Math. Program., Ser. A 106, 261–285 (2006)

Digital Object Identifier (DOI) 10.1007/s10107-005-0606-3

Ted K. Ralphs · Matthew V. Galati

Decomposition and Dynamic Cut Generation in Integer Linear Programming

Received: 19 September 2003 / Accepted: 15 April 2005
Published online: August 10, 2005 – © Springer-Verlag 2005

Abstract. Decomposition algorithms such as Lagrangian relaxation and Dantzig-Wolfe decomposition are well-known methods that can be used to generate bounds for mixed-integer linear programming problems. Traditionally, these methods have been viewed as distinct from polyhedral methods, in which bounds are obtained by dynamically generating valid inequalities to strengthen an initial linear programming relaxation. Recently, a number of authors have proposed methods for integrating dynamic cut generation with various decomposition methods to yield further improvement in computed bounds. In this paper, we describe a framework within which most of these methods can be viewed from a common theoretical perspective. We then discuss how the framework can be extended to obtain a decomposition-based separation technique we call *decompose and cut*. As a by-product, we describe how these methods can take advantage of the fact that solutions with known structure, such as those to a given relaxation, can frequently be separated much more easily than arbitrary real vectors.

Key words. Integer Programming – Dantzig-Wolfe Decomposition – Lagrangian Relaxation – Branch and Cut – Branch and Price – Decomposition Algorithms

1. Introduction

In this paper, we discuss methods for computing bounds on the value of an optimal solution to a mixed-integer linear program (MILP). Computing such bounds is an essential element of the branch and bound algorithm, which is the most effective and most commonly used method for solving general MILPs. Bounds are generally computed by solving a *bounding subproblem*, which is either a relaxation or a dual of the original problem. The most commonly used bounding subproblem is the linear programming (LP) relaxation. The LP relaxation is often too weak to be effective, but it can be strengthened by the addition of dynamically generated valid inequalities. Alternatively, traditional decomposition techniques, such as Dantzig-Wolfe decomposition [19] or Lagrangian relaxation [22, 14], can also be used to obtain improved bounds.

Methods based on cut generation have traditionally been considered distinct from decomposition methods, but several authors have suggested combining the two approaches to yield further improvements (these contributions are reviewed in Section 3). In this paper, we present a framework that shows how these hybrid methods can be

T. K. Ralphs: Dept. of Industrial and Systems Engineering, Lehigh University, Bethlehem, PA, USA.
e-mail: tkralphs@lehigh.edu

M. V. Galati: Analytical Solutions - Operations R & D, SAS Institute, Cary, NC, USA.
e-mail: matthew.galati@sas.com

viewed from a common theoretical perspective as generalizations of the cutting plane method. From this viewpoint, we show how various methods from the literature are related, as well as discuss a method called *decompose and cut* that follows a similar paradigm. One element common to these methods is the ability to take advantage of the fact that the separation problem is often much easier for solutions with combinatorial structure than for arbitrary real vectors. This can simplify the implementation of these methods and allow more rapid development than with traditional cutting plane implementations. We discuss the importance of this and provide several examples of its usefulness.

The goal of this paper is not to provide a computational comparison. Such comparisons are necessarily problem-dependent and based largely on empirical evidence. Although some general statements can be made, the lines between the various bounding techniques presented here are blurry at best and many subtle variations are possible. Our goal is to provide the reader with insight that may be useful in guiding the choice of method. By illustrating the relationships between various techniques, we provide a methodological framework within which it is easy to switch between variants by replacing certain algorithmic components. We are currently developing a software framework that provides just such a capability. A more detailed treatment of this material is also provided in [56].

To simplify the exposition, we consider only pure integer linear programming problems (ILPs) with bounded feasible regions, although the framework can be extended to more general settings. For the remainder of the paper, we consider an ILP whose feasible set is the integer vectors contained in the polyhedron $\mathcal{Q} = \{x \in \mathbb{R}^n : Ax \geq b\}$, where $A \in \mathbb{Q}^{m \times n}$ is the constraint matrix and $b \in \mathbb{Q}^m$ is the right-hand-side vector. Let $\mathcal{F} = \mathcal{Q} \cap \mathbb{Z}^n$ be the feasible set and let $\mathcal{P}$ be the convex hull of $\mathcal{F}$.

We consider two problems associated with the polyhedron $\mathcal{P}$. The *optimization problem* for $\mathcal{P}$ is that of determining

$$z_{IP} = \min_{x \in \mathbb{Z}^n} \left\{ c^\top x \mid Ax \geq b \right\} = \min_{x \in \mathcal{P}} \left\{ c^\top x \right\} = \min_{x \in \mathcal{F}} \left\{ c^\top x \right\} \tag{1}$$

for a given cost vector $c \in \mathbb{Q}^n$, where z_{IP} is set to infinity if $\mathcal{F}$ is empty. A related problem is the *separation problem* for $\mathcal{P}$. Given $x \in \mathbb{R}^n$, the problem of separating x from $\mathcal{P}$ is that of deciding whether $x \in \mathcal{P}$ and if not, determining $a \in \mathbb{R}^n$ and $\beta \in \mathbb{R}$ such that $a^\top y \geq \beta \ \forall y \in \mathcal{P}$ but $a^\top x < \beta$. A pair $(a, \beta) \in \mathbb{R}^{n+1}$ such that $a^\top y \geq \beta \ \forall y \in \mathcal{P}$ is a *valid inequality* for $\mathcal{P}$ and is said to be *violated* by $x \in \mathbb{R}^n$ if $a^\top x < \beta$. In [30], it was shown that the separation problem for $\mathcal{P}$ is polynomially equivalent to the optimization problem for $\mathcal{P}$.

To apply the principle of decomposition, we consider the relaxation of (1) defined by

$$\min_{x \in \mathbb{Z}^n} \left\{ c^\top x \mid A'x \geq b' \right\}, = \min_{x \in \mathcal{P}'} \left\{ c^\top x \right\} = \min_{x \in \mathcal{F}'} \left\{ c^\top x \right\} \tag{2}$$

where $\mathcal{F} \subset \mathcal{F}' = \{x \in \mathbb{Z}^n \mid A'x \geq b'\}$ for some $A' \in \mathbb{Q}^{m' \times n}$, $b' \in \mathbb{Q}^{m'}$ and $\mathcal{P}'$ is the convex hull of $\mathcal{F}'$. As usual, we assume that there exists an *effective* algorithm for optimizing over $\mathcal{P}'$. We are deliberately using the term *effective* here to denote an algorithm that has an acceptable average-case running time, since this is the relevant measure of running time for our purposes. Along with $\mathcal{P}'$ is associated a set of *side* *constraints*. Let $[A'', b''] \in \mathbb{Q}^{m'' \times n}$ be a set of additional inequalities needed to describe $\mathcal{F}$, i.e., $[A'', b'']$ is such that $\mathcal{F} = \{x \in \mathbb{Z}^n \mid A'x \geq b', A''x \geq b''\}$. We denote by $\mathcal{Q}'$ the polyhedron described by the inequalities $[A', b']$ and by $\mathcal{Q}''$ the polyhedron described by the inequalities $[A'', b'']$. Hence, the initial LP relaxation is the linear program defined by $\mathcal{Q} = \mathcal{Q}' \cap \mathcal{Q}''$, and the *LP bound* is given by

$$z_{LP} = \min_{x \in \mathbb{R}^n} \left\{ c^\top x \mid A'x \geq b', A''x \geq b'' \right\} = \min_{x \in \mathcal{Q}} \left\{ c^\top x \right\}. \tag{3}$$

This is slightly more general than the traditional framework, in which $[A', b']$ and $[A'', b'']$ are a partition of the rows of $[A, b]$.

2. Traditional Decomposition Methods

The goal of the decomposition approach is to improve on the LP bound by taking advantage of our ability to optimize over and/or separate from $\mathcal{P}'$. We briefly review the classical bounding methods that take this approach in order to establish terminology and notation.

Lagrangian Relaxation. For a given vector of *dual multipliers* $u \in \mathbb{R}_+^{m''}$, the *Lagrangian relaxation* of (1) is given by

$$z_{LR}(u) = \min_{s \in \mathcal{F}'} \left\{ \left(c^\top - u^\top A'' \right) s + u^\top b'' \right\} \tag{4}$$

It is easily shown that $z_{LR}(u)$ is a lower bound on z_{IP} for any $u \geq 0$. The problem

$$z_{LD} = \max_{u \in \mathbb{R}_+^{m''}} \left\{ z_{LR}(u) \right\} \tag{5}$$

of maximizing this bound over all choices of dual multipliers is a dual to (1) called the *Lagrangian dual* (LD) and also provides the lower bound z_{LD}, which we call the *LD bound*. A vector of multipliers that yield the largest bound are called *optimal (dual) multipliers*. For the remainder of the paper, let $\hat{u}$ be such a vector.

The Lagrangian dual can be solved by any of a number of subgradient-based optimization procedures or by rewriting it as the equivalent linear program

$$z_{LD} = \max_{\alpha \in \mathbb{R}, u \in \mathbb{R}_+^{m''}} \left\{ \alpha + u^\top b'' \mid \alpha \leq \left(c^\top - u^\top A'' \right) s \ \forall s \in \mathcal{F}' \right\}, \tag{6}$$

and solving it using a cutting plane algorithm. In any case, the main computational effort is in evaluating $z_{LR}(u)$ for a given sequence of dual multipliers u. This is an optimization problem over $\mathcal{P}'$, which we assumed could be solved effectively. This general approach is described in more detail in [32].

Dantzig-Wolfe Decomposition. The approach of Dantzig-Wolfe decomposition is to reformulate (1) by implicitly requiring the solution to be a member of $\mathcal{F}'$, while explicitly enforcing the inequalities $[A'', b'']$. Relaxing the integrality constraints of this reformulation, we obtain the linear program

$$z_{DW} = \min_{\lambda \in \mathbb{R}_{+}^{\mathcal{F}'}} \left\{ c^{\top} \left(\sum_{s \in \mathcal{F}'} s\lambda_s \right) \;\middle|\; A'' \left(\sum_{s \in \mathcal{F}'} s\lambda_s \right) \geq b'', \sum_{s \in \mathcal{F}'} \lambda_s = 1 \right\}, \tag{7}$$

which we call the *Dantzig-Wolfe LP* (DWLP). Although the number of columns in this linear program is $|\mathcal{F}'|$, it can be solved by dynamic column generation, where the column-generation subproblem is again an optimization problem over $\mathcal{P}'$ equivalent to that of evaluating $z_{LR}(u)$ for a vector u arising as the dual multipliers of the constraints of (7).

It is easy to verify that the DWLP is the dual of (6), which immediately shows that $z_{DW} = z_{LD}$ (see [52] for a detailed treatment of this fact). Hence, z_{DW} is a valid lower bound on z_{IP} that we call the *DW bound*. Note that if we let $\hat{\alpha} = z_{LR}(\hat{u}) - \hat{u}^{\top} b''$, then $(\hat{u}, \hat{\alpha})$ is an optimal solution for the LP (6) and hence also an optimal dual solution to the DWLP. An optimal primal solution to (7) is referred to as an *optimal Dantzig-Wolfe (DW) decomposition*. For the remainder of the paper, let $\hat{\lambda}$ be such a solution. If we combine the members of $\mathcal{F}'$ using $\hat{\lambda}$, to obtain

$$\hat{x}_{DW} = \sum_{s \in \mathcal{F}'} s\hat{\lambda}_s, \tag{8}$$

then we see that $z_{DW} = c^{\top}\hat{x}_{DW}$. Since $\hat{x}_{DW}$ must lie within $\mathcal{P}' \subseteq \mathcal{Q}'$ and also within $\mathcal{Q}''$, this shows that $z_{DW} \geq z_{LP}$. A general treatment of Dantzig-Wolfe decomposition can be found in [45].

Cutting Plane Method. In the cutting plane method, inequalities describing $\mathcal{P}'$ (i.e., the facet-defining inequalities) are generated dynamically by separating the solutions to a series of LP relaxations from $\mathcal{P}'$. In this way, the initial LP relaxation is iteratively augmented to obtain the *CP bound*,

$$z_{CP} = \min_{x \in \mathbb{R}^n} \left\{ c^{\top}x \mid A''x \geq b'', Dx \geq d \right\} = \min_{x \in \mathcal{P}' \cap \mathcal{Q}''} \left\{ c^{\top}x \right\}, \tag{9}$$

where $[D, d]$ are the facet-defining inequalities for $\mathcal{P}'$. We refer to this augmented linear program as the *cutting plane LP* (CPLP) and any optimal solution to CPLP as an *optimal fractional solution*. For the remainder of the paper, let $\hat{x}_{CP}$ be such a solution. Note that $\hat{x}_{DW}$, as defined in (8), is an optimal solution to this augmented linear program. Hence, the CP bound is equal to both the DW bound and the LD bound. A general treatment of the cutting plane method can be found in [50].

A Common Framework. The following well-known result of Geoffrion [26] relates the three methods just described.

Theorem 1. $z_{IP} \geq z_{LD} = z_{DW} = z_{CP} \geq z_{LP}$.

A graphical depiction of this common bound is shown in Figure 1. Theorem 1 shows that (5), (7), and (9) represent three different formulations for the problem of computing this bound. As such, the methods we have just described are really only distinguished

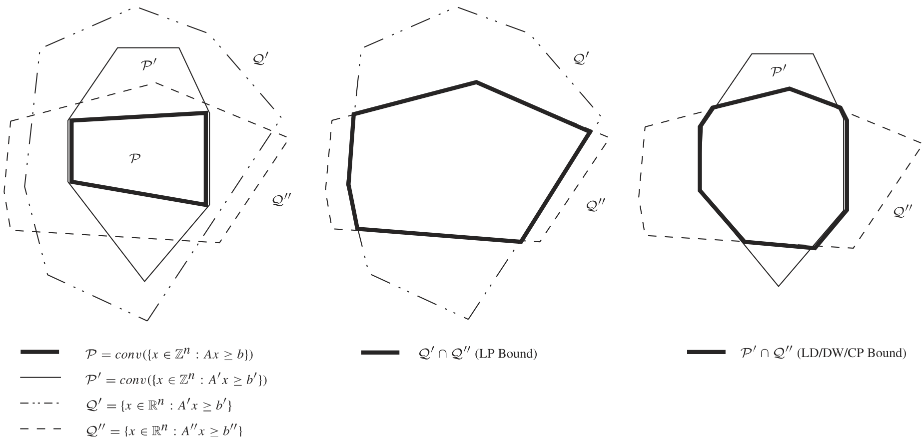

Fig. 1. Illustration of the LP and LD/DW/CP bounds

by the solution algorithms typically applied in each case, as well as by the auxiliary solution information each formulation yields. More is said about algorithms for solving these bounding subproblem in Section 3.4. For now, we focus on developing a common way of viewing these methods.

As we have seen, the basis for each of the methods is that we are given a polyhedron $\mathcal{P}$ over which we would like to optimize, along with two additional polyhedra, denoted here by $\mathcal{Q}''$ and $\mathcal{P}'$, each of which contain $\mathcal{P}$. The polyhedron $\mathcal{Q}''$ typically has a small description that can be represented explicitly, while the polyhedron $\mathcal{P}'$ has a much larger description and is represented implicitly, i.e., portions of the description are generated dynamically using our ability to effectively optimize/separate. To describe their roles in this framework, we call $\mathcal{P}$ the *original polyhedron*, $\mathcal{Q}''$ the *explicit polyhedron*, and $\mathcal{P}'$ the *implicit polyhedron*. Note that, although traditional decomposition methods insist that $\mathcal{Q}''$ have a small description, the methods described in Section 3 allow portions of an outer description of $\mathcal{Q}''$ to also be generated dynamically. In Section 4, we describe some applications in which this is the case.

By the Weyl-Minkowski Theorem, every bounded rational polyhedron has two descriptions—one as the intersection of half-spaces (the *outer representation*) and one as the convex hull of its extreme points (the *inner representation*) [51]. The conceptual difference between the formulations utilized by the three methods just reviewed is that Dantzig-Wolfe decomposition and Lagrangian relaxation utilize an inner representation of $\mathcal{P}'$, generated dynamically by solving the corresponding optimization problem, whereas the cutting plane method relies on an outer representation of $\mathcal{P}'$, generated dynamically by solving the separation problem for $\mathcal{P}'$. For this reason, we call Dantzig-Wolfe decomposition and Lagrangian relaxation *inner methods*, and the cutting plane method an *outer method*. In theory, we have the same choice of representation for the explicit polyhedron—it is intriguing to ponder the implications of this choice. Note that this framework encompasses methods not normally thought of as decomposition methods. In particular, by defining the implicit polyhedron to be a polyhedron defined by classes of inequalities for which there are effective separation algorithms, we can view cutting plane methods as just another type of decomposition method. This viewpoint sheds new light on their relationship to traditional decomposition methods.

3. Integrated Decomposition Methods

One of the apparent advantages of outer methods over inner methods is the option of adding heuristically generated inequalities valid for $\mathcal{P}$ to the cutting plane LP to improve the bound discussed in Section 2. Such inequalities may "cut off" portions of $\mathcal{P}'$ to yield a better outer approximation of $\mathcal{P}$. This dynamic generation of additional valid inequalities in outer methods can be thought of as a dynamic tightening of either the explicit or the implicit polyhedron. Such a tightening procedure can also be incorporated into inner methods in a straightforward way. Viewing these inequalities as dynamically tightening the explicit polyhedron yields a generalization of the cutting plane method obtained by replacing the cutting plane LP with either a Dantzig-Wolfe LP or a Lagrangian dual as the bounding subproblem. We call this class of bounding procedures *integrated decomposition methods* because they integrate inner and outer methods. The steps of this generalized method are shown in Figure 2.

The important step in Figure 2 is Step 2, generating a set of *improving inequalities*, i.e., inequalities valid for $\mathcal{P}$ that when added to the description of the explicit polyhedron result in an increase in the computed bound. Putting aside the question of exactly how Step 2 is to be accomplished, the approach is straightforward. Step 1 is performed as in a traditional decomposition framework. Step 3 is accomplished by simply adding the newly generated inequalities to the list $[A'', b'']$ and reforming the appropriate bounding subproblem. Note that it is also possible to develop an analog based on an interpretation of the cutting plane method as a dynamic tightening of the implicit polyhedron [62]. In this case, the implicit polyhedron associated with the subproblem to be solved in Step 1 may also change dynamically. We have not yet investigated this class of methods.

Although some forms of this general method have appeared in the literature, they have received little attention thus far and naming conventions are not well-established. We would like to suggest here a naming convention that emphasizes the close relationship of these methods to each other. When the bounding subproblem is a Dantzig-Wolfe

An Integrated Decomposition Method
<u>Input</u>: An instance of ILP.
<u>Output</u>: A lower bound on the optimal solution value for the instance.

1. Solve the bounding subproblem, which is one of
   $z_{CP} = \min_{x \in \mathcal{P}'} \{c^\top x \mid A'' x \geq b''\}$,
   $z_{LD} = \max_{u \in \mathbb{R}^{m''}_{+}} \min_{s \in \mathcal{F}'} \{(c^\top - u^\top A'')s + u^\top b''\}$, or
   $z_{DW} = \min_{\lambda \in \mathbb{R}^{\mathcal{F}'}_{+}} \{c^\top (\sum_{s \in \mathcal{F}'} s \lambda_s) \mid A''(\sum_{s \in \mathcal{F}'} s \lambda_s) \geq b'', \sum_{s \in \mathcal{F}'} \lambda_s = 1\}$,
   to obtain a valid lower bound z.
2. Attempt to generate a set of improving inequalities $[\hat{D}, \hat{d}]$ valid for $\mathcal{P}$.
3. If valid inequalities were found in Step 2, form a new bounding subproblem by setting $[A'', b''] \leftarrow \left[\begin{smallmatrix} A'' & b'' \\ \hat{D} & \hat{d} \end{smallmatrix}\right]$. Then, go to Step 1.
4. If no valid inequalities were found in Step 2, then output z.

Fig. 2. Basic outline of an integrated decomposition method

LP, we call the resulting method *price and cut*. When employed in a branch and bound framework, the overall technique is called *branch, price, and cut*. This method has been studied by a number of authors [11, 12, 35, 60, 61] and is described in more detail in Section 3.1. When the bounding subproblem is a Lagrangian dual, we call the method *relax and cut*. When relax and cut is used as the bounding procedure in a branch and bound framework, we call the overall method *branch, relax, and cut*. Variants of this method have also been studied previously by several authors (see [42] for a survey) and is described in more detail in Section 3.2. Finally, in Section 3.3, we describe a variant of the cutting plane method that employs a decomposition-based separation procedure. We call this method *decompose and cut* and embed it within a branch and bound framework to obtain the method *branch, decompose, and cut*.

As we alluded to earlier, the distinction between what we call price and cut and what we call relax and cut may not be that easy to make, since modern methods for solving the subproblems in each case can be quite similar. The rough distinction we make between them here, however, is in the amount of primal solution information produced as a by-product of the solution process. When solving a Dantzig-Wolfe LP, we assume that an optimal DW decomposition is produced *exactly*. When solving a Lagrangian dual, we assume only approximate primal solution information, if any at all, is available. Solution methods for these subproblems are discussed in more detail in Section 3.4.

As we have already mentioned, Step 2 is the crux of integrated decomposition methods. In the context of the cutting plane method, this step is usually accomplished by applying one of the many known techniques for separating $\hat{x}_{CP}$ from $\mathcal{P}$ (see [1]). Violation of $\hat{x}_{CP}$ is a necessary condition for an inequality to be improving, and hence such an inequality is *likely* to be effective. However, unless the inequality separates the entire optimal face F to the cutting plane LP, it will not be improving. Because we want to refer to these well-known results later in the paper, we state them formally as theorem and corollary without proof. See [59] for a thorough treatment of the theory of linear programming that leads to this result.

Theorem 2. *Let F be the face of optimal solutions to an LP solved directly over $\mathcal{P}' \cap \mathcal{Q}''$ with objective function c. Then $(a, \beta) \in \mathbb{R}^{n+1}$ is an improving inequality if and only if $a^\top x < \beta$ for all $x \in F$.*

Corollary 1. *If $(a, \beta) \in \mathbb{R}^{n+1}$ is an improving inequality, then $a^\top \hat{x}_{CP} < \beta$.*

Fortunately, even in the case when F is not separated in its entirety, the augmented cutting plane LP must have a different optimal solution, which in turn may be used to generate more potential improving inequalities. Since the condition of Theorem 2 is difficult to verify, one typically terminates the bounding procedure when increases resulting from additional inequalities become “too small.” In the next two sections, we examine how improving inequalities can be generated when the bounding subproblem is either a Dantzig-Wolfe LP or a Lagrangian dual. We then return to the cutting plane method to discuss how decomposition can be used directly to aid in solving the separation problem.

3.1. Price and Cut

Finding Improving Inequalities. Using the Dantzig-Wolfe LP as the bounding subproblem in Figure 2 results in a procedure that alternates between generating columns and generating valid inequalities. Such concurrent generation of columns and valid inequalities is difficult in general because the addition of valid inequalities can destroy the structure of the column-generation subproblem (for a discussion of this, see [60]). Having solved the Dantzig-Wolfe LP, however, one can easily recover an optimal solution to the cutting plane LP using (8) and try to generate improving inequalities as in the cutting plane method. The generation of valid inequalities thus takes place in the original space and does not destroy the structure of the column-generation subproblem in the Dantzig-Wolfe LP. This approach enables dynamic generation of valid inequalities, while still retaining the bound improvement and other advantages yielded by Dantzig-Wolfe decomposition. A recent paper by Arãgao and Uchoa discusses this technique in more detail [21].

Because the same valid inequalities are generated with this method as would be generated in the cutting plane method, these two dynamic methods achieve the same bound in principle. Price and cut, however, produces additional primal information that we may be able to use to our advantage. In particular, the optimal DW decomposition $\hat{\lambda}$ provides a decomposition of $\hat{x}_{DW}$ into a convex combination of members of $\mathcal{F}'$. We refer to elements of $\mathcal{F}'$ that have a positive weight in this combination as *members of the decomposition*. The following theorem shows how such a decomposition can be used to derive an alternate necessary condition for an inequality to be improving.

Theorem 3. *If $x \in \mathbb{R}^n$ violates the inequality $(a, \beta) \in \mathbb{R}^{(n+1)}$ and $\lambda \in \mathbb{R}_+^{\mathcal{F}'}$ is such that $\sum_{s \in \mathcal{F}'} \lambda_s = 1$ and $x = \sum_{s \in \mathcal{F}'} s\lambda_s$, then there must exist an $s \in \mathcal{F}'$ with $\lambda_s > 0$ such that s also violates the inequality (a, β) .*

Proof. Let $x \in \mathbb{R}^n$ and $(a, \beta) \in \mathbb{R}^{(n+1)}$ be given such that $a^\top x < \beta$. Also, let $\lambda \in \mathbb{R}_+^{\mathcal{F}'}$ be given such that $\sum_{s \in \mathcal{F}'} \lambda_s = 1$ and $x = \sum_{s \in \mathcal{F}'} s\lambda_s$. Suppose that $a^\top s \geq \beta$ for all $s \in \mathcal{F}'$ with $\lambda_s > 0$. Since $\sum_{s \in \mathcal{F}'} \lambda_s = 1$, we have $a^\top (\sum_{s \in \mathcal{F}'} s\lambda_s) \geq \beta$. Hence, $a^\top x = a^\top (\sum_{s \in \mathcal{F}'} s\lambda_s) \geq \beta$, which is a contradiction. □

In other words, an inequality can be improving only if it is violated by at least one member of the decomposition. If $\mathcal{I}$ is the set of all improving inequalities, then the following corollary is a direct consequence of Theorem 3.

Corollary 2. $\mathcal{I} \subseteq \mathcal{V} = \{(a, \beta) \in \mathbb{R}^{(n+1)} \mid a^\top s < \beta$ *for some* $s \in \mathcal{F}'$ *such that* $\hat{\lambda}_s > 0\}$.

The importance of this result is that in many cases, it is easier to separate members of $\mathcal{F}'$ from $\mathcal{P}$ than to separate arbitrary real vectors. We call this approach *structured separation*. There are a number of well-known polyhedra for which the problem of separating an arbitrary real vector is difficult, but the problem of separating a solution to a given relaxation is easy. Some examples are discussed in Section 4. In Figure 3, we propose a new separation procedure to be embedded in price and cut that takes advantage of this fact.

The running time of the procedure in Figure 3 depends in part on the cardinality of the decomposition. Carathéodory's Theorem assures us that there exists a decomposition with less than or equal to $\dim(\mathcal{P}') + 1$ members. Unfortunately, even if we limit our search to a particular known class of inequalities, the number of such inequalities violated by each member of $\mathcal{D}$ in Step 2 may be extremely large and these inequalities may not be violated by $\hat{x}_{DW}$ (such an inequality cannot be improving). Unless we enumerate *every* inequality in the set $\mathcal{V}$ from Corollary 2, either implicitly or explicitly,

Separation using a Dantzig-Wolfe Decomposition
Input: A DW decomposition $\hat{\lambda}$.
Output: A set of potentially improving inequalities for $\mathcal{P}$.

1. Form the set $\mathcal{D} = \{s \in \mathcal{F}' : \hat{\lambda}_s > 0\}$.
2. For each $s \in \mathcal{D}$, attempt to separate s from $\mathcal{P}$ to obtain a set $[D^s, d^s]$ of violated inequalities.
3. Let $[D^x, d^x]$ be composed of the inequalities found in Step 2 that are also violated by $\hat{x}_{DW}$.
4. Return $[D^x, d^x]$ as the set of potentially improving inequalities.

Fig. 3. Separation using a Dantzig-Wolfe decomposition

the procedure does not guarantee that an improving inequality will be found, even if one exists. In cases where it is possible to examine the set $\mathcal{V}$ in polynomial time, the worst-case complexity of the entire procedure is the same as that of optimizing over $\mathcal{P}'$. Obviously, it is thus unlikely that the set $\mathcal{V}$ can be examined in polynomial time in situations in which the separation problem for the class in question is $\mathcal{NP}$-complete. In such cases, the procedure to select inequalities that are likely to be violated by $\hat{x}_{DW}$ in Step 2 is necessarily a problem-dependent heuristic. The effectiveness of this heuristic can be improved in a number of ways, some of which are discussed in [57]. Further details will be provided in a companion paper on the computational aspects of these methods.

Connections to Other Methods. By making connections to the cutting plane method, we can gain further insight. Consider the set

$$\mathcal{S} = \left\{s \in \mathcal{F}' \mid \left(c^\top - \hat{u}^\top A''\right)s = \hat{\alpha}\right\}, \tag{10}$$

where $\hat{\alpha}$ is again defined to be $z_{LR}(\hat{u}) - \hat{u}^\top b''$, so that $(\hat{u}, \hat{\alpha})$ is an optimal dual solution to the DWLP. Since $\mathcal{S}$ is comprised exactly of those members of $\mathcal{F}'$ corresponding to columns of the Dantzig-Wolfe LP with reduced cost zero, complementary slackness guarantees that the set $\mathcal{S}$ must contain all members of the decomposition. The following theorem follows directly from this observation.

Theorem 4. *conv$(\mathcal{S})$ is a face of $\mathcal{P}'$ and contains $\hat{x}_{DW}$.*

Proof. We first show that conv$(\mathcal{S})$ is a face of $\mathcal{P}'$. First, note that

$$\left(c^\top - \hat{u}^\top A'', \hat{\alpha}\right)$$

defines a valid inequality for $\mathcal{P}'$, since $\hat{\alpha}$ was defined to be $z_{LR}(\hat{u}) - \hat{u}^\top b''$, which is the optimal value of a solution to the problem of minimizing over $\mathcal{P}'$ with objective function $c^\top - \hat{u}^\top A''$. Thus, the set

$$G = \left\{x \in \mathcal{P}' \mid \left(c^\top - \hat{u}^\top A''\right)x = \hat{\alpha}\right\}, \tag{11}$$

is a face of $\mathcal{P}'$ that contains $\mathcal{S}$. We claim that conv$(\mathcal{S}) = G$. Since G is convex and contains $\mathcal{S}$, it also contains conv$(\mathcal{S})$, so we just need to show that conv$(\mathcal{S})$ contains G. We do so by showing that the extreme points of G are members of $\mathcal{S}$. By construction, all extreme points of $\mathcal{P}'$ are members of $\mathcal{F}'$. Furthermore, the extreme points of G are also extreme points of $\mathcal{P}'$ and therefore must be members of $\mathcal{F}'$. It follows that the extreme points of G must be members of $\mathcal{S}$. Hence, $\text{conv}(\mathcal{S}) = G$ and $\text{conv}(\mathcal{S})$ is a face of $\mathcal{P}'$.

The fact that $\hat{x}_{DW} \in \text{conv}(\mathcal{S})$ follows from the fact that $\hat{x}_{DW}$ is a convex combination of members of $\mathcal{S}$. □

An important consequence of this result is contained in the following corollary.

Corollary 3. *If F is the face of optimal solutions to an LP solved over $\mathcal{P}' \cap \mathcal{Q}''$ with objective function c, then $F \subseteq conv(\mathcal{S}) \cap \mathcal{Q}''$.*

Proof. Let $x \in F$ be given. Then $x \in \mathcal{P}' \cap \mathcal{Q}''$ by definition and also

$$c^\top x = z_{CP} = z_{LD} = \hat{\alpha} + \hat{u}^\top b'' = \hat{\alpha} + \hat{u}^\top A'' x, \tag{12}$$

where the last equality in this chain is a consequence of complementary slackness. It follows that $(c^\top - \hat{u}^\top A'')x = \hat{\alpha}$ and thus $x \in G = \text{conv}(\mathcal{S})$ from the proof of Theorem 4 above. □

Hence, the convex hull of the decomposition is a subset of $\text{conv}(\mathcal{S})$ that contains $\hat{x}_{DW}$ and can be thought of as a surrogate for the face of optimal solutions to the cutting plane LP. Combining this corollary with Theorem 2, we conclude that separation of $\mathcal{S}$ from $\mathcal{P}$ is a sufficient condition for an inequality to be improving. Although this sufficient condition is difficult to verify in practice, it does provide additional motivation for the method described in Figure 3.

The convex hull of $\mathcal{S}$ is typically a proper face of $\mathcal{P}'$. It is possible, however, for $\hat{x}_{DW}$ to be an inner point of $\mathcal{P}'$.

Theorem 5. *If $\hat{x}_{DW}$ is an inner point of $\mathcal{P}'$, then $conv(\mathcal{S}) = \mathcal{P}'$.*

Proof. We prove the contrapositive. Suppose $\text{conv}(\mathcal{S})$ is a proper face of $\mathcal{P}'$. Then there exists a facet-defining valid inequality $(a, \beta) \in \mathbb{R}^{n+1}$ such that $\text{conv}(\mathcal{S}) \subseteq \{x \in \mathbb{R}^n \mid ax = \beta\}$. By Theorem 4, $\hat{x}_{DW} \in \text{conv}(\mathcal{S})$ and $\hat{x}_{DW}$ therefore cannot satisfy the definition of an inner point. □

In this case, illustrated graphically in Figure 4(a), $z_{DW} = z_{LP}$ and Dantzig-Wolfe decomposition does not improve the bound. All columns of the Dantzig-Wolfe LP have reduced cost zero and any member of $\mathcal{F}'$ can be made a member of the decomposition. A necessary condition for an optimal fractional solution to be an inner point of $\mathcal{P}'$ is that the value of the dual variable corresponding to the convexity constraint (i.e., α in (6)) in an optimal dual solution to the Dantzig-Wolfe LP be zero. This condition indicates that the chosen relaxation may be too weak.

As we discuss further in Section 3.2, the penalty term in the objective function of the Lagrangian subproblem (4) perturbs the original objective function so that the face of $\mathcal{P}'$ it induces is $\text{conv}(\mathcal{S})$. A second case of potential interest is when the face F of Corollary 3 is equal to $\text{conv}(\mathcal{S}) \cap \mathcal{Q}''$, illustrated graphically in Figure 4(b). This occurs when the face of $\mathcal{P}'$ induced by the original objective function vector c is equal to $\text{conv}(\mathcal{S})$ and hence the penalty term is zero. This condition can be detected by examining the objective function values of the members of the decomposition. If they are all identical, any member of the decomposition that is contained in $\mathcal{Q}''$ (if one exists) must be optimal for the original ILP, since it is feasible and has objective function value equal to z_{LP}. In this case, all constraints of the Dantzig-Wolfe LP *other than* the convexity constraint

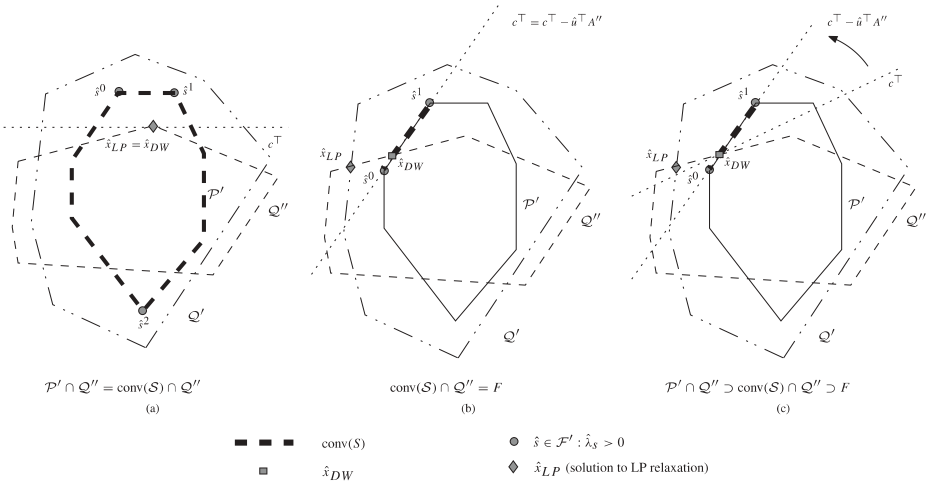

Fig. 4. The relationship of $\mathcal{P}' \cap \mathcal{Q}''$, conv($\mathcal{S}$), and the face F, for different cost vectors

must have dual value zero, since removing them does not change the optimal solution value. The more typical case, in which F is a proper subset of conv($\mathcal{S}$) $\cap\, \mathcal{Q}''$ and the penalty term is nonzero, is shown in Figure 4(c).

3.2. Relax and Cut

Finding Improving Inequalities. When the bounding subproblem is the Lagrangian dual, it is more difficult to obtain the primal solution information readily available to us in both the cutting plane method and price and cut. The amount of primal solution information depends on the algorithm used to solve the Lagrangian dual. With methods such as the volume algorithm [9], it is possible to obtain an approximate primal solution. For the sake of discussion, however, we assume in what follows that no primal solution information is available. In such a case, we can attempt to separate $\hat{s} = \text{argmin}\ z_{LR}(\hat{u})$ from $\mathcal{P}$, where $\hat{u}$ is the vector of optimal dual multipliers defined earlier. Since $\hat{s}$ is a member of $\mathcal{F}'$, we are again taking advantage of our ability to separate members of $\mathcal{F}'$ from $\mathcal{P}$ effectively. If successful, we immediately "dualize" this new constraint by adding it to $[A'', b'']$, as described in Section 3. This has the effect of introducing a new dual multiplier and slightly perturbing the Lagrangian objective function.

As with both the cutting plane and price and cut methods, the difficulty with relax and cut is that the valid inequalities generated by separating $\hat{s}$ from $\mathcal{P}$ may not be improving, as Guignard first observed in [31]. Furthermore, we cannot verify the condition of Corollary 1, which is the best available necessary condition for an inequality to be improving. To deepen our understanding of the potential effectiveness of the valid inequalities generated during relax and cut, we further examine the relationship between $\hat{s}$ and $\hat{x}_{DW}$.

Connections to Other Methods. By considering again the reformulation of the Lagrangian dual as the linear program (6), we observe that each constraint binding at an optimal solution corresponds to an alternative optimal solution to the Lagrangian subproblem with multipliers $\hat{u}$. The binding constraints of (6) correspond to variables with reduced cost zero in the Dantzig-Wolfe LP (7), so it follows immediately that the set $\mathcal{S}$ from (10) is also the set of all alternative optimal solutions to $z_{LR}(\hat{u})$.

Because $\hat{x}_{DW}$ is both an optimal solution to an LP solved over $\mathcal{P}' \cap \mathcal{Q}''$ with objective function c and is contained in conv$(\mathcal{S})$, it also follows that

$$c^\top \hat{x}_{DW} = c^\top \hat{x}_{DW} + \hat{u}^\top (b'' - A''\hat{x}_{DW}) = (c^\top - \hat{u}^\top A'')\hat{x}_{DW} + \hat{u}^\top b''.$$

In other words, the penalty term in the objective function of the Lagrangian subproblem (4) serves to rotate the original objective function so that it becomes parallel to the face conv$(\mathcal{S})$, while the constant term $\hat{u}^\top b''$ ensures that $\hat{x}_{DW}$ has the same cost with both the original and the Lagrangian objective function. This is illustrated in Figure 4(c).

One conclusion that can be drawn from these observations is that solving the Dantzig-Wolfe LP produces a *set* of alternative optimal solutions to the Lagrangian subproblem with multipliers $\hat{u}$, at least one of which must be violated by a given improving inequality. This yields a verifiable necessary condition for a generated inequality to be improving. Relax and cut, in its most straightforward incarnation, produces one member of this set. In this case, even if improving inequalities exist, it is possible that none of them are violated by the member of $\mathcal{S}$ so produced, especially if it has a small weight in the optimal DW decomposition $\hat{\lambda}$. Note, however, that by keeping track of the solutions to the Lagrangian subproblem that are produced while solving the Lagrangian dual, one can approximate the optimal DW decomposition. This is the approach taken by the volume algorithm [9] and other subgradient-based methods. As in price and cut, when $\hat{x}_{DW}$ is an inner point of $\mathcal{P}'$, the decomposition does not improve the bound and all members of $\mathcal{F}'$ are alternative optimal solutions to the Lagrangian subproblem. This situation is depicted in Figure 4(a). In this case, separating an optimal solution to $z_{LR}(\hat{u})$ from $\mathcal{P}$ is unlikely to yield an improving inequality.

3.3. Decompose and Cut

The use of an optimal DW decomposition to aid in separation is easy to extend to a traditional branch and cut framework using a technique we call *decompose and cut*, originally proposed in [55] and further developed in [36] and [57]. Consider the optimal fractional solution $\hat{x}_{CP}$ obtained directly by solving the cutting plane LP and suppose that given $s \in \mathcal{F}'$, we can determine effectively whether $s \in \mathcal{F}$ and if not, generate a valid inequality (a, β) violated by s. By first *decomposing* $\hat{x}_{CP}$ (i.e., expressing $\hat{x}_{CP}$ as a convex combination of members of $\mathcal{F}'$) and then separating each member of this decomposition from $\mathcal{P}$ in a fashion analogous to that described in Figure 3, we may be able to separate $\hat{x}_{CP}$ from the polyhedron $\mathcal{P}$.

The difficult step is finding the decomposition of $\hat{x}_{CP}$. This can be accomplished by solving a linear program whose columns are the members of $\mathcal{F}'$, as described in Figure 5. This linear program is reminiscent of a Dantzig-Wolfe LP and in fact can be solved using an analogous column-generation scheme, as described in Figure 6. This scheme can be seen as inverting the method described in Section 3.1, since it begins with the fractional solution $\hat{x}_{CP}$ and tries to compute a decomposition, instead of the other

Separation in Decompose and Cut
Input: $\hat{x} \in \mathbb{R}^n$
Output: A valid inequality for $\mathcal{P}$ violated by $\hat{x}$, if one is found.

1. Apply standard separation techniques to separate $\hat{x}$. If one of these returns a violated inequality, then STOP and output the violated inequality.
2. Otherwise, solve the linear program

$$\max_{\lambda \in \mathbb{R}_+^{\mathcal{F}'}} \{\mathbf{0}^\top \lambda : \sum_{s \in \mathcal{F}'} s\lambda_s = \hat{x}, \sum_{s \in \mathcal{F}'} \lambda_s = 1\}, \tag{13}$$

   as in Figure 6.
3. The result of Step 2 is either (1) a valid inequality (a, β) for $\mathcal{P}$ that is violated by $\hat{x}$, or (2) a subset $\mathcal{D}$ of members of $\mathcal{F}'$ participating in a convex combination of $\hat{x}$. In the first case, go to Step 4. In the second case, STOP and output the violated inequality.
4. Attempt to separate each member of $\mathcal{D}$ from $\mathcal{P}$. For each inequality violated by a member of $\mathcal{D}$, check whether it is also violated by $\hat{x}$. If an inequality violated by $\hat{x}$ is encountered, STOP and output it.

Fig. 5. Separation in the decompose and cut method

Column Generation in Decompose and Cut
Input: $\hat{x} \in \mathbb{R}^n$
Output: Either (1) a valid inequality for $\mathcal{P}$ violated by $\hat{x}$; or (2) a subset $\mathcal{D}$ of members of $\mathcal{F}'$ participating in a convex combination of $\hat{x}$.

2.0 Generate an initial subset $\mathcal{G}$ of $\mathcal{F}'$.
2.1 Solve (13), replacing $\mathcal{F}'$ by $\mathcal{G}$. If this linear program is feasible, then the elements of $\mathcal{F}'$ corresponding to the nonzero components of $\hat{\lambda}$, the current solution, comprise the set $\mathcal{D}$, so STOP.
2.2 Otherwise, let (a, β) be a valid inequality for conv$(\mathcal{G})$ violated by $\hat{x}$. Solve the optimization problem over $\mathcal{P}'$ with cost vector a and let s be the resulting solution. If the optimal value is less than β, then add s to $\mathcal{G}$ and go to 2.1. Otherwise, (a, β) is an inequality valid for $\mathcal{P}' \supseteq \mathcal{P}$ and violated by $\hat{x}$, so STOP.

Fig. 6. Column generation for the decompose and cut method

way around. By the equivalence of optimization and separation, we conclude that the worst-case complexity of finding a decomposition of $\hat{x}_{CP}$ is polynomially equivalent to that of optimizing over $\mathcal{P}'$.

Once the decomposition is found, it can be used as in price and cut to locate a violated inequality by the methodology discussed earlier. This procedure is shown in Figure 5. In contrast to price and cut, however, it is possible that $\hat{x}_{CP} \notin \mathcal{P}'$. This could occur, for instance, if exact separation methods for $\mathcal{P}'$ are too expensive to apply consistently. In this case, it is obviously not possible to find a decomposition in Step 2. The proof of infeasibility for the linear program (13), however, provides an inequality separating $\hat{x}_{CP}$ from $\mathcal{P}'$ at no additional expense. Hence, even if we fail to find a decomposition, we still find an inequality valid for $\mathcal{P}$ and violated by $\hat{x}_{CP}$. This idea was originally suggested in [55] and was further developed in [36]. A similar concept was also discovered and developed independently by Applegate et al. [3].

Applying decompose and cut in every iteration as the sole means of separation is polynomially equivalent to price and cut. In practice, however, the decomposition is only computed when needed, i.e., when less expensive separation heuristics fail to separate the optimal fractional solution. This could give decompose and cut an edge in terms of computational efficiency. In other respects, the computations performed in each method are similar.

3.4. Implementation and Extensions

In practice, there are many variations on the general theme described here. The details surrounding implementation of these methods will be covered in a separate paper, but we would like to give the reader a taste for the issues involved and for the existing methodology. An important aspect of the implementation of these methods is the algorithm used for solving the subproblem in Step 1 of the algorithm in Figure 2. Three general categories of methods for solving such subproblems are simplex methods, interior point methods, and subgradient methods. Simplex methods provide accurate primal solution information, but updates to the dual solution each iteration are relatively expensive. In their most straightforward form, they also tend to converge slowly when used to solve a Dantzig-Wolfe LP by column generation because of the fact that they produce basic dual solutions, which can change substantially from one iteration to the next. This problem can be addressed by implementing one of a number of stabilization methods that prevent the dual solution from changing "too much" from one iteration to the next (for a survey, see [39]). In their most straightforward form, subgradient methods do not produce primal solution information. However, it is possible to extract approximate primal solutions from variants of subgradient such as the volume algorithm [9]. Subgradient methods also have convergence issues without some form of stabilization. A recent class of algorithms that has proven effective in this regard is bundle methods [18]. Interior point methods may provide a middle ground by providing accurate primal solution information and more stable dual solutions [58, 28]. In addition, hybrid methods that alternate between simplex and subgradient methods for updating the dual solution have also shown promise [10, 33].

An even more general framework containing the methods described here can be obtained by viewing them as alternating between a *master* problem that updates solution information and a procedure for using that solution information to generate an improved approximation of $\mathcal{P}$ by solving either a *pricing* or a *cutting* subproblem. In this generalized framework, we do not insist on solving the subproblem in Step 1 of the algorithm in Figure 2 to optimality before generating cuts, but rather allow the method to alternate freely between the pricing and cutting subproblems [17, 29, 42]. The software framework we are developing will allow essentially any sequence of solution updates, pricing, and cutting by any of the methods discussed here. This leads to a wide range of possibilities, very few of which have been investigated in the literature so far. For a treatment of this more general viewpoint, see [56].

4. Applications

In this section, we illustrate the concepts presented so far with three examples. For each example, we discuss the key elements needed to apply the framework: (1) the original ILP formulation, (2) the explicit and implicit polyhedra, and (3) known classes of valid inequalities that can be used to dynamically tighten the explicit polyhedron by using structured separation techniques. The well-known *template paradigm* for separation, so named by Applegate et al. [3], is the standard approach for generating violated valid inequalities when solving MILPs. This paradigm operates on the precept that it is sometimes possible to effectively solve the separation problem for a given class of inequalities valid for the polyhedron $\mathcal{P}$, though the general separation problem for $\mathcal{P}$ is difficult. Our framework extends this paradigm by considering classes of valid inequalities for which the separation of an arbitrary real vector is difficult but for which separation of solutions to a specified relaxation can be accomplished effectively. In addition to the three examples presented here, there are a number of common ILPs with classes of valid inequalities and relaxations that fit into this framework, such as the Generalized Assignment Problem [53], the Edge-Weighted Clique Problem [34], the Traveling Salesman Problem [5], the Knapsack Constrained Circuit Problem [38], the Rectangular Partition Problem [16], the Linear Ordering Problem [15], and the Capacitated Minimum Spanning Tree Problem [24].

4.1. Vehicle Routing Problem

We first consider the *Vehicle Routing Problem* (VRP) introduced by Dantzig and Ramser [20]. In this $\mathcal{NP}$-hard optimization problem, a fleet of k vehicles with uniform capacity C must service known customer demands for a single commodity from a common depot at minimum cost. Let $V = \{1, \ldots, |V|\}$ index the set of customers and let the depot have index 0. Associated with each customer $i \in V$ is a demand d_i. The cost of travel from location i to location j is denoted by c_{ij} for $i, j \in V \cup \{0\}$. We assume that $c_{ij} = c_{ji} > 0$ if $i \neq j$ and $c_{ii} = 0$.

By constructing an associated complete undirected graph G with vertex set $N = V \cup \{0\}$ and edge set $E = N \times N$, we can formulate the VRP as an integer program. A *route* is an ordered subset $R = (i_1, i_2, \ldots, i_m)$ of V with associated edge set $E_R = \{\{i_j, i_{j+1}\} : j \in 0, \ldots, m\}$, where $i_0 = i_{m+1} = 0$. A feasible solution is then any subset of E that is the union of the edge sets of k disjoint routes R_i, $i \in [1..k]$, each satisfying the capacity restriction, i.e., $\sum_{j \in R_i} d_j \leq C \; \forall i \in [1..k]$. Each route corresponds to a set of customers serviced by one of the k vehicles. To simplify the presentation, we define some additional notation. Let $\delta(S) = \{\{i, j\} \in E \mid i \in S, j \notin S\}$, $E(S : T) = \{\{i, j\} \mid i \in S, j \in T\}$, $E(S) = E(S : S)$ and $x(F) = \sum_{e \in F} x_e$.

By associating a variable with each edge in the graph, we obtain the following formulation of this ILP [37]:

$$\min \sum_{e \in E} c_e x_e$$

$$x(\delta(\{0\})) = 2k \tag{14}$$

$$x(\delta(\{v\})) = 2 \qquad \forall v \in V, \tag{15}$$

$$x(\delta(S)) \geq 2b(S) \qquad \forall S \subseteq V, \; |S| > 1, \tag{16}$$

$$x_e \in \{0, 1\} \qquad \forall e \in E(V), \tag{17}$$

$$x_e \in \{0, 1, 2\} \qquad \forall e \in \delta(0). \tag{18}$$

Here, $b(S)$ represents a lower bound on the number of vehicles required to service the set of customers S. Inequalities (14) ensure that there are exactly k vehicles, each departing from and returning to the depot, while inequalities (15) require that each customer must be serviced by exactly one vehicle. Inequalities (16), known as the *generalized subtour elimination constraints* (GSECs) can be viewed as a generalization of the subtour elimination constraints from the *Traveling Salesman Problem* (TSP) and enforce connectivity of the solution, as well as ensuring that no route has total demand exceeding capacity C. For ease of computation, we can define $b(S) = \left\lceil \left(\sum_{i \in S} d_i\right)/C \right\rceil$, a trivial lower bound on the number of vehicles required to service the set of customers S.

Returning to our earlier notation and setup, the set of feasible solutions to the VRP is

$$\mathcal{F} = \left\{ x \in \mathbb{R}^E \mid x \text{ satisfies } (14) - (18) \right\},$$

and $\mathcal{P} = \text{conv}(\mathcal{F})$ is then the VRP polytope. Many classes of valid inequalities for the VRP polytope have been reported in the literature (see [49] for a survey). Significant effort has been devoted to developing efficient algorithms for separating an arbitrary fractional point using these classes of inequalities (see [43] for recent results).

For dynamic tightening of the explicit polyhedron, we concentrate here on the separation of GSECs. The separation problem for GSECs is $\mathcal{NP}$-complete (see [4]), even when $b(S)$ is taken to be $\left\lceil \left(\sum_{i \in S} d_i\right)/C \right\rceil$, as above. In [43], Lysgaard et al. review standard heuristic procedures for separating arbitrary fractional solutions from the GSEC polyhedron (the polyhedron described by all GSECs). Although GSECs are part of the formulation presented above, there are exponentially many of them, so they must either be generated dynamically or included as part of the description of the implicit polyhedron. We discuss three alternatives for the implicit polyhedron: the *Perfect b-Matching* polytope, the *Degree-constrained k-Tree* polytope, and the *Multiple Traveling Salesman* polytope. For each of these alternatives, solutions to the relaxation can be easily separated from the GSEC polyhedron.

Perfect b-Matching Problem. With respect to the graph G, the *Perfect b-Matching Problem* is to find a minimum weight subgraph of G such that $x(\delta(v)) = b_v \ \forall v \in V$ for some $b \in \mathbb{Z}_+^N$. By dropping the GSECs from the VRP formulation, we obtain an instance of the Perfect b-Matching Problem with associated implicit polyhedron $\mathcal{P}' = \text{conv}(\mathcal{F}')$, where

$$\mathcal{F}' = \left\{ x \in \mathbb{R}^E \mid x \text{ satisfies } (14), (15), (17), (18) \right\}.$$

In [48], Müller-Hannemann and Schwartz present several efficient polynomial algorithms for solving the Perfect b-Matching Problem. Note that in this case, the explicit polyhedron needed to completely describe the VRP polyhedron includes the GSECs (16). In practice, however, we start with the explicit polyhedron $\mathcal{Q}''$ comprised of a small set of GSECs and generate the others dynamically, as described earlier.

In [47], Miller uses the b-matching relaxation to solve the VRP by branch, relax, and cut. He suggests separating members of $\mathcal{F}'$ from the GSEC polytope as follows. Consider a member s of $\mathcal{F}'$ and its support graph G_s (a b-matching). If G_s is disconnected, then each component immediately induces a violated GSEC. On the other hand, if G_s is connected, we first remove the edges incident with the depot vertex and find the connected components, which comprise the routes described earlier. To identify a violated GSEC, we compute the total demand of each route, checking whether it exceeds capacity. If not, the solution is feasible for the original ILP and does not violate any GSECs. If so, the set S of customers on any route whose total demand exceeds capacity induces a violated GSEC. This separation routine runs in $O(|V|)$ time and can be used in any of the integrated decomposition methods previously described. Figure 7 gives an optimal fractional solution (a) to an LP relaxation of the VRP expressed as a convex combination of two b-matchings (b) and (c). In this example, the capacity $C = 35$ and by inspection we find a violated GSEC in the second b-matching (c) with S equal to the indicated component. This inequality is also violated by the optimal fractional solution, since $\hat{x}(\delta(S)) = 3.0 < 4.0 = 2b(S)$.

Minimum Degree-constrained k-Tree Problem. A *k-tree* is defined as a spanning subgraph of G that has $|V|+k$ edges (recall that G has $|V|+1$ vertices). A *degree-constrained k-tree* (k-DCT), as defined by Fisher in [23], is a k-tree with degree $2k$ at node 0. The *Minimum k-DCT Problem* is that of finding a minimum cost k-DCT, where the cost of a k-DCT is the sum of the costs on the edges present in the k-DCT. Fisher [23] introduced this relaxation of the VRP as part of a Lagrangian relaxation-based algorithm for solving the VRP.

The k-DCT polyhedron is obtained by first adding the redundant constraint

$$x(E) = |V| + k, \tag{19}$$

which holds for any $x \in \mathcal{P}$, then deleting the degree constraints (15), and relaxing the capacity to $C = \sum_{i \in S} d_i$. Relaxing the capacity gives $b(S) = 1$ for all $S \subseteq V$, and effectively replaces (16) with

$$\sum_{e \in \delta(S)} x_e \geq 2, \forall S \subseteq V, \ |S| > 1. \tag{20}$$

The implicit polyhedron is then defined to be $\mathcal{P}' = \mathrm{conv}(\mathcal{F}')$, where

$$\mathcal{F}' = \left\{ x \in \mathbb{R}^E \mid x \text{ satisfies (14), (17), (19), (20)} \right\}.$$

The explicit polyhedron is then initially described by the constraints (15). In [63], Wei and Yu give an algorithm for solving the Minimum k-DCT Problem with running time $O(|V|^2 \log |V|)$. In [46], Martinhon et al. study the use of the k-DCT relaxation for the VRP in the context of branch, relax and cut. Again, consider separating a member s of $\mathcal{F}'$ from the polyhedron defined by all GSECs. It is easy to see that for GSECs, an algorithm identical to that described above can be applied. Figure 7 gives the optimal fractional solution (a) expressed as a convex combination of four k-DCTs (d)–(g). Removing the depot edges, and checking each component's demand, we easily identify the violated GSEC indicated in (g).

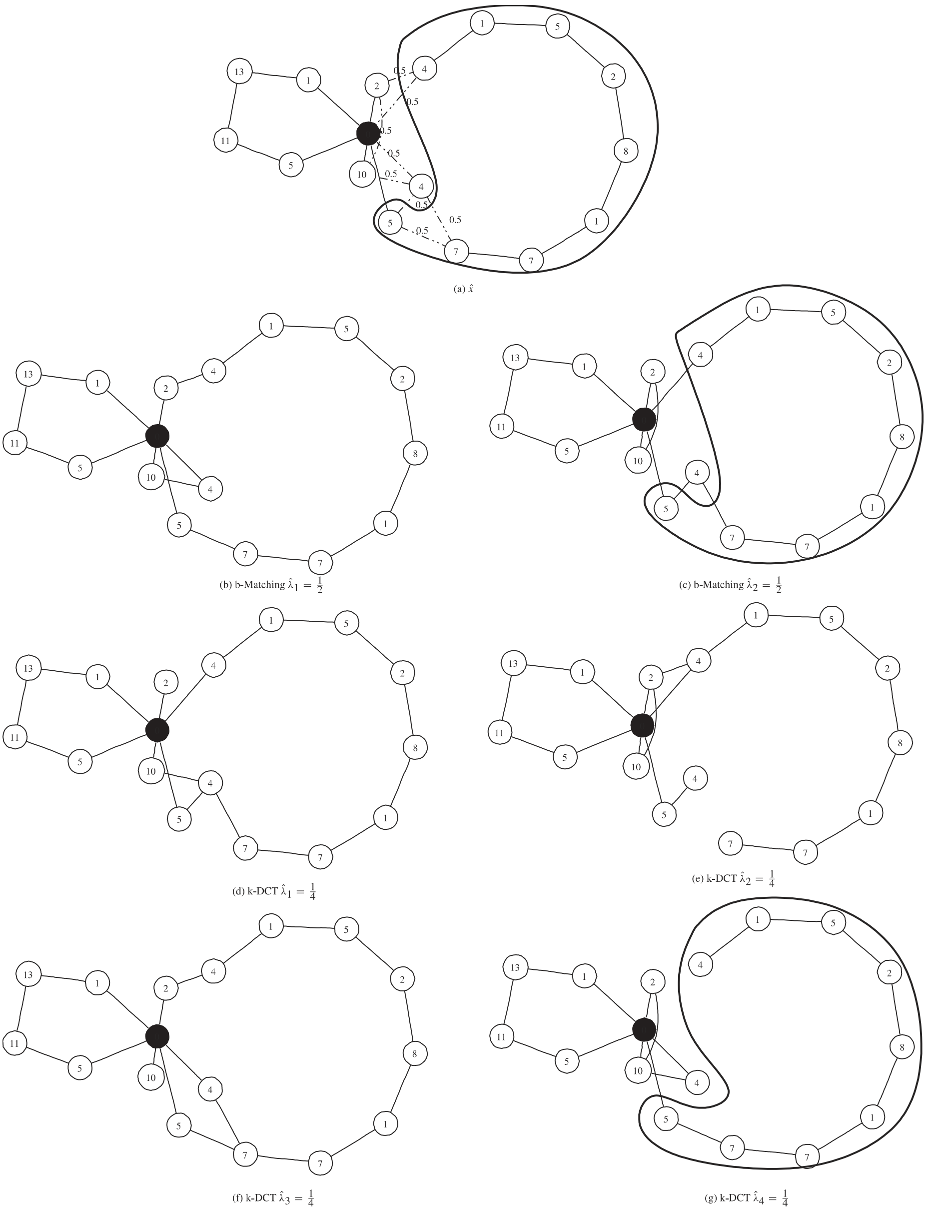

Fig. 7. Example of a decomposition into b-Matchings and k-DCTs

Multiple Traveling Salesman Problem. The *Multiple Traveling Salesman Problem* (k-TSP) is an uncapacitated version of the VRP obtained by adding the degree constraints to the k-DCT polyhedron. The implicit polyhedron is then defined as $\mathcal{P}' = \text{conv}(\mathcal{F}')$, where

$$\mathcal{F}' = \left\{ x \in \mathbb{R}^E \mid x \text{ satisfies (14), (15), (17), (18), (20)} \right\}.$$

Although the k-TSP is an $\mathcal{NP}$-hard optimization problem, small instances can be solved effectively by transformation into an equivalent TSP obtained by adjoining to the graph $k-1$ additional copies of vertex 0 and its incident edges. In this case, we again start with the explicit polyhedron comprised of a small set of GSECs and generate the others dynamically. In [57], Ralphs et al. report on an implementation of branch, decompose and cut using the k-TSP as a relaxation.

4.2. Three-Index Assignment Problem

The *Three-Index Assignment Problem* (3AP) is that of finding a minimum-weight partition of the vertices of a complete tri-partite graph $K_{n,n,n}$ into cliques of size three. Let I, J and K be the three vertex sets defining the tri-partite graph, with $|I| = |J| = |K| = n$, and let $H = I \times J \times K$ be the set of all cliques of size three. By associating a variable with each member of H, 3AP can be formulated as the following binary integer program:

$$\min \sum_{(i,j,k)\in H} c_{ijk}x_{ijk}$$

$$\sum_{(j,k)\in J\times K} x_{ijk} = 1 \quad \forall i \in I \tag{21}$$

$$\sum_{(i,k)\in I\times K} x_{ijk} = 1 \quad \forall j \in J \tag{22}$$

$$\sum_{(i,j)\in I\times J} x_{ijk} = 1 \quad \forall k \in K \tag{23}$$

$$x_{ijk} \in \{0,1\} \qquad \forall (i,j,k) \in H \tag{24}$$

A number of applications of 3AP, which is known to be $\mathcal{NP}$-hard [25], can be found in the literature (see Piersjalla [18,19]). The set of feasible solutions to 3AP is

$$\mathcal{F} = \left\{ x \in \mathbb{R}^H \mid x \text{ satisfies } (21) - (24) \right\}$$

and $\mathcal{P} = \text{conv}(\mathcal{F})$ is then the 3AP polytope.

In [7], Balas and Saltzman study the polyhedral structure of the 3AP polytope and introduce several classes of facet-inducing inequalities. Let $u, v \in H$ and define $|u \cap v|$ to be the number of coordinates for which the vectors u and v have the same value. Let $C(u) = \{w \in H \mid |u \cap w| = 2\}$ and $C(u,v) = \{w \in H \mid |u \cap w| = 1, |w \cap v| = 2\}$. We consider two classes of facet-inducing inequalities for $\mathcal{P}$. For $u \in H$ and any $x \in \mathcal{P}$, we have

$$x_u + \sum_{w\in C(u)} x_w \leq 1, \tag{25}$$

yielding a class of inequalities whose members are referred to by the label $Q_1(u)$, as in [7]. Similarly, for $u, v \in H$ with $|u \cap v = 0|$ and any $x \in \mathcal{P}$, we have

$$x_u + \sum_{w\in C(u,v)} x_w \leq 1, \tag{26}$$

yielding a class of inequalities whose members are referred to by the label $P_1(u, v)$, also as in [7]. In [6], Balas and Qi describe algorithms that solve the separation problem for the polyhedra defined by the inequalities in these two classes in $O(n^3)$ time.

Balas and Saltzman consider the use of the classical Assignment Problem (AP) as a relaxation of 3AP in an early implementation of branch, relax, and cut [8]. Following their lead, we define the implicit polyhedron to be $\mathcal{P}' = \text{conv}(\mathcal{F}')$, where

$$\mathcal{F}' = \left\{ x \in \mathbb{R}^H \mid x \text{ satisfies } (22) - (24) \right\}.$$

The explicit polyhedron is then initially described by the constraints (21). The AP can be solved in $O(n^{5/2} \log(nC))$ time, where $C = \max_{w \in H} c_w$, by the cost-scaling algorithm [2]. Consider separating a member s of $\mathcal{F}'$ from the polyhedron defined by $Q_1(u)$ for all $u \in H$. Let $L(s)$ be the set of n triplets corresponding to the nonzero components of s (the assignment from J to K). It is easy to see that if there exist $u, v \in L(s)$ such that $u = (i_0, j_0, k_0)$ and $v = (i_0, j_1, k_1)$, i.e., the assignment *overcovers* the set I, then both $Q_1(i_0, j_0, k_1)$ and $Q_1(i_0, j_1, k_0)$ are violated by s. Figure 8 shows the decomposition of an optimal fractional solution $\hat{x}$ (a) into a convex combination of assignments (b-d). The pair of triplets (0, 3, 1) and (0, 0, 3) satisfies the condition just discussed and identifies two violated valid inequalities, $Q_1(0, 3, 3)$ and $Q_1(0, 0, 1)$ that are violated by the second assignment, shown in (c). The latter is also violated by $\hat{x}$ and is illustrated in (e). This separation routine runs in $O(n)$ time.

Now consider separating a member s of $\mathcal{F}'$ from the polyhedron defined by $P_1(u, v)$ for all $u, v \in H$. As above, for any pair of assignments that correspond to nonzero components of s and have the form $(i_0, j_0, k_0), (i_0, j_1, k_1)$, we know s violates

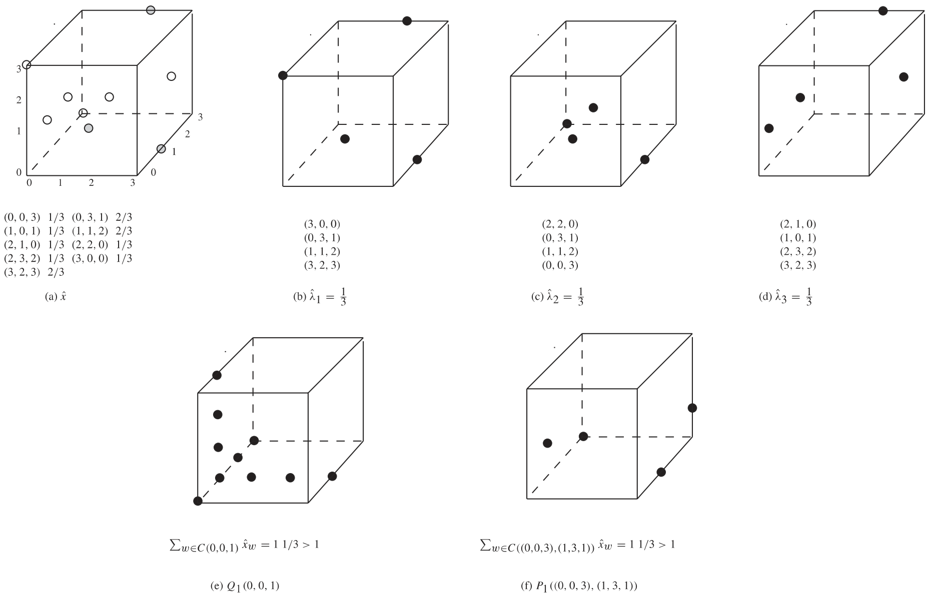

Fig. 8. Example of a decomposition into assignments

$P_1((i_0, j_0, k_0), (i, j_1, k_1)), \forall i \neq i_0$ and $P_1((i_0, j_1, k_1), (i, j_0, k_0)), \forall i \neq i_0$. In Figure 8, the second assignment (c) violates $P_1((0, 0, 3), (1, 3, 1))$. This inequality is also violated by $\hat{x}$ and is illustrated in (f). Once again, this separation routine runs in $O(n)$ time.

4.3. Steiner Tree Problem

Let $G = (V, E)$ be a complete undirected graph with vertex set $V = \{1, ..., |V|\}$, edge set E, and a positive weight c_e associated with each edge $e \in E$. Let $T \subseteq V$ define the set of *terminals*. The *Steiner Tree Problem* (STP), which is $\mathcal{NP}$-hard, is that of finding a subgraph that spans T (called a *Steiner tree*) and has minimum edge cost. In [13], Beasley formulated the STP as a side constrained *Minimum Spanning Tree Problem* (MSTP) as follows. Let $r \in T$ be a given terminal and consider an artificial vertex 0. Construct the augmented graph $\bar{G} = (\bar{V}, \bar{E})$ where $\bar{V} = V \cup \{0\}$ and $\bar{E} = E \cup \{\{i, 0\} \mid i \in (V \setminus T) \cup \{r\}\}$. Let $c_{i0} = 0$ for all $i \in (V \setminus T) \cup \{r\}$. The STP is then equivalent to finding a minimum spanning tree (MST) in $\bar{G}$ subject to the additional restriction that any vertex $i \in (V \setminus T)$ incident to an edge $\{i, 0\} \in \bar{E}$ must have degree one.

By associating a binary variable x_e with each edge $e \in \bar{E}$, indicating whether or not the edge is selected, we can formulate the STP as the following integer program:

$$\min \sum_{e \in E} c_e x_e$$

$$x(\bar{E}) = |\bar{V}| - 1 \tag{27}$$

$$x(E(S)) \leq |S| - 1 \quad \forall S \subseteq \bar{V} \tag{28}$$

$$x_{i0} + x_e \leq 1 \quad \forall e \in \delta(i), i \in (V \setminus T) \tag{29}$$

$$x_e \in \{0, 1\} \quad \forall e \in \bar{E} \tag{30}$$

Inequalities (27) and (28) ensure that the solution forms a spanning tree on $\bar{G}$. Inequalities (28) are subtour elimination constraints (similar to those used in the TSP). Inequalities (29) are the side constraints that ensure the solution can be converted to a Steiner tree by dropping the edges in $\bar{E} \setminus E$.

The members of

$$\mathcal{F} = \left\{ x \in \mathbb{R}^{\bar{E}} \mid x \text{ satisfies } (27) - (30) \right\}$$

then correspond to feasible solutions of the STP and we call $\mathcal{P} = \text{conv}(\mathcal{F})$ the STP polytope. We consider two classes of valid inequalities for the STP polytope that are lifted versions of the subtour elimination constraints (SECs). For any $x \in \mathcal{P}$, we have

$$x(E(S)) + x(E(S \setminus T : \{0\})) \leq |S| - 1 \; \forall S \subseteq V, S \cap T \neq \emptyset, \text{and} \tag{31}$$

$$x(E(S)) + x(E(S \setminus \{v\} : \{0\})) \leq |S| - 1 \; \forall S \subseteq V, S \cap T = \emptyset, v \in S \tag{32}$$

The class of valid inequalities (31) were independently introduced by Goemans [27], Lucena [40], and Margot, Prodon, and Liebling [44], for another extended formulation of STP. The inequalities (32) were introduced in [27, 44]. The separation problem for

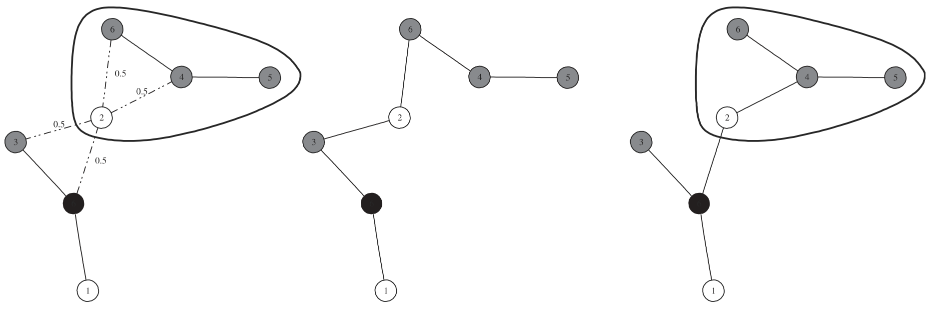

Fig. 9. Example of a decomposition into minimum spanning trees

inequalities (31) and (32) can be solved in $O(|V|^4)$ time through a series of max-flow computations.

In [41], Lucena considers the use of MSTP as a relaxation of STP in the context of a branch, relax, and cut algorithm. Inequalities (29) describe the explicit polyhderon, while the implicit polyhedron is defined to be $\mathcal{P}' = \text{conv}(\mathcal{F}')$, where

$$\mathcal{F}' = \left\{ x \in \mathbb{R}^{\bar{E}} \mid x \text{ satisfies (27), (28), (30)} \right\}.$$

The MSTP can be solved in $O(|E| \log |V|)$ time using Prim's algorithm [54]. Consider the separation of a member of $s \in \mathcal{F}'$ from the polyhedron defined by the lifted subtour inequalities (31) and (32). In order to identify a violated inequality of the form (31) or (32) we remove the artificial vertex 0 and find the connected components on the resulting subgraph. Any component of size greater than 1 that does not contain r and does contain a terminal defines a violated SEC (31). In addition, if the component does not contain any terminals, then each vertex in the component that was not connected to the artificial vertex defines a violated SEC (32).

Figure 9 shows an optimal fractional solution (a) to an LP relaxation of the STP expressed as a convex combination of two spanning trees (b) and (c). In this figure, the artificial vertex is black, the terminals are gray and $r = 3$. By removing the artificial vertex, we easily find a violated SEC by considering the second spanning tree (c) with S equal to the marked component. This inequality is also violated by the optimal fractional solution, since $\hat{x}(E(S)) + \hat{x}(E(S \setminus T \ : \ \{0\})) = 3.5 > 3 = |S| - 1$. It should also be noted that the first spanning tree (b), in this case, is in fact feasible for the original problem.

5. Conclusions and Future Work

In this paper, we presented a framework for integrating dynamic cut generation (outer methods) with traditional decomposition methods (inner methods). We have also discussed a paradigm for the generation of improving inequalities based on decomposition and the separation of solutions to a relaxation, a problem that is often much easier than that of separating arbitrary real vectors. Viewing the cutting plane method, Lagrangian relaxation, and Dantzig-Wolfe decomposition in a common algorithmic framework can yield new insight into all three methods. The next step in this research is to complete a computational study that will aid practitioners in making more informed choices between the many possible variants we have discussed. As part of this study, we are implementing a generic framework that will allow users to test these methods simply by providing a relaxation, a solver for that relaxation, and separation routines for solutions to the relaxation. Such a framework will enable access to a wide range of alternatives for computing bounds using decomposition and cut generation.

References

1. Aardal, K., van Hoesel, S.: Polyhedral techniques in combinatorial optimization. Statistica Neerlandica **50**, 3–26 (1996)
2. Ahuja, R., Magnanti, T., Orlin, J.: Network Flows: Theory, Algorithms, and Applications. Prentice Hall, Englewood Cliffs, NJ, 1993
3. Applegate, D., Bixby, R., Chvátal, V., Cook, W.: TSP cuts which do not conform to the template paradigm. In: Computational Combinatorial Optimization, Springer, 2001 pp. 261–303
4. Augerat, P., Belenguer, J.M., Benavent, E., Corberán, A., Naddef, D., Rinaldi, G.: Computational Results with a Branch and Cut Code for the Capacitated Vehicle Routing Problem. Technical Report 949-M, Université Joseph Fourier, Grenoble, France, 1995
5. Balas, E., Christofides, N.: A restricted Lagrangean approach to the traveling salesman problem. Mathematical Programming **21**, 19–46 (1981)
6. Balas, E., Qi, L.: Linear-time separation algorithms for the three-index assignment polytope. Discrete Applied Mathematics **43**, 1–12 (1993)
7. Balas, E., Saltzman, M.J.: Facets of the three-index assignment polytope. Discrete Applied Mathematics **23**, 201–229 (1989)
8. Balas, E., Saltzman, M.J.: An algorithm for the three-index assignment problem. Operations Research **39**, 150–161 (1991)
9. Barahona, F., Anbil, R.: The volume algorithm: Producing primal solutions with a subgradient method. Mathematical Programming **87**, 385–399 (2000)
10. Barahona, F., Jensen, D.: Plant location with minimum inventory. Mathematical Programming **83**, 101–111 (1998)
11. Barnhart, C., Hane, C.A., Vance, P.H.: Using branch-and-price-and-cut to solve origin-destination integer multi-commodity flow problems. Operations Research **48**, 318–326 (2000)
12. Barnhart, C., Johnson, E.L., Nemhauser, G.L., Savelsbergh, M.W.P., Vance, P.H.: Branch and price: Column generation for solving huge integer programs. Operations Research **46**, 316–329 (1998)
13. Beasley, J.E.: A SST-based algorithm for the steiner problem in graphs. Networks **19**, 1–16 (1989)
14. Beasley, J.E.: Lagrangean relaxation. In: C.R. Reeves (ed.), Modern Heuristic Techniques for Combinatorial Optimization. Wiley, 1993
15. Belloni, A., Lucena, A.: A Lagrangian heuristic for the linear ordering problem. In: M.G.C. Resende, J. Pinho de Sousa (eds.), Metaheuristics: Computer Decision-Making. Kluwer Academic, 2003, pp. 123–151
16. Calheiros, F.C., Lucena, A., de Souza, C.: Optimal rectangular partitions. Networks **41**, 51–67 (2003)
17. Caprara, A., Fischetti, M., Toth, P.: Algorithms for the set covering problem. Annals of Operations Research **98**, 353–371 (2000)
18. Carraresi, P., Frangioni, A., Nonato, M.: Applying bundle methods to optimization of polyhedral functions: An applications-oriented development. Technical Report TR-96-17, Universitá di Pisa, 1996
19. Dantzig, G., Wolfe, P.: Decomposition principle for linear programs. Operations Research **8**, 101–111 (1960)
20. Danzig, G.B., Ramser, R.H.: The truck dispatching problem. Management Science **6**, 80–91 (1959)
21. Poggi de Aragão, M., Uchoa, E.: Integer program reformulation for robust branch-and-cut-and-price. Working paper, Pontifíca Universidade Católica do Rio de Janeiro, 2004. Available from http://www.inf.puc-rio.br/~uchoa/doc/rbcp-a.pdf
22. Fisher, M.L.: The Lagrangian relaxation method for solving integer programming problems. Management Science **27**, 1–18 (1981)
23. Fisher, M.L.: Optimal solution of vehicle routing problems using minimum k-trees. Operations Research **42**, 626–642 (1994)

24. Fukasawa, R., Poggi de Aragão, M., Reis, M., Uchoa, E.: Robust branch-and-cut-and-price for the capacitated minimum spanning tree problem. In: Proceedings of the International Network Optimization Conference. Evry, France, 2003, pp. 231–236
25. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman and Company, New York, 1979
26. Geoffrion, A.: Lagrangian relaxation for integer programming. Mathematical Programming Study **2**, 82–114 (1974)
27. Goemans, M.X.: The steiner tree polytope and related polyhedra. Mathematical Programming **63**, 157–182 (1994)
28. Goffin, J.L., Vial, J.P.: Convex nondifferentiable optimization: A survey focused on the analytic center cutting plane method. Technical Report 99.02, Logilab, Geneva, Switzerland, 1999
29. Gondzio, J., Sarkissian, R.: Column generation with a primal-dual method. Technical report, University of Geneva, Logilab, HEC Geneva, 1996
30. Grötschel, M., Lovász, L., Schrijver, A.: The ellipsoid method and its consequences in combinatorial optimization. Combinatorica **1**, 169–197 (1981)
31. Guignard, M.: Efficient cuts in Lagrangean "relax-and-cut" schemes. European Journal of Operational Research **105**, 216–223 (1998)
32. Guignard, M.: Lagrangean relaxation. Top **11**, 151–228 (2003)
33. Huisman, D., Jans, R., Peeters, M., Wagelmans, A.: Combining column generation and Lagrangian relaxation. Technical report, Erasmus Research Institute of Management, Rotterdamn, The Netherlands, 2003
34. Hunting, M., Faigle, U., Kern, W.: A Lagrangian relaxation approach to the edge-weighted clique problem. European Journal of Operational Research **131**, 119–131 (2001)
35. Kohl, N., Desrosiers, J., Madsen, O.B.G., Solomon, M.M., Soumis, F.: 2-path cuts for the vehicle routing problem with time windows. Transportation Science **33**, 101–116 (1999)
36. Kopman, L.: A New Generic Separation Routine and Its Application In a Branch and Cut Algorithm for the Capacitated Vehicle Routing Problem. PhD thesis, Cornell University, May 1999
37. Laporte, G., Nobert, Y., Desrouchers, M.: Optimal routing with capacity and distance restrictions. Operations Research **33**, 1050–1073 (1985)
38. Linderoth, J., Galati, M.: Knapsack constrained circuit problem. Unpublished working paper, 2004
39. Lübbecke, M.E., Desrosiers, J.: Selected topics in column generation. Technical Report 008-2004, Technische Universität Berlin, 2004
40. Lucena, A.: Tight bounds for the steiner problem in graphs. Talk given at TIMS-XXX-SOBRAPO XXIII Joint International Meeting, Rio de Janeiro, July 1991
41. Lucena, A.: Steiner problem in graphs: Lagrangean relaxation and cutting planes. COAL Bulletin **28**, 2–8 (1992)
42. Lucena, A.: Non-delayed relax-and-cut algorithms. Working paper, Departmaneto de Administração, Universidade Federal do Rio de Janeiro, 2004
43. Lysgaard, J., Letchford, A.N., Eglese, R.W.: A new branch-and-cut algorithm for the capacitated vehicle routing problem. Mathematical Programming **100**, 423–445 (2004)
44. Margot, F., Prodon, A., Liebling, T.M.: Tree polyhedron on 2-tree. Mathematical Programming **63**, 183–192 (1994)
45. Richard Martin, K.: Large Scale Linear and Integer Optimization - A Unified Approach. Kluwer, Boston, MA, 1999
46. Martinhon, C., Lucena, A., Maculan, N.: Stronger k-tree relaxations for the vehicle routing problem. European Journal of Operational Research **158**, 56–71 (2004)
47. Miller, D.: A matching based exact algorithm for capacitated vehicle routing problems. ORSA Journal on Computing **7**, 1–9 (1995)
48. Müller-Hannemann, M., Schwartz, A.: Implementing weighted b-matching algorithms: Towards a flexible software design. In: Proceedings of the Workshop on Algorithm Engineering and Experimentation (ALENEX99). Vol. **1619** of Lecture notes in computer science. Springer-Verlag, Baltimore, MD, 1999, pp. 18–36
49. Naddef, D., Rinaldi, G.: Branch-and-cut algorithms for the capacitated VRP. In: P. Toth, D. Vigo (eds.), The Vehicle Routing Problem. SIAM, 2002, pp. 53–84
50. Nemhauser, G.L., Wolsey, L.A.: Integer and Combinatorial Optimization. John Wiley & Sons, Inc., USA, 1st edition, 1988
51. Nemhauser, G.L., Wolsey, L.A.: Integer and Combinatorial Optimization. Wiley, New York, 1988
52. Parker, R.G., Rardin, R.L.: Discrete Optimization. Academic Press, Inc., 1988
53. Pigatti, A.: Modelos e algoritmos para o problema de aloção generalizada e aplicações. PhD thesis, Pontifícia Universidade Católica do Rio de Janeiro, 2003
54. Prim, R.: Shortest connection networks and some generalizations. Bell System Technical Journal. **36**, 1389–1401 (1957)

55. Ralphs, T.K.: Parallel Branch and Cut for Vehicle Routing. PhD thesis, Cornell University, May 1995
56. Ralphs, T.K., Galati, M.V.: Decomposition in integer programming. In: J. Karlof (ed), Integer Programming: Theory and Practice. CRC Press, 2005. To appear
57. Ralphs, T.K., Kopman, L., Pulleyblank, W.R., Trotter, L.E., Jr: On the capacitated vehicle routing problem. Mathematical Programming **94**, 343–359 (2003)
58. Rousseau, L.M., Gendreau, M., Feillet, D.: Interior point stabilization for column generation. Working paper, Cahier du Gerad, 2003. Available from http://www.lia.univ-avignon.fr/fich_art/380-IPS.pdf
59. Chvátal, V.: Linear Programming. W.H. Freeman and Company, 1983
60. van den Akker, J.M., Hurkens, C.A.J., Savelsbergh, M.W.P.: Time-indexed formulations for machine scheduling problems: Column generation. INFORMS Journal on Computing **12**, 111–124 (2000)
61. Vanderbeck, F.: Lot-sizing with start-up times. Management Science **44**, 1409–1425 (1998)
62. Vanderbeck, F.: A generic view of Dantzig-Wolfe decomposition in mixed integer programming. Working paper, 2003. Available from http://www.math.u-bordeaux.fr/~fv/papers/dwcPap.pdf
63. Wei, G., Yu, G.: An improved $O(n^2 \log n)$ algorithm for the degree-constrained minimum k-tree problem. Technical report, The University of Texas at Austin, Center for Management of Operations and Logistics, 1995