Decomposition Methods

T.K. Ralphs* M.V. Galati†

July 5, 2010

1 Introduction

Decomposition methods are techniques for exploiting the tractable substructures of an integer program in order to obtain improved solution procedures. In particular, the goal is to derive improved methods of bounding the optimal solution value, which can then be used to drive a branch-and-bound algorithm. Such methods are the preferred solution approaches for a wide range of important models arising in practice and have also been the basis for solution approaches for many well-known combinatorial problems. A small sample of the problems for which decomposition approaches have been proposed in the literature includes the Multicommodity Flow Problem [11], the Cutting Stock Problem [15, 37], the Traveling Salesman Problem [19], the Generalized Assignment Problem [18, 35], the Bin Packing Problem [39], the Axial Assignment Problem [2], The Steiner Tree Problem [25], the Single-Machine Scheduling Problem [36], the Graph Coloring Problem [27], and the Capacitated Vehicle Routing Problem (CVRP) [1, 10, 32, 12].

To expose the desired substructure, a common approach is to relax a set of "complicating constraints." This is the approach taken by the Dantzig-Wolfe decomposition, Lagrangian relaxation, and cutting plane methods. Substructure can also be exposed by fixing the values of a set of variables, i.e., considering restrictions of the original problem. This is the approach taken by Benders' decomposition. This article reviews decomposition methodologies based on relaxation of constraints and examines how they are used to solve mixed integer linear programs. For a broader overview, including variable restriction methods, see [41].

To simplify the exposition, we consider only pure integer linear programs (ILPs) with finite upper and lower bounds on all variables, so that the set of feasible solutions is finite. The framework can easily be extended to more general settings. For the remainder of the paper, we consider an ILP instance whose feasible set is the integer vectors contained in the polyhedron $\mathcal{Q} = \{x \in \mathbb{R}^n \mid Ax \geq b\}$, 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 set of all *feasible solutions* to the ILP and let the polyhedron $\mathcal{P}$ be the convex hull of $\mathcal{F}$. In terms of this notation, the ILP is to determine

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

where $c \in \mathbb{Q}^n$. By convention, $z_{IP} = \infty$ if $\mathcal{F} = \emptyset$ (the problem is infeasible).

*Department of Industrial and Systems Engineering, Lehigh University, Bethlehem, PA 18015, ted@lehigh.edu, http://coral.ie.lehigh.edu/~ted

†Advanced Analytics - Operations R & D, SAS Institute, Chesterbrook, PA 19087 matthew.galati@sas.com, http://coral.ie.lehigh.edu/~magh

2 The Principle of Decomposition

Computing bounds is an essential element of the branch-and-bound algorithm, which is the most effective and most commonly used method for solving such mathematical programs. Bounds are often computed by solving a *bounding subproblem* that is a tractable relaxation 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 in solving difficult ILPs. Decomposition techniques based on constraint relaxation can be used to obtain improved bounding subproblems.

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

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

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 exist practical[1] algorithms for optimizing over and/or separating from $\mathcal{P}'$. With respect to $\mathcal{F}'$, let $[A'', b''] \in \mathbb{Q}^{m'' \times n''}$ be a set of additional *side constraints* 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

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

Note that $[A', b']$ and $[A'', b'']$ are often a partition of the rows of $[A, b]$ into a set of "nice constraints" and a set of "complicating constraints," but this is not a strict requirement.

Optimization and/or separation over $\mathcal{P}'$ may be more tractable than over $\mathcal{P}$ for a number of reasons. First, it may simply be that $\mathcal{P}'$ has a known structure for which we have well-developed techniques. This first case arises frequently when the original problem is formed by adding side constraints to a well-studied base problem. Second, the resulting relaxation may decompose due to identifiable block-diagonal structure of the matrix A'. In this case, optimization over $\mathcal{P}'$ can be accomplished by optimizing over each of these blocks independently (possibly in parallel). This second case arises when the constraints of A'' are constraints that link a collection of otherwise independent subsystems (see Example 2). A particular case in which decomposition is often applied very effectively is when A' consists of a set of identical blocks. In this case, the optimization problem over $\mathcal{P}'$ reduces to optimization over just one of the blocks. This can be seen as a way of eliminating symmetry present in the original problem and is one of the primary reasons why decomposition methods work well on certain classes of combinatorial problems.

3 Decomposition Techniques

In this section, we discuss three techniques for computing bounds on z_{IP} that are based on the principle of decomposition. In Section 4, we show that these methods are very closely related.

3.1 Lagrangian Relaxation

For a given vector of *dual multipliers* $u \in \mathbb{R}^{m''}_+$, the *Lagrangian relaxation* [9, 6, 28, 23] of (IP) is given by

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

[1] The term *practical* is not defined rigorously, but denotes an algorithm with a "reasonable" average-case running time.

It is then easily shown that $z_{LR}(u)$ is a lower bound on z_{IP}. The problem

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

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

The Lagrangian function $z_{LR}(u)$ is piecewise linear concave in u and hence can be maximized using the subgradient algorithm to obtain z_{LD}. Alternatively, it can also be solved by rewriting it as the equivalent linear program

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

Of course, this linear program has a large number of constraints and so must be solved by means of a cut generation algorithm (known as Kelley's cutting plane algorithm [21]). In either approach to solving the Lagrangian dual, most of the computational effort goes into evaluating $z_{LR}(u)$ (called the *Lagrangian subproblem*) for a given sequence of dual multipliers u. This is an optimization problem over $\mathcal{P}'$ and can be solved effectively by assumption. Both of these approaches are described in detail in [29].

3.2 Dantzig-Wolfe Decomposition

The approach of Dantzig-Wolfe decomposition (DWD) [8] is to reformulate (IP) by implicitly requiring the solution to be a member of $\mathcal{F}'$, while explicitly enforcing the inequalities $[A'', b'']$. Relaxing the integrality constraints on the variables from the original formulation, we obtain the linear program

$$z_{DW} = \min_{\lambda \in \mathbb{R}^{\mathcal{F}'}_{+}} \{c^{\top}(\sum_{s \in \mathcal{F}'} s\lambda_s) \mid A''(\sum_{s \in \mathcal{F}'} s\lambda_s) \geq b'', \sum_{s \in \mathcal{F}'} \lambda_s = 1\}. \tag{DWLP}$$

We call this LP the *Dantzig-Wolfe LP* (DWLP), though it is frequently referred to as the *master problem* in the literature. Although the number of columns in this linear program is $|\mathcal{F}'|$, it can be solved by column generation[2]. In each iteration of such a method, a restricted version of (DWLP), commonly referred to as the *restricted master problem* (RMP), is solved to obtain optimal primal and dual solutions. The column generation subproblem is to generate a new column that has negative reduced cost with respect to the dual solution $\hat{u}$ of the RMP (if one exists). This is equivalent to evaluating $z_{LR}(\hat{u})$, as in (LR).

Because the DWLP is solved by column generation, methods that use a DWLP to obtain bounds within a branch-bound-bound algorithm are sometimes referred to generically as *column generation methods* [5, 24]. In fact, most column generation algorithms for solving integer programs, even those for which the column generation derives directly from a "natural" formulation, can be seen as arising from the application of Dantzig-Wolfe decomposition to an underlying "compact" model, even if that compact model is not always first formulated explicitly [42]. Branch-and-bound approaches in which column generation is used to obtain a bound in each node are known as *branch-and-price* algorithms [34, 35, 40].

It is easy to verify that (DWLP) is an LP dual of (LDLP), which immediately shows that $z_{DW} = z_{LD}$ (see [31] for a detailed treatment of this fact). Hence, z_{DW} is a valid lower bound on z_{IP} that we

[2] Note that we can equivalently replace $\mathcal{F}'$ with the extreme points of $\mathcal{P}'$.

call the *DW bound*. Note that the dual variables in (DWLP) correspond to the dual multipliers from (LR). Furthermore, if we combine the members of $\mathcal{F}'$ using $\hat{\lambda}$, we obtain an optimal solution

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

to the Dantzig-Wolfe LP, which we call the *optimal fractional solution* corresponding to $\hat{\lambda}$. Note that (MAP) provides a mapping of each point in the feasible set of the Dantzig-Wolfe reformulation to a unique member of $\mathcal{P}' \cap \mathcal{Q}''$. This mapping will be used extensively in the methods of Section 5.1. Using this mapping, we can 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 $z_{DW} \geq z_{LP}$.

3.3 Cutting Plane Method

Although the cutting plane method [16, 7, 30, 20] is not usually thought of as a decomposition method, it can in fact be viewed as such. Under our assumption that the separation problem associated with $\mathcal{P}'$ can be solved effectively, a cutting plane method in which inequalities describing $\mathcal{P}'$ (i.e., the facet-defining inequalities) are generated dynamically can be implemented. By separating the solutions to a series of augmented LP relaxations from $\mathcal{P}'$, as in the usual cutting plane approach, the initial LP relaxation can be iteratively augmented to obtain the *CP bound*

$$z_{CP} = \min_{x \in \mathcal{P}'} \{c^\top x \mid A''x \geq b''\}. \tag{CPLP}$$

Note that $\hat{x}_{DW}$, as defined in (MAP), is an optimal solution to this augmented linear program. Hence, the CP bound is equal to both the DW bound and the LD bound. We refer to the augmented linear program above as the *cutting plane LP* (CPLP).

Although we have described the cutting plane method here as being based on our ability to solve the separation problem exactly over a single polyhedron $\mathcal{P}'$ containing $\mathcal{P}$, one of the advantages of the cutting plane method over the first two decomposition approaches is that it is possible in practice to generate valid inequalities from multiple decompositions simultaneously. This can further improve the bound beyond what can be achieved with either of the other two methods alone. Furthermore, it is also possible to obtain a valid bound without solving the separation problem *exactly*, i.e., the procedure does not have to guarantee to generate a valid inequality violated by the solution to the current LP relaxation whenever one exists. More details on these relationships between the methods are presented in the next section. In Section 5, we discuss how to integrate cutting plane methods with other decomposition techniques.

4 Relationship of the Techniques

The following well-known result showed that the three approaches just described are simply three different algorithms for computing the same quantity.

Theorem 1 *(Geoffrion [14])* $z_{IP} \geq c^\top \hat{x}_{DW} = z_{LD} = z_{DW} = z_{CP} = \min\{c^\top x \mid \mathcal{P}' \cap \mathcal{Q}''\} \geq z_{LP}$.

We refer to the quantity $z_D = \max\{c^\top x \mid \mathcal{P}' \cap \mathcal{Q}''\}$ as the *decomposition bound*. The proof of this result follows directly from the discussion of Section 3. Example 1 illustrates the application of the decomposition principle on a simple two-variable ILP and Figure 1 demonstrates computation of the bound graphically.

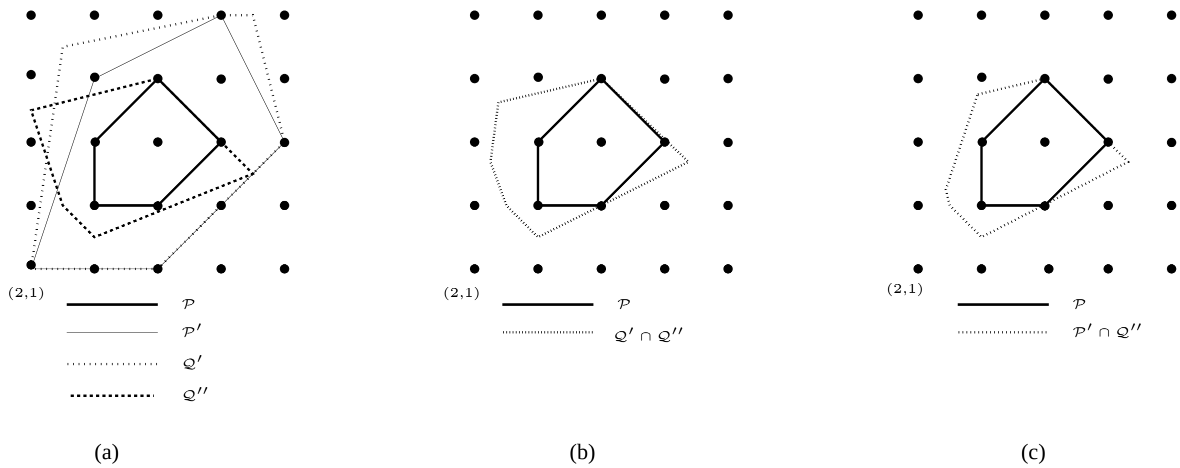

Figure 1: Polyhedra (Example 1: SILP)

Example 1: SILP Consider the following ILP in two variables:

$$
\begin{array}{llr}
\min & x_1 & \\
\text{s.t.} & 7\,x_1 - x_2 \geq 13, & (1) \\
& x_2 \geq 1, & (2) \\
& -\,x_1 + x_2 \geq -3, & (3) \\
& -4\,x_1 - x_2 \geq -27, & (4) \\
& -\,x_2 \geq -5, & (5) \\
& 0.2\,x_1 - x_2 \geq -4, & (6) \\
& -\,x_1 - x_2 \geq -8, & (7) \\
& -0.4\,x_1 + x_2 \geq 0.3, & (8) \\
& x_1 + x_2 \geq 4.5, & (9) \\
& 3\,x_1 + x_2 \geq 9.5, & (10) \\
& 0.25\,x_1 - x_2 \geq -3, & (11) \\
& x \in \mathbb{Z}^2. & (12)
\end{array}
$$

In this example, we let

$$
\begin{aligned}
\mathcal{P} &= \operatorname{conv}\left\{x \in \mathbb{R}^2 \mid x \text{ satisfies } (1) - (12)\right\}, \\
\mathcal{Q}' &= \left\{x \in \mathbb{R}^2 \mid x \text{ satisfies } (1) - (6)\right\}, \\
\mathcal{Q}'' &= \left\{x \in \mathbb{R}^2 \mid x \text{ satisfies } (7) - (11)\right\}, \text{ and} \\
\mathcal{P}' &= \operatorname{conv}(\mathcal{Q}' \cap \mathbb{Z}^2).
\end{aligned}
$$

In Figure 1(a), we show the associated polyhedra, where the set of feasible solutions $\mathcal{F} = \mathcal{Q}' \cap \mathcal{Q}'' \cap \mathbb{Z}^2 = \mathcal{P}' \cap \mathcal{Q}'' \cap \mathbb{Z}^2$ and $\mathcal{P} = \operatorname{conv}(\mathcal{F})$. Figure 1(b) depicts the continuous approximation $\mathcal{Q}' \cap \mathcal{Q}''$, while Figure 1(c) shows the improved approximation $\mathcal{P}' \cap \mathcal{Q}''$. For the objective function in this example, optimization over $\mathcal{P}' \cap \mathcal{Q}''$ leads to an improvement over the LP bound obtained by optimization over $\mathcal{Q}$. ■

It is easy to see that we can only have $z_D > z_{LP}$ when $\mathcal{P}' \subset \mathcal{Q}'$. This is formalized in the following corollary of Theorem 1.

Corollary 1 *If $\mathcal{Q}'$ is an integral polyhedron, then $z_D = z_{LP}$.*

It is important to point out that this result does not depend on the objective function—if $\mathcal{Q}'$ is integral, then $z_D = z_{LP}$ for *any* objective function vector. On the other hand, the converse is not true. If $\mathcal{Q}'$ is not integral, we might still have $z_D = z_{LP}$ for *certain* objective functions. In the following example, we illustrate the result of the corollary.

Example 2: GAP The Generalized Assignment Problem (GAP) is that of finding a minimum cost assignment of n tasks to m machines such that each task is assigned to precisely one machine subject to capacity restrictions on the machines. With each possible assignment, we associate a binary variable x_{ij}, which, if set to one, indicates that machine i is assigned to task j. For ease of notation, let us define two index sets $M = \{1, \ldots, m\}$ and $N = \{1, \ldots, n\}$. Then an ILP formulation of GAP is as follows:

$$\min \sum_{i \in M} \sum_{j \in N} c_{ij} x_{ij} \ ,$$

$$\sum_{j \in N} w_{ij} x_{ij} \leq b_i \ \forall i \in M, \tag{13}$$

$$\sum_{i \in M} x_{ij} = 1 \ \ \forall j \in N, \tag{14}$$

$$x_{ij} \in \{0, 1\} \qquad \forall i, j \in M \times N.$$

In this formulation, equations (14) ensure that each task is assigned to exactly one machine. Inequalities (13) ensure that for each machine, the capacity restrictions are met.

One possible decomposition of GAP is to let the relaxation be defined by the assignment constraints as follows:

$$\begin{aligned} \mathcal{Q}' &= \left\{ x_{ij} \in \mathbb{R}^+ \ \forall i, j \in M \times N | \ x \text{ satisfies } (14) \right\} \text{ and} \\ \mathcal{Q}'' &= \left\{ x_{ij} \in \mathbb{R}^+ \ \forall i, j \in M \times N | \ x \text{ satisfies } (13) \right\}. \end{aligned}$$

For this decomposition, the polytope $\mathcal{Q}'$ is integral, i.e., $\mathcal{Q}' = \text{conv}(\mathcal{Q}' \cap \{0, 1\}^{M \times N})$. Hence, according to Corollary 1, the decomposition bound z_D is equal to that of the LP relaxation. If we instead choose a relaxation defined by the capacity constraints, we get the following:

$$\begin{aligned} \mathcal{Q}' &= \left\{ x_{ij} \in \mathbb{R}^+ \ \forall i, j \in M \times N | \ x \text{ satisfies } (13) \right\} \text{ and} \\ \mathcal{Q}'' &= \left\{ x_{ij} \in \mathbb{R}^+ \ \forall i, j \in M \times N | \ x \text{ satisfies } (14) \right\}. \end{aligned}$$

In this case, the relaxation is a set of knapsack problems. These knapsack polyhedra do not have the integrality property, so the decomposition bound *may* be better than the LP bound. This is also an example of a decomposable subproblem—the optimization over $\mathcal{P}'$ reduces to a set of independent knapsack problems, one for each machine, which can be solved effectively in practice. ■

It should now be clear that these three methods are very closely related. In each of the three methods, we are given a polyhedron $\mathcal{P}$ over which we would like to optimize, along with two polyhedra that contain $\mathcal{P}$: the polyhedron $\mathcal{Q}''$, which has a small description and can be represented explicitly, and the polyhedron $\mathcal{P}'$, with a much larger description that must be represented implicitly. The conceptual difference between Dantzig-Wolfe decomposition, Lagrangian relaxation, and the cutting plane method is that both 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 corresponding separation problem. In this framework, most dynamic methods for solving ILPs, can be viewed as decomposition methods.

In fact, there are even deeper connections between decomposition methods based on outer approximation and the more traditional methods based on inner representation that serve to shed further light on the relationship. Recall that after solving (DWLP), we can obtain the optimal fractional solution $\hat{x}_{DW}$ corresponding to the optimal solution $\hat{\lambda}$ using the mapping (MAP). Note that $\hat{x}_{DW}$ is a convex combination of members of $\mathcal{F}'$. In particular, it is a combination of the members of the set $\{s \in \mathcal{F}' \mid \hat{\lambda}_s > 0\}$, which we will refer to as the *decomposition* of $\hat{x}_{DW}$ arising from $\hat{\lambda}$. It can be shown that $\hat{x}_{DW}$ must always be contained in a proper face of $\mathcal{P}'$ that we can characterize precisely as follows. Consider the set

$$\mathcal{S} = \{s \in \mathcal{F}' \mid (c^\top - \hat{u}^\top A'')s = (c^\top - \hat{u}^\top A'')\hat{s}\}, \tag{15}$$

where $\hat{s}$ is some member of the decomposition (and hence corresponds to a basic variable in an optimal solution to (DWLP)). The following theorem states that the set $\mathcal{S}$ must contain all members of the decomposition. The proof is straightforward and can be found in [33].

Theorem 2 $\{s \in \mathcal{F}' \mid \hat{\lambda}_s > 0\} \subseteq \mathcal{S}$.

In fact, we can show something stronger. The set $\mathcal{S}$ is comprised exactly of those members of $\mathcal{F}'$ corresponding to columns of the (DWLP) with reduced cost zero, so we have the following theorem. The proof is again straightforward and can be found in [33].

Theorem 3 $\text{conv}(\mathcal{S})$ *is a face of* $\mathcal{P}'$ *and contains* $\hat{x}_{DW}$.

An important consequence of this result is the following corollary, which states that the face of optimal solutions to (CPLP) is contained in $\text{conv}(\mathcal{S}) \cap \mathcal{Q}''$.

Corollary 2 *If* F *is the face of optimal solutions to* (CPLP)*, then* $F \subseteq \text{conv}(\mathcal{S}) \cap \mathcal{Q}''$.

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 (CPLP).

As noted earlier, $\text{conv}(\mathcal{S})$ is typically a proper face of $\mathcal{P}'$. It is possible, however, for $\hat{x}_{DW}$ to be an inner point of $\mathcal{P}'$. In this case, illustrated graphically in Figure 2(a), $z_{DW} = z_{LP}$ and Dantzig-Wolfe decomposition does not improve the bound. All columns of (DWLP) have reduced cost zero and any extremal member of $\mathcal{F}'$ could be made a member of the decomposition. Hence, the effectiveness of the procedure is reduced and the chosen relaxation may be too weak. A necessary condition for an optimal fractional solution to be an inner point of $\mathcal{P}'$ is that the dual value of the convexity constraint in an optimal solution to the (DWLP) be zero. This result is further examined in the next section.

A second case of potential interest is when $F = \text{conv}(\mathcal{S}) \cap \mathcal{Q}''$, illustrated graphically in Figure 2(b). This condition can be detected by examining the objective function values of the members of the

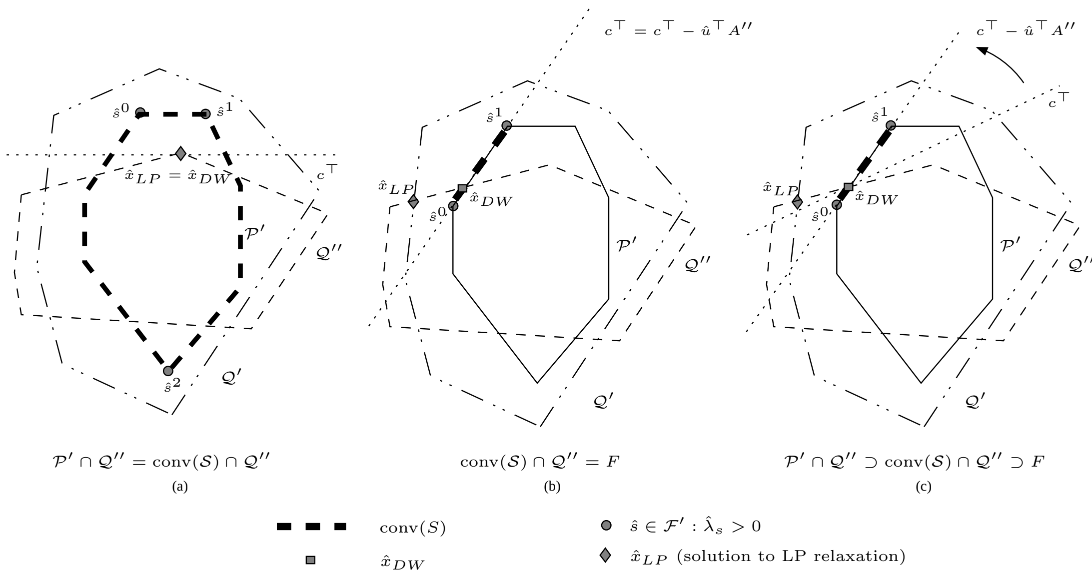

Figure 2: Illustration of bounds for different cost vectors

decomposition. If they are all equal, then 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_D. In this case, all constraints of (DWLP) *other than* the convexity constraint 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 $\text{conv}(\mathcal{S}) \cap \mathcal{Q}''$, is shown in Figure 2(c).

5 Advanced Methods

As previously mentioned, the cutting plane method has a number of advantages in practice over traditional decomposition approaches. Within the framework presented here, the generation of valid inequalities can be interpreted as a dynamic tightening of either $\mathcal{Q}''$ or $\mathcal{P}'$. We will focus on the former interpretation and discuss two methods that employ either (DWLP) or (LD) as the bounding subproblem, but also integrate generation of valid inequalities.

We first consider the basic steps of the cutting plane method, assuming the problem at hand is feasible and bounded. To begin, we construct an initial LP relaxation and solve it. We then perform the separation step, attempting to generate inequalities valid for $\mathcal{P}$ but violated by the resulting solution. If successful, we add these inequalities to the LP relaxation, hoping to improve the bound. This procedure is iterated until no more violated inequalities are found. The most important step is that of separation, which results in a tightening of $\mathcal{Q}''$ and (hopefully) an improved bound.

In principle, this procedure can be generalized by replacing (CPLP) with either (DWLP) or (LD) as the bounding subproblem. The steps of this generalized method are shown in Figure 3. The important step is Step 3, that of generating a set of *improving inequalities*, i.e., inequalities valid for $\mathcal{P}$ that when added to the description of the $\mathcal{P}'$ result in an increase in the computed bound. Aside from Step 3, the method is straightforward. Steps 1 and 2 are performed as in a traditional decomposition framework. Step 4 is accomplished by simply adding the newly generated inequalities to the list $[A'', b'']$ and re-forming the appropriate bounding subproblem.

It is important to point out that the generalized methods described in Figure 3 requires that we

Decomposition with Cut Generation

1. Construct the initial bounding subproblem P^0 and set $i \leftarrow 0$.

   $z_{CP} = \min_{x \in \mathcal{P}'} \{c^\top x \mid A'' x \geq b''\}$
   $z_{LR} = \max_{u \in \mathbb{R}^n_+} \min_{x \in \mathcal{P}'} \{(c^\top - u^\top A'')x + u^\top b''\}$
   $z_{DW} = \min_{\lambda \in \mathbb{R}^{\mathcal{F}'}_+} \{c^\top (\sum_{s \in \mathcal{F}'} s\lambda_s) \mid A''(\sum_{s \in \mathcal{F}'} s\lambda_s) \geq b'', \sum_{s \in \mathcal{F}'} \lambda_s = 1\}$

2. Solve P^i to obtain a valid lower bound z^i.

3. Generate a set of *improving inequalities* $[D^i, d^i]$ valid for $\mathcal{P}$.

4. If valid inequalities were found in Step 3, form the bounding subproblem P^{i+1} by setting $[A'', b''] \leftarrow \left[\begin{smallmatrix} A'' & b'' \\ D^i & d^i \end{smallmatrix}\right]$. Then, set $i \leftarrow i + 1$ and go to Step 2.

5. If no valid inequalities were found in Step 3, then output z^i.

Figure 3: Basic outline of the dynamic decomposition method

can solve the subproblem (LR) for *any* objective function. In some cases, the algorithm used for solving the subproblem may only be efficient when the objective function has certain structure. In such cases, it may only be possible to integrate generation of classes of valid inequalities whose addition does not destroy the required structure of the subproblem.

In the cutting plane method, techniques for generating valid inequalities typically measure the potential effectiveness of adding a given valid inequality by the degree to which it violates $\hat{x}_{DW}$, since violation of $\hat{x}_{DW}$ is a necessary condition for an inequality to be improving. It is not clear, however, what criteria should be used to measure potential effectiveness if the bounding subproblem is either (DWLP) or (LD). In the next two sections, we discuss these two cases in more detail.

5.1 Branch, Price, and Cut

We first consider using (DWLP) as the bounding subproblem in the procedure of Figure 3, a procedure we call *price and cut*. When employed as the bounding procedure in a branch-and-bound framework, the overall technique is called *branch, price, and cut* and has been studied by a number of authors [38, 22, 4, 36]. Simultaneous generation of columns and valid inequalities is difficult in general because the addition of valid inequalities may destroy the structure of the column generation subproblem (for a discussion of this, see [36]). Having solved (DWLP), however, we can easily recover an optimal solution to (CPLP) using (MAP) and try to generate improving inequalities using standard separation methods for $\mathcal{P}$. Because the generation of these valid inequalities takes place in the space of the original formulation, it does not destroy the structure of the column generation subproblem (with the possible exception of the objective function, as discussed above). Hence, the approach enables dynamic generation of valid inequalities while still retaining the bound improvement and other advantages yielded by a Dantzig-Wolfe decomposition. It is important to note, however, that the case of block decompositions with identical subproblems presents unique challenges to the implementation of price and cut, since collapsing the identical subproblems into a single subproblem means that the mapping (MAP) cannot be used to provide a unique optimal fractional solution (see [40] for a discussion).

The same bound improvement obtained in price and cut could be realized in the cutting plane method by adding the generated inequalities directly to (CPLP). Despite the apparent symmetry, there is one fundamental difference between the cutting plane method and price and cut. An optimal solution to (DWLP) provides a decomposition of $\hat{x}_{DW}$ (MAP) into a convex combination of members of $\mathcal{F}'$. In particular, the weight on each member of $\mathcal{F}'$ in the combination is given by the optimal decomposition.

An important observation is that an inequality can be improving only if it is violated by at least one member of the decomposition. This follows easily from the fact that the degree of violation of $\hat{x}_{DW}$ is a convex combination of the degrees of violation of each member of the decomposition. In some cases, it is much easier to separate members of $\mathcal{F}'$ from $\mathcal{P}$ than to separate arbitrary real vectors. In fact, it is easy to find polyhedra for which the problem of separating an arbitrary real vector is difficult, but the problem of separating a solution to a given combinatorial relaxation is easy. This notion has been discussed in the literature in several contexts [32, 26, 3] and leads to a separation technique that can be embedded within price and cut[33]. The idea is to replace direct separation of the fractional point (which may be difficult) with separation of members of the decomposition, which act as surrogates. Given a decomposition, the efficiency of this procedure depends both on the cardinality of the decomposition (which must be less than or equal to $\dim(\mathcal{P}') + 1$) and the time required to separate members of $\mathcal{P}'$ from $\mathcal{P}$.

5.2 Branch, Relax, and Cut

We now move on to discuss the dynamic generation of valid inequalities when the bounding subproblem is (LD). When employed in a branch-and-bound framework, the overall method is called *branch, relax, and cut* and has also been studied previously by several authors (see [26] for a survey). Solving the LD as the linear program (LDLP) is equivalent to solving (DWLP), so in this case, the methods just discussed apply directly without modification. We assume from here on that the LD is solved in the form (LD), using subgradient optimization.

Suppose we have solved (LD) to obtain $\hat{u}$, a vector of optimal multipliers and $\hat{s} = \text{argmin } z_{LR}(\hat{u})$. As before, let $\hat{\lambda}$ be an optimal decomposition and $\hat{x}_{DW} = \sum_{s \in \mathcal{F}'} s\hat{\lambda}_s$ be an optimal fractional solution. The main consequence of using subgradient optimization to solve (LD) is that we do not obtain the primal solution information available to us with both the cutting plane method and price and cut. This means there is no obvious way of constructing an optimal fractional solution or verifying that any generated valid inequalities would be violated by such a solution. We can, however, attempt to separate the solution to the Lagrangian relaxation (LR), which is a member of $\mathcal{F}'$, from $\mathcal{P}$. Note that again, we are 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 $[A'', b'']$. This has the effect of introducing a new dual multiplier and slightly perturbing the objective function used to solve the Lagrangian relaxation.

As with both the previously discussed 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 [17]. As we have already noted, we cannot verify that generated inequalities are violated by an optimal fractional solution, which is the usually employed 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 now further examine the relationship between $\hat{s}$ and $\hat{x}_{DW}$.

From the reformulation of the LD as the linear program (LDLP), we can see 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 (LDLP) correspond to variables of (DWLP) with reduced cost zero, so it follows immediately that the set of all alternative solutions to the Lagrangian subproblem with multipliers $\hat{u}$ is the set $\mathcal{S}$ from (15).

Because $\hat{x}_{DW}$ is both an optimal solution to (CPLP) and is contained in $\mathrm{conv}(\mathcal{S})$, it also follows that

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

In other words, the penalty term in the objective function of the Lagrangian subproblem (LR) serves to rotate the original objective function so that it becomes parallel to the face $\mathrm{conv}(\mathcal{S})$, 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 shown in Figure 2(c).

The fundamental connection between relax and cut and price and cut is that solving (DWLP) produces a *set* of alternative optimal solutions to the LD, 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, on the other hand, produces only one member of this set, though possibly at a much lower computational cost. Even if improving inequalities exist, it is possible that none of them are violated by $\hat{s}$, especially if $\hat{s}$ has a small weight in the optimal decomposition. 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 dual multipliers $\hat{u}$. This situation is depicted in Figure 2(a). In this case, separating $\hat{s}$ is unlikely to yield an improving inequality.

6 Conclusion

Decomposition algorithms capitalize on our knowledge of identifiable substructure in a given integer programming model. This knowledge could be in the form of either a method of optimizing over or separating from the convex hull of solutions to a given relaxation. In the former case, an inner representation of the convex hull can be implicitly used in either a Dantzig-Wolfe decomposition or Lagrangian relaxation framework. In the latter case, an outer representation can be used in a cutting plane framework. The use of advanced techniques that integrate generation of valid inequalities with the use of either Dantzig-Wolfe decomposition or Lagrangian relaxation are an attempt to achieve the "best of both worlds." Implementations of these advanced algorithms are becoming more common, and this trend is sure to continue as more software tools become available.

References

[1] Agarwal, Y., Mathur, K., and Salkin, H. M. A set-partitioning-based exact algorithm for the vehicle routing problem. *Networks 19* (1989), 731–749.

[2] Balas, E., and Saltzman, M. Facets of the three-index assignment polytope. *Discrete Applied Mathematics 23* (1989), 201–229.

[3] Balas, E., and Saltzman, M. An algorithm for the three-index assignment problem. *Operations Research 39* (1991), 150–161.

[4] Barnhart, C., Hane, C. A., and Vance, P. H. Using branch-and-price-and-cut to solve origin-destination integer multi-commodity flow problems. *Operations Research 48* (2000), 318–326.

[5] Barnhart, C., Johnson, E. L., Nemhauser, G. L., Savelsbergh, M. W. P., and Vance, P. H. Branch and price: Column generation for solving huge integer programs. *Operations Research 46* (1998), 316–329.

[6] Beasley, J. Lagrangean relaxation. In *Modern Heuristic Techniques for Combinatorial Optimization*, C. Reeves, Ed. Wiley, 1993.

[7] Dantzig, G. B., Fulkerson, D. R., and Johnson, S. M. Solution of a large scale traveling salesman problem. *Operations Research 2* (1954), 393–410.

[8] Dantzig, G. B., and Wolfe, P. Decomposition principle for linear programs. *Operations Research 8*, 1 (1960), 101–111.

[9] Fisher, M. The Lagrangian relaxation method for solving integer programming problems. *Management Science 27* (1981), 1–18.

[10] Fisher, M. Optimal solution of vehicle routing problems using minimum k-trees. *Operations Research 42*, 4 (1994), 626–642.

[11] Ford, L. R., and Fulkerson, D. R. A suggested computation for maximal multicommodity network flows. *Management Science 5* (1958), 97–101.

[12] Fukasawa, R., Longo, H., Lysgaard, J., de Aragão, M. P., Reis, M., Uchoa, E., and Werneck, R. Robust branch-and-cut-and-price for the capacitated vehicle routing problem. *Mathematical Programming 106* (2006), 491–511.

[13] Galati, M. V. *Decomposition in Integer Programming*. PhD thesis, Lehigh University, 2009.

[14] Geoffrion, A. Lagrangian relaxation for integer programming. *Mathematical Programming Study 2* (1974), 82–114.

[15] Gilmore, P. C., and Gomory, R. E. A linear programming approach to the cutting stock problem. *Operations Research 9* (1961), 849–859.

[16] Gomory, R. E. Outline of an algorithm for integer solutions to linear programs. *Bulletin of the American Mathematical Monthly 64* (1958), 275–278.

[17] Guignard, M. Efficient cuts in Lagrangean "relax-and-cut" schemes. *European Journal of Operational Research 105* (1998), 216–223.

[18] Guignard, M., and Rosenwein, M. An improved dual-based algorithm for the generalized assignment problem. *Operations Research 37* (1989), 658–663.

[19] Held, M., and Karp, R. M. The traveling salesman problem and minimum spanning trees. *Operations Research 18* (1970), 1138–1162.

[20] Hoffman, K., and Padberg, M. LP-based combinatorial problem solving. *Annals of Operations Research 4* (1985), 145–194.

[21] Kelley, J. E. The cutting plane method for solving convex programs. *Journal of the SIAM 8* (1960), 703–712.

[22] Kohl, N., Desrosiers, J., Madsen, O., Solomon, M., and Soumis, F. 2-path cuts for the vehicle routing problem with time windows. *Transportation Science 33* (1999), 101–116.

[23] Lemaréchal, C. Lagrangean relaxation. In *Computational Combinatorial Optimization*, M. Jünger and D. Naddef, Eds. Springer, 2001, pp. 112–156.

[24] Lübbecke, M., and Desrosier, J. Selected topics in column generation. *Operations Research 53*, 6 (2005), 1007–1023.

[25] Lucena, A. Steiner problem in graphs: Lagrangean relaxation and cutting planes. *COAL Bulletin 28* (1992), 2–8.

[26] Lucena, A. Non delayed relax-and-cut algorithms. *Annals of Operations Research 140*, 1 (2005), 375–410.

[27] Mehrotra, A., and Trick, M. A column generation approach to graph coloring. *INFORMS Journal on Computing 8*, 4 (1996), 344–354.

[28] Neame, P. J. *Nonsmooth Dual Methods in Integer Programming.* PhD thesis, University of Melbourne, 1999.

[29] Nemhauser, G., and Wolsey, L. *Integer and Combinatorial Optimization.* Wiley, New York, 1988.

[30] Padberg, M. W., and Rinaldi, G. A branch and cut algorithm for the solution of large scale traveling salesman problems. *SIAM Review 33* (1991), 60–100.

[31] Parker, R., and Rardin, R. *Discrete Optimization.* Academic Press, Inc., 1988.

[32] Ralphs, T., Kopman, L., Pulleyblank, W., and Trotter Jr., L. On the capacitated vehicle routing problem. *Mathematical Programming 94* (2003), 343–359.

[33] Ralphs, T. K., and Galati, M. V. Decomposition and dynamic cut generation in integer linear programming. *Mathematical Programming 106*, 2 (2006), 261–285.

[34] Ryan, D., and Foster, B. An integer programming approach to scheduling. In *Computer scheduling of public transport urban passenger vehicle and crew scheduling*, A. Wren, Ed. North-Holland, 1981, pp. 269–80.

[35] Savelsbergh, M. A branch-and-price algorithm for the generalized assignment problem. *Operations Research 45*, 6 (1997), 831–841.

[36] van den Akker, J., Hurkens, C., and Savelsbergh, M. Time-indexed formulations for machine scheduling problems: Column generation. *INFORMS Journal on Computing 12* (2000), 111–124.

[37] Vance, P. H., Barnhart, C., Johnson, E. L., and Nemhauser, G. L. Solving binary cutting stock problems by column generation and branch and bound. *Computational Optimization and Applications 3* (1994), 111–130.

[38] Vanderbeck, F. Lot-sizing with start-up times. *Management Science 44* (1998), 1409–1425.

[39] Vanderbeck, F. Computational study of a column generation algorithm for bin packing and cutting stock problems. *Mathematical Programming 86*, 3 (1999), 565–594.

[40] Vanderbeck, F. Branching in branch-and-price: a generic scheme. *Mathematical Programming* (2010). To appear, available at `http://www.springerlink.com/content/2572975068528215/`.

[41] Vanderbeck, F., and Wolsey, L. *Reformulation and Decomposition of Integer Programs.* Springer, 2010, pp. 431–498.

[42] Villeneuve, D., Desrosiers, J., Lübbecke, M., and Soumis, F. On compact formulations for integer program solved by column generation. *Annals of Operations Research 139* (2005), 375–388.