Algorithms for Mixed Integer Bilevel Programs
Exploiting the Lower-level Value Function

Scott DeNegre and Ted Ralphs

Department of Industrial and Systems Engineering
Lehigh University

INFORMS 2009, San Diego
12 October 2009
A linear program (LP) is the problem of minimizing a linear objective over the feasible region

\[ S_{LP} = \{ x \in \mathbb{R}^n \mid Ax \geq b \} , \]

defined by \( A \in \mathbb{Q}^{m \times n} \), \( b \in \mathbb{R}^m \). That is, the goal of linear programming is to determine

\[ \bar{z}_{LP} = \min_{x \in S_{LP}} cx, \]

where \( c \in \mathbb{Q}^n \).

- Changing any member of the triple \((A, b, c)\) yields a perturbation of (LP).
- We focus on changes to the right-hand-side vector \( b \).
Parametric LP

Considering all possible right-hand-sides yields the *parameterized* version of (LP):

\[
(z_{LP}(v) = \min_{x \in S_{LP}(v)} cx, \quad (LP(v)))
\]

where

\[
S_{LP}(v) = \{ x \in \mathbb{R}^n_+ \mid Ax \geq v \},
\]

for all \( v \in \mathbb{R}^n \).

\( z_{LP} : \mathbb{R}^m \to \mathbb{R} \cup \{ \pm \infty \} \) is the LP *value function*.

- For each \( d \in \mathbb{R}^m \), \( z_{LP} \) returns the optimal value of (LP).
- \( z_{LP} \) is piecewise linear and convex over \( \Omega_{LP} = \{ v \in \mathbb{R}^m \mid S_{LP}(v) \neq \emptyset \} \).
A mixed integer linear program (MILP) is a well-known generalization of LP.
- In MILP, a specified subset of the variables is required to take on integer values.
- Let this subset be indexed 1 through \( p \leq n \).

Formally, the goal of mixed integer linear programming is to determine

\[
\begin{align*}
    z_{IP} &= \min_{x \in S_{IP}} cx, \\
    S_{IP} &= \left\{ x \in \mathbb{Z}^p_+ \times \mathbb{R}^{n-p}_+ \mid Ax \geq b \right\},
\end{align*}
\]

where \((A, b, c)\) are defined as before.
Parametric MILP

The parameterized version of (MILP) is defined as:

\[
z_{IP}(v) = \min_{x \in S_{IP}(v)} cx, \quad (\text{MILP}(v))
\]

where

\[
S_{IP}(v) = \left\{ x \in \mathbb{Z}^p_+ \times \mathbb{R}^{n-p}_+ \mid Ax \geq v \right\},
\]

for all \( v \in \mathbb{R}^n \).

As in the LP case, we refer to \( z_{IP} : \mathbb{R}^m \to \mathbb{R} \cup \{\pm \infty\} \) as the value function.

- By convention, we say \( z_{IP} = \infty \) if \( d \notin \Omega_{IP} = \{ v \in \mathbb{R}^m \mid S_{IP}(v) \neq \emptyset \} \) and \( z_{IP} = -\infty \) if the objective value is unbounded.

- \( z_{IP} \) is piecewise polyhedral but nonconvex.
Multilevel Programming

- Multilevel programming is a generalization of traditional mathematical programming that applies to hierarchical decision systems.

- In a multilevel program:
  - The variables are divided into groups controlled by separate DMs.
  - The constraints of each DM involve the variables of DMs at higher levels in the hierarchy.
  - The DMs have independent, possibly conflicting objectives.

- Conceptually, the variables are fixed sequentially in accordance with the inherent system hierarchy.

- We assume individual rationality of the DMs.
  - DMs will be able to predict the reaction of other DMs to decisions made above them.
  - We can collapse the hierarchy into a single optimization model in which the decisions made at the highest level determine the system outcome.
We focus on *bilevel programs*, described as follows.

- \( x \in X \subseteq \mathbb{R}^{n_1}_+ \) are the *upper-level variables*
- \( y \in Y \subseteq \mathbb{R}^{n_2}_+ \) are the *lower-level variables*

**Bilevel Program**

\[
\begin{align*}
\min \left\{ c^1 x + d_1^1 \bar{y} \mid x \in P_U \cap X, \bar{y} \in \arg\min \{d_2^2 y \mid y \in S_L(x) \cap Y\} \right\} \\
\text{(BP)}
\end{align*}
\]

The *upper- and lower-level feasible regions* are

\[
P_U = \{ x \in \mathbb{R}^{n_1}_+ \mid A^1 x \geq b^1 \}
\]

and

\[
S_L(x) = \{ y \in \mathbb{R}^{n_2}_+ \mid G^2 y \geq b^2 - A^2 x \} .
\]
Notation

We utilize the following notation:

\[
\begin{align*}
\Omega &= \{(x, y) \in \mathbb{R}_+^{n_1} \times \mathbb{R}_+^{n_2} \mid x \in \mathcal{P}_U, y \in \mathcal{S}_L(x)\} \\
\Omega^I &= \Omega \cap X \times Y \\
M(x) &= \text{argmin}\{d^2y \mid y \in \mathcal{S}_L(x)\} \\
M^I(x) &= M(x) \cap Y \\
\mathcal{F} &= \{(x, y) \mid x \in \mathcal{P}_U, y \in M(x)\} \\
\mathcal{F}^I &= \{(x, y) \mid x \in \mathcal{P}^I_U, y \in M^I(x)\}
\end{align*}
\]

We can think of a bilevel program as a traditional optimization problem in which a parametric program has been embedded.
Technical Assumptions

We make the following assumptions in order to ensure the problem is well-posed and has a solution.

Assumptions

1. For every action by the leader, the follower has a rational reaction \((S_L(x) \cap Y \neq \emptyset \text{ for all } x \in \mathcal{P}_U \cap X)\).
2. The follower is semi-cooperative (the leader may choose among alternative members of \(M^I(x)\)).
3. The feasible set \(\mathcal{F}^I\) is nonempty and compact.

The BP can be simply stated as:

Bilevel Program

\[
\max_{(x,y) \in \mathcal{F}^I} c^1x + d^1y. \\
\text{(BP)}
\]
The following instance of (BP) is from Moore and Bard (1990).

\[
\begin{align*}
\text{min} & \quad -x - 10\bar{y} \\
\text{subject to} & \quad \bar{y} \in \arg\min \{y : -25x + 20y \leq 30, \\
& \quad x + 2y \leq 10, \\
& \quad 2x - y \leq 15, \\
& \quad 2x + 10y \geq 15, \\
& \quad y \in Y \} 
\end{align*}
\]

From the figure, we can make several observations:

1. \( \mathcal{F} \subseteq \Omega, \mathcal{F}^I \subseteq \Omega^I, \text{ and } \Omega^I \in \Omega \)
2. \( F^I \not\subseteq \mathcal{F} \)
3. In this instance, \( z_{IP}(v), v \in \{1, \ldots, 8\} \) is identical to \( F^I \)

This motivates our study of the value function of the lower-level program.
In algorithms for solving MILPs, we frequently use the following properties.

**Properties**

1. If the continuous relaxation has no feasible solution, then neither does the original problem.
2. If the continuous relaxation has a solution, then its objective value is a valid lower bound on that of the original problem.
3. If the solution to the continuous relaxation is integral, then it is optimal for the original problem.

Properties 2 and 3 result from the fact that the set of feasible solutions for the original MILP is contained in the feasible set of the relaxation.

*THIS IS NOT THE CASE FOR MIBLP*
Properties of MIBLPs

In this example:

- Optimizing over $\mathcal{F}$ yields the integer solution $(8, 1)$, with the upper-level objective value 18.
- Imposing integrality yields the solution $(2, 2)$, with upper-level objective value 22.

From this we can make two important observations:

1. The objective value obtained by relaxing integrality is not a valid bound on the solution value of the original problem since we may have

   $$\min_{(x,y)\in\mathcal{F}} c^1 x + d^1 y > \min_{(x,y)\in\mathcal{F}^I} c^1 x + d^1 y.$$ 

2. Even when solutions to $\min_{(x,y)\in\mathcal{F}} c^1 x + d^1 y$ are in $\mathcal{F}^I$, they are not necessarily optimal.

Thus, only *Property 1* remains valid.
Bounding Method

Relaxing integrality conditions *and* the requirement \( y \in M^I(x) \) yields the relaxation

\[
\min_{(x,y) \in \Omega} c^1 x + d^1 y. \tag{LR}
\]

- The resulting bound can be used in combination with a standard variable branching scheme to yield an algorithm that solves (BP).
- The bound is too weak to be effective on interesting problems.
- We can strengthen the linear relaxation with inequalities valid for \( F^I \) to improve the bound.

### Bilevel Feasibility Conditions

- \((x, y) \in \Omega\),
- \((x, y) \in X \times Y\), and
- \(y \in M^I(x)\).
Pure Integer Problems

In the case where \( X = \mathbb{Z}^{n_1} \) and \( Y = \mathbb{Z}^{n_2} \), we can derive simple valid inequalities to strengthen the relaxation. Let

\[
A := \begin{bmatrix} A^1 \\ A^2 \end{bmatrix}, \quad G := \begin{bmatrix} 0 \\ G^2 \end{bmatrix}, \quad \text{and} \quad b := \begin{bmatrix} b^1 \\ b^2 \end{bmatrix}.
\]

A basic feasible solution \((\hat{x}, \hat{y}) \in \Omega^I\) to (LR) is the unique solution to

\[
d_i'x + g_i'y = b_i, \quad i \in I
\]

where \( I \) is the set of active constraints at \((\hat{x}, \hat{y})\).

This implies that

\[
\left\{ (x, y) \in \Omega^I \mid \sum_{i \in I} a_i'x + g_i'y = \sum_{i \in I} b_i \right\} = \{ (\hat{x}, \hat{y}) \}
\]

and \( \sum_{i \in I} a_i'x + g_i'y \leq \sum_{i \in I} b_i \) is valid for \( \Omega \).
A Valid Inequality

$$\sum_{i \in I} a_i'x + g_i'y \leq \sum_{i \in I} b_i - 1$$ is valid for \( \Omega^I \setminus \{(x, y)\} \).

We can see from the figure that these cuts are fairly weak.
Value Function Reformulation

Let

\[ z_L(x) = \min_{y \in S_L(x) \cap Y} d^2 y. \]

We can rewrite (BP) as:

\[
\begin{align*}
\min & \quad c^1 x + d^1 y \\
\text{subject to} & \quad A^1 x \geq b^1 \\
& \quad A^2 x + G^2 y \geq b^2 \\
& \quad d^2 y = z_L(x) \\
& \quad x \in X, y \in Y.
\end{align*}
\]

1. Determining the structure of \( z_L \) is very difficult in general.
2. We can derive approximations of the value function.
Upper Approximations

We saw, in the previous talk, how to construct upper-approximations.

- We adopt all assumptions given there in what follows here.
- Let $y^*$ be an optimal solution to

$$\min_{y \in S_L} d^2 y$$

with right-hand-side $x = \bar{x}$.

- One upper-bounding function is given by:

$$f(x) = d_I^2 y_I^* + z_{L_c} (b^2 - A^2 x - G_i^2 y_i^*)$$

where $I = \{1, \ldots, p_2\}$ and $C = \{p_2 + 1, \ldots, n_2\}$ and $z_{L_c}$ is the value function of the continuous relaxation of the lower-level problem.

- This bound is strong.

We can apply this upper bound in several ways.
Applying the Approximation

One way to apply the bounding function is generate a series of such upper bounds at different right-hand-sides.

- Replacing $d^2y = z_L(x)$ with
  \[ d^2y \leq f(x), \]

  results in a relaxation that would produce a bound.

- Let $H^U$ be the set of upper-bounding functions for each $x \in U \subset \mathbb{R}^{m_2}$, for some collection of upper-level solutions.

Then, we have the following relaxation of (BP):

\[
\begin{align*}
\min & \quad c^1x + d^1y \\
\text{subject to} & \quad A^1x \geq b^1 \\
& \quad A^2x + G^2y \geq b^2 \\
& \quad d^2y \leq \min_{f \in H^U} f(x) \\
& \quad x \in X, y \in Y.
\end{align*}
\]

Generating “enough” of these functions yields the original problem.
Applying the Approximation (cont.)

Alternatively, we can use this bounding function to generate valid disjunctions to be used in a branch-and-bound or branch-and-cut algorithm.

- Note $z_{Lc} = \max_{u \in V} \{ u(b^2 - A^2x) \}$, where $V$ is the set of extreme points of the dual polytope.
- Thus, for each $x$, $f$ corresponds to some dual feasible basis $u$.

This allows us to create the disjunction of the form:

\[
\begin{align*}
\text{Disjunction} & \\
\left\{ \begin{array}{c}
u = u_1 \\
d^2y \leq f_1(x)
\end{array} \right\} & \text{OR} & \cdots & \text{OR} & \left\{ \begin{array}{c}
u = u_{|V|} \\
d^2y \leq f_{|V|}(x)
\end{array} \right\}.
\end{align*}
\]

We illustrate this method in for the case of a lower-level problem with a single constraint next.
Lower-level Problems with a Single Constraint

Suppose the lower-level problem contains only a single constraint.

\[ S_L(x) = \{ y \in \mathbb{R}^{n_2}_+ \mid g^2 y = b^2 - a^2 x \} . \]

Let \( C^+ = \{ i \in C \mid g_i^2 > 0 \} \) and \( C^- = \{ i \in C \mid g_i^2 < 0 \} \), and

\[ \eta^C = \min \left\{ \frac{d_i^2}{g_i^2} \mid i \in C^+ \right\} \quad \text{and} \quad \zeta^C = \max \left\{ \frac{d_i^2}{g_i^2} \mid i \in C^- \right\} . \]

Then, this disjunction reduces to:

**Bilevel Feasibility Disjunction**

\[
\begin{align*}
&a^2 x \\
&\zeta^C a^2 x + d^2 y \geq \zeta^C a^2 \bar{x} + d^2 y^* \quad \text{OR} \quad a^2 x \\
&\eta^C a^2 x + d^2 y \leq \eta^C a^2 \bar{x} + d^2 y^*
\end{align*}
\]
Illustrating the disjunction

This can immediately be used to develop a stronger branching scheme when solutions \((\bar{x}, \bar{y}) \in \Omega^I\) such that \(\bar{y} \notin M^I(\bar{x})\) are found.
A Disjunctive Cut Approach

Consider the two polyhedra that result if we impose this disjunction on the original set of constraints in \( \Omega \). This yields the polyhedra:

\[
P^1 = \begin{cases} 
A^1x \\ a^2x + g^2y \\ a^2x \\ -\zeta_c a^2x - d^2y \\ x, y 
\end{cases} \geq \begin{cases} 
b^1 \\ b^2 \\ a^2\bar{x} \\ -\zeta_c a^2\bar{x} - d^2y^* \\ 0 
\end{cases}
\]

and

\[
P^2 = \begin{cases} 
A^1x \\ a^2x + g^2y \\ -a^2x \\ -\eta_c a^2x - d^2y \\ x, y 
\end{cases} \geq \begin{cases} 
b^1 \\ b^2 \\ -a^2\bar{x} \\ -\eta_c a^2\bar{x} - d^2y^* \\ 0. 
\end{cases}
\]
Let \((u^i, v^i, w^i, z^i)\) be multipliers for the constraints in polyhedron \(P^i\). The following inequalities are valid for \(P^1\):

\[
\begin{align*}
  u^1 A^1 x + v^1 a^2 x + w^1 a^2 \bar{x} - z^1 \zeta C a^2 x + v^1 g^2 y - z^1 d^2 y & \geq \\
  u^1 b^1 + v^1 b^2 + w^1 a^2 \bar{x} - z^1 (\zeta C a^2 \bar{x} + d^2 y^*)
\end{align*}
\]

and \(P_2\):

\[
\begin{align*}
  u^2 A^1 x + v^2 a^2 x - w^2 a^2 x - z^2 \eta C a^2 x + v^2 g^2 y - z^2 d^2 y & \geq \\
  u^2 b^1 + v^2 b^2 - w^2 a^2 \bar{x} - z^2 (\eta C a^2 \bar{x} + d^2 y^*)
\end{align*}
\]

It is well-known that, given these inequalities, we can construct an inequality \(\alpha x + \beta y \geq \gamma\) that is valid for \(\text{conv}(P^1 \cup P^2)\) by selecting \(\alpha\), \(\beta\), and \(\gamma\) such that

\[
\alpha \geq \max\{\pi^1_1, \pi^2_1\}, \quad \beta \geq \max\{\pi^1_2, \pi^2_2\}, \quad \text{and} \quad \gamma \leq \min\{\pi^1_0, \pi^2_0\}.
\]
Thus, the inequality $\alpha x + \beta y \geq \gamma$ is valid for $\text{conv}(P_1 \cup P_2)$ if

$$\alpha - (u_1^+ - u_1^-)A^1 - (v_1^+ - v_1^-)a^2 - w^1a^2 + z^1\zeta^Ca^2 \geq 0$$

$$\alpha - (u_2^+ - u_2^-)A^1 - (v_2^+ - v_2^-)a^2 + w^2a^2 + z^2\eta^Ca^2 \geq 0$$

$$\beta - (v_1^+ - v_1^-)g^2 + z^1d^2 \geq 0$$

$$\beta - (v_2^+ - v_2^-)g^2 + z^2d^2 \geq 0$$

$$\gamma - (u_1^+ - u_1^-)b^1 - (v_1^+ - v_1^-)b^2 - w^1a^2\bar{x} + z^1(\zeta^Ca^2\bar{x} - d^2y^*) \leq 0$$

$$\gamma - (u_2^+ - u_2^-)b^1 - (v_2^+ - v_2^-)b^2 + w^2a^2\bar{x} + z^2(\eta^Ca^2\bar{x} - d^2y^*) \leq 0$$

$$u^1, u^1, u^2, u^2, v^1, v^2, w^1, w^2, z^1, z^2 \geq 0.$$
To find the deepest cut, we can solve the *cut generation LP*:

\[
\begin{align*}
\text{min} & \quad \alpha \bar{x} + \beta \bar{y} - \gamma \\
\text{s.t.} & \quad \alpha - (u^{1+} - u^{1-})A^1 - (v^{1+} - v^{1-})A^2 - w^1a^2 + z^1\zeta C a^2 \geq 0 \\
& \quad \alpha - (u^{2+} - u^{2-})A^1 - (v^{2+} - v^{2-})a^2 + w^2A^2 + z^2\eta C a^2 \geq 0 \\
& \quad \beta - (v^{1+} - v^{1-})g^2 + z^1d^2 \geq 0 \\
& \quad \beta - (v^{2+} - v^{2-})g^2 + z^2d^2 \geq 0 \\
& \quad \gamma - (u^{1+} - u^{1-})b^1 - (v^{1+} - v^{1-})b^2 - w^1a^2\bar{x} + z^1(\zeta C a^2\bar{x} + d^2y^*) \leq 0 \\
& \quad \gamma - (u^{2+} - u^{2-})b^1 - (v^{2+} - v^{2-})b^2 + w^2a^2\bar{x} + z^2(\eta C a^2\bar{x} + d^2y^*) \leq 0 \\
& \quad u^{1+} + u^{1-} + v^{1+} + v^{1-} + w^1 + z^1 + u^{2+} + u^{2-} + v^{2+} + v^{2-} + w^2 + z^2 = 1 \\
& \quad u^{1+}, u^{1-}, u^{2+}, u^{2-}, v^{1+}, v^{1-}, v^{2+}, v^{2-}, w^1, w^2, z^1, z^2 \geq 0,
\end{align*}
\]

similar to that used in constructing lift-and-project cuts.
Current Work

We are currently working on developing methods:

- Reduce the size of the disjunction in the general case
- Employ the lower approximations in our algorithmic framework
- Preprocess the problems to increase the algorithm’s speed.