(\{i\} : S)) > 2|S| - 2$). To find these teeth, the authors used the near-minimum cut algorithm of Nagamochi et al. (1997) which, though polynomial, is rather tricky to implement.

It turns out that finding these candidate teeth is almost trivial in the primal case. For simplicity of notation, and without loss of generality, we suppose from now on that the vertices have been re-numbered so that the tour represented by $\bar{x}$ passes through the vertices $1, \ldots, |V|$ in cyclic order. We also take indices modulo $|V|$.

Lemma 1. *A vertex i and a set S satisfy $2\bar{x}(E(S)) + \bar{x}(E(\{i\} : S)) = 2|S| - 1$ if and only if one of the following two conditions holds:*

(i) $S = \{i + 1, \ldots, j\}$ *for some* $j \in \{i + 1, \ldots, i + |V| - 2\}$.
(ii) $S = \{j, \ldots, i - 1\}$ *for some* $j \in \{i + 2, \ldots, i + |V| - 1\}$.

Proof. The condition essentially says that the SECs on S and $S \cup \{i\}$ must be tight at $\bar{x}$. This implies that the nodes in S are consecutive in the tour and that node i is either immediately before or after S in the tour. □

Lemma 2. *A vertex i and a set S satisfy $2\bar{x}(E(S)) + \bar{x}(E(\{i\} : S)) = 2|S| - 2$ if and only if one of the following two conditions holds:*

– *(i)* $S = \{j, \ldots, k\}$ *for some* $i + 2 \le j \le k \le i + |V| - 2$.
– *(ii)* $S = \{j, \ldots, k\} \setminus \{i\}$ *for some* $i - |V| + 3 \le j \le k \le j + |V| - 2$.

Proof. The condition holds if and only if the SEC on S is tight, and the SEC on $S \cup \{i\}$ has a slack of exactly 1. This implies that the nodes in S are consecutive in the tour and that node i is not adjacent to any node in S. □

Now recall that primal separation for x^* and $\bar{x}$ can be converted into standard separation for the point $\tilde{x} = \epsilon x^* + (1 - \epsilon)\bar{x}$, where ϵ is some suitable small quantity.

Any candidate tooth at $\tilde{x}$ must satisfy $2\tilde{x}(E(S))+\tilde{x}(E(\{i\}:S)) > 2|S|-2$. Since $\tilde{x}$ can be made arbitrarily close to $\bar{x}$, such a candidate tooth must satisfy the conditions in one of the above two lemmas. We can search for such candidate teeth simply by scanning through all i and all sets S satisfying the conditions. It is easy to check that this can be performed in $\mathcal{O}(n^2|E^*|)$ time. In fact, the bottleneck of this operation is actually storing the sets (which may take $\mathcal{O}(n^3|E^*|)$ space to represent).

We close this section with a 'historical note'. We discovered a polynomial-time *primal* separation algorithm for the simple comb inequalities in 2001 (see Letchford and Lodi 2001). At the time, the complexity of the standard separation problem for these inequalities was unknown. It was the existence of the primal separation algorithm which inspired us to search for the standard separation algorithm presented in Letchford and Lodi (2002b).

4 Conclusion

As mentioned in the introduction, it is known that primal separation is never more difficult than standard separation. We have shown that, in some cases, primal separation is also no *easier*. However, in several important cases we have shown that primal separation is indeed easier. This is also confirmed by the observations of Eisenbrand et al. (2003), and suggests that primal cutting plane algorithms should be re-examined, particularly for the symmetric TSP.

References

Applegate D, Bixby RE, Chvátal V, Cook W (1998) On the solution of travelling salesman problems. *Documenta Mathematica*, Extra Volume ICM III, pp 645–656

Balas E (1975) Facets of the knapsack polytope. *Mathematical Programming* 8: 146–164

Balas E (1989) The asymmetric assignment problem and some new facets of the travelling salesman polytope. *SIAM Journal of Discrete Mathematics* 2: 425–451

Bollobás B, Erdos P (1976) Cliques in random graphs. *Proc. Camb. Phil. Soc.* 80: 419–427

Borndörfer R, Weismantel R (2000) Set packing relaxations of integer programs. *Mathematical Programming* 88: 425–450

Caprara A, Fischetti M (1996) {0, 1/2}-Chvátal-Gomory cuts. *Mathematical Programming* 74: 221–235

Caprara A, Fischetti M (1997) Branch-and-Cut Algorithms. In: Dell'Amico M, Maffioli F, Martello S (eds) *Annotated Bibliographies in Combinatorial Optimization*. Wiley, New York, pp 45–64

Crowder H, Johnson EL, Padberg MW (1983) Solving large-scale zero-one linear programming problems. *Operations Research* 31: 803–834

Eisenbrand F (1999) On the membership problem for the elementary closure of a polyhedron. *Combinatorica* 19: 297–300

Eisenbrand F, Rinaldi G, Ventura P (2003) Primal separation for 0/1 polytopes. *Mathematical Programming* 95: 475–491

Ferreira CE (1994) On Combinatorial Optimization Problems Arising in Computer Systems Design. PhD Thesis, Technische Universität Berlin

Fischetti M, Toth P (1997) A polyhedral approach to the asymmetric travelling salesman problem. *Management Science* 43: 1520–1536

Garey MR, Johnson DS (1979) *Computers and Intractibility: A Guide to the Theory of $\mathcal{NP}$-completeness*. Freeman, San Francisco

Gerards AMH, Schrijver A (1986) Matrices with the Edmonds-Johnson property. *Combinatorica* 6: 365–379

Grötschel M, Lovász L, Schrijver AJ (1988) *Geometric Algorithms and Combinatorial Optimization*. Wiley, New York

Gu Z, Nemhauser GL, Savelsbergh MWP (1998) Lifted cover inequalities for 0-1 integer programs: computation. *INFORMS Journal on Computing* 10: 427–437

Letchford AN, Lodi A (2003) An augment-and-branch-and-cut framework for mixed 0-1 programming. In: Jünger M, Reinelt G, Rinaldi G (eds) *Eureka, you Shrink!* (Springer-LNCS 2570). Springer, Berlin Heidelberg New York, pp 119–133

Letchford AN, Lodi A (2001) Primal separation algorithms. Technical report, D.E.I.S., University of Bologna, OR/01/5. Unpublished

Letchford AN, Lodi A (2002a) Primal cutting plane algorithms revisited. *Mathematical Methods of Operations Research* 56: 67–81

Letchford AN, Lodi A (2002b) Polynomial-time separation of simple comb inequalities. In: Cook WJ, Schulz AS (eds) *Integer Programming and Combinatorial Optimization - IPCO 2002* (LNCS 2337). Springer, Berlin Heidelberg New York, pp 96–111

Nagamochi H, Nishimura K, Ibaraki T (1997) Computing all small cuts in undirected networks. *SIAM Journal of Discrete Mathematics* 10: 469–481

Nagamochi H, Ono T, Ibaraki T (1994) Implementing an efficient minimum capacity cut algorithm. *Mathematical Programming* 67: 325–341

Nemhauser GL, Wolsey LA (1988) *Integer and Combinatorial Optimization*. Wiley, New York

Padberg MW (1973) On the facial structure of set packing polyhedra. *Mathematical Programming* 5: 199–215

Padberg MW, Grötschel M (1985) Polyhedral computations. In: Lawler E, Lenstra, J, RinnooyKan A, Shmoys D (eds) *The Traveling Salesman Problem*. Wiley, Chichester, pp 307–360

Padberg MW, Hong S (1980) On the symmetric travelling salesman problem: a computational study. *Mathematical Programming Study* 12: 78–107

Padberg MW, Rao MR (1982) Odd minimum cut-sets and *b*-matchings. *Mathematics of Operations Research* 7: 67–80

Padberg MW, Rinaldi G (1991) A branch-and-cut algorithm for the resolution of large-scale symmetric travelling salesman problems. *SIAM Review* 33: 60–100

Papadimitriou CH, Yannakakis M (1984) The complexity of facets (and some facets of complexity). *Journal of Computer and Systems Science* 28: 244–259

Schrijver A (1986) *Theory of Linear and Integer Programming*. Wiley, New York