To Branch or to Cut.

Or, what am I going to do with this disjunction?

Ashutosh Mahajan[1] and Ted Ralphs[2]

[1]Mathematics and Computer Science Division
Argonne National Laboratory

[2]Industrial and Systems Engineering
Lehigh University

August 27, 2009

COR@L
COMPUTATIONAL OPTIMIZATION RESEARCH AT LEHIGH

The Main Question

For this talk ...

Given a "General Disjunction", we can use it in branch-and-bound or to generate valid inequalities. How do we decide to use it?

The Main Question

For this talk . . .

Given a “General Disjunction”, we can use it in branch-and-bound or to generate valid inequalities. How do we decide to use it?

The above question is mostly useless

The real question is: When should we stop cutting and start branching?

- Cutting plane methods have greatly improved our ability to solve Integer Programs.
- Cutting planes alone (the ones used today) are not sufficient.
- Theoretically, as well as computationally.
- It is important to understand when should we resort to cutting and when to branching.
- It is so difficult . . .

We look at the fundamental building block of branch-and-bound and cutting-plane algorithms: A Disjunction

Quick Review

The Problem

$$\begin{aligned} z_{IP} &= \min cx \\ s.t. \;\; Ax &\geq b \qquad \text{(IP)} \\ x &\in \mathbb{Z}^n, \end{aligned}$$

where $A \in \mathbb{Q}^{m \times n}$, $b \in \mathbb{Q}^m$, $c \in \mathbb{Q}^n$, $m, n \in \mathbb{N}$ are given.

Quick Review

The Problem

$$
\begin{aligned}
z_{IP} &= \min cx \\
s.t. \;\; Ax &\geq b \qquad \text{(IP)} \\
x &\in \mathbb{Z}^n,
\end{aligned}
$$

where $A \in \mathbb{Q}^{m \times n}$, $b \in \mathbb{Q}^m$, $c \in \mathbb{Q}^n$, $m, n \in \mathbb{N}$ are given.

A Relaxation

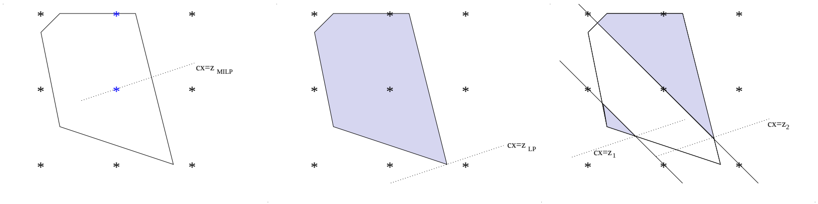
$$
\begin{aligned}
z_{LP} &= \min cx \\
s.t. \;\; Ax &\geq b \qquad \text{(LP)} \\
x &\in \mathbb{R}^n,
\end{aligned}
$$

where $A \in \mathbb{Q}^{m \times n}$, $b \in \mathbb{Q}^m$, $c \in \mathbb{Q}^n$, $m, n \in \mathbb{N}$ are given.

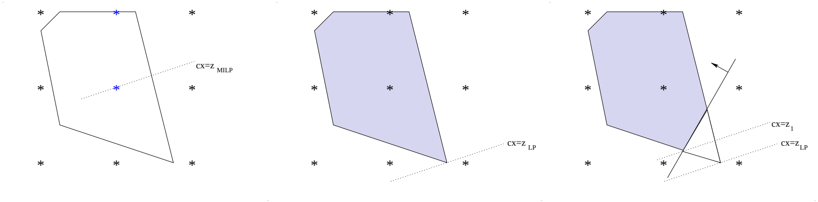
Basic Approach:

1. $z_{LP} \leq z_{IP}$ provides a lower bound (z_l) on z_{IP}.
2. Any $\hat{x} \in Z^d \times \mathbb{R}^{n-d}$ s.t. $A\hat{x} \geq b$ provides an upper bound (z_u) on z_{IP}.
3. "Tighten" the feasible region of the (LP) relaxation iteratively.
4. Repeat until $z_l = z_u$.

Two Algorithms



Branch and Bound Algorithm



Cutting Plane Algorithm

Disjunctions

If $(\hat{\pi}, \hat{\pi}_0) \in \mathbb{Z}^{n+1}$, any $\hat{x} \in \mathbb{Z}^n$ must satisfy the disjunction

$$\hat{\pi}\hat{x} \leq \hat{\pi}_0 \vee \hat{\pi}\hat{x} \geq \hat{\pi}_0 + 1 \tag{1}$$

- When $\pi = ([0, \ldots, 0, 1, 0, \ldots, 0])$, we call (π, π_0) a Variable Disjunction. E.g. $x_2 \leq 1 \vee x_2 \geq 2$.
- Otherwise we call it a General Disjunction. e.g. $2x_1 + 5x_2 - 2x_3 \leq 0 \vee 2x_1 + 5x_2 - 2x_3 \geq 1$.
- There are other types of disjunctions as well.

For the given IP:

$$\begin{aligned} z_{IP} &= \min cx \\ s.t.\ \ Ax &\geq b \qquad (2) \\ x &\in \mathbb{Z}^n, \end{aligned}$$

(1) can now be strengthened to:

Any $\hat{x}$ feasible to (IP) must satisfy:

$$\begin{array}{ll} Ax \geq b & \\ \hat{\pi}x \leq \hat{\pi}_0 & (\mathcal{P}_1) \\ x \in \mathbb{R}^n & \end{array} \quad \bigvee \quad \begin{array}{ll} Ax \geq b & \\ \hat{\pi}x \geq \hat{\pi}_0 + 1 & (\mathcal{P}_2) \\ x \in \mathbb{R}^n & \end{array}$$

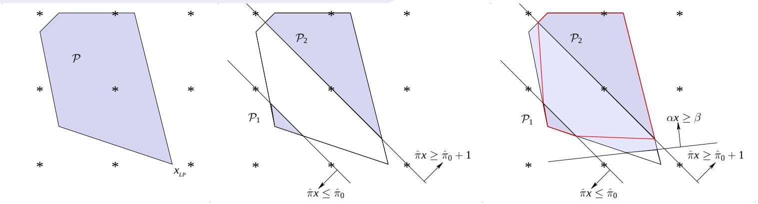
Disjunctions for . . .

. . . Branching

- Solve $\min_{x \in \mathcal{P}_1} cx$, $\min_{x \in \mathcal{P}_2} cx$ separately.
- $\mathcal{P}_1 \cup \mathcal{P}_2 \subseteq \mathcal{P}$.
- Two different subproblems after branching.

. . . Generating Split Cuts

- Find an inequality $(\alpha, \beta) \in \mathbb{R}^{n+1}$ valid for $cl(conv(\mathcal{P}_1 \cup \mathcal{P}_2))$.
- Add (α, β) to (LP) to get a tighter relaxation.



Same disjunction, Different purposes

The same disjunction (π, π_0) can be used for either purpose.

Some History

Branching

- Mostly limited to variable disjunctions.
- Disjunctions that "improve the bound" the most are favorable: Strong Branching, Pseudo-cost Branching, Reliability Branching
- (Benichou, 1971), (Linderoth and Savelsbergh, 1999), (Achterberg et al., 2005)
- General disjunctions have been used for polynomial time algorithms in fixed dimension. (Lenstra, 1983) etc.

Some History

Branching

- Mostly limited to variable disjunctions.
- Disjunctions that "improve the bound" the most are favorable: Strong Branching, Pseudo-cost Branching, Reliability Branching
- (Benichou, 1971), (Linderoth and Savelsbergh, 1999), (Achterberg et al., 2005)
- General disjunctions have been used for polynomial time algorithms in fixed dimension. (Lenstra, 1983) etc.

Split Inequalities

1. (Cook et al., 1990), (Nemhauser & Wolsey, 1990), (Balas, 1971)
2. Nearly all classes of cuts are Split Cuts: Chvátal-Gomory, Lift and Project, Flow & Cover, …(Cornuejols, 2008).
3. Inequalities with "larger violation" are favorable.
4. Underlying disjunctions could be variable disjunctions or general disjunctions.

Give and Take

Why not use branching variable-disjunctions to generate cutting planes?

Lift and Project. (Balas et. al, 1993), (Balas and Perregaard, 2002).

Give and Take

Why not use branching variable-disjunctions to generate cutting planes?

Lift and Project. (Balas et. al, 1993), (Balas and Perregaard, 2002).

Why not use (general) cutting-plane-disjunctions for branching?

- (Karamanov and Cornuéjols, 2007). Several disjunctions from GMI.
- (Cornuéjols et. al, 2008). Several improved disjunctions from the Simplex Tableau.
- Use “strong-branching” to select the best disjunction.

Give and Take

Why not use branching variable-disjunctions to generate cutting planes?

Lift and Project. (Balas et. al, 1993), (Balas and Perregaard, 2002).

Why not use (general) cutting-plane-disjunctions for branching?

- (Karamanov and Cornuéjols, 2007). Several disjunctions from GMI.
- (Cornuéjols et. al, 2008). Several improved disjunctions from the Simplex Tableau.
- Use “strong-branching” to select the best disjunction.

We try the following

Use general-branching-disjunctions to generate cuts.

What is a nice general-branching-disjunction?

Give and Take

Why not use branching variable-disjunctions to generate cutting planes?

Lift and Project. (Balas et. al, 1993), (Balas and Perregaard, 2002).

Why not use (general) cutting-plane-disjunctions for branching?

- (Karamanov and Cornuéjols, 2007). Several disjunctions from GMI.
- (Cornuéjols et. al, 2008). Several improved disjunctions from the Simplex Tableau.
- Use "strong-branching" to select the best disjunction.

We try the following

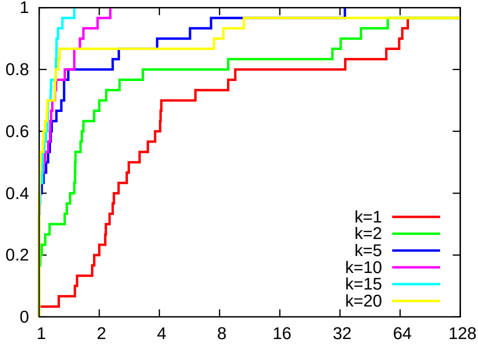
Use general-branching-disjunctions to generate cuts.

What is a nice general-branching-disjunction?

One that maximizes bound — a strongest general disjunction.

Finding a strong general disjunction

- See (A.M. and Ralphs, 2009a, 2009b) for details.
- Formulated a MIP with parameter K, such that it is feasible if and only if there exists a disjunction $(\hat{\pi}, \hat{\pi}_0)$ that will improve the bound of original instance to at least K.
- Solve a sequence of MIPs with varying K.
- Add additional constraints (like # non-zeros in disjunction $\leq k$).



Performance Profile of number of nodes in branch-and-bound tree.

Moving from Branching to Cutting

- Find a strong general disjunctions as before.
- Use it to find valid inequalities.
- $cx \geq K$ is a one of them. We are interested in more (and different) inequalities.
- How to find these?

It depends

- What type of inequalities do we want? – C-G cuts, Split cuts.
- What quality measure is used? – Violation of the current LP solution (x_{LP}).

C-G Cuts

For a given IP:

$$\begin{aligned} z_{IP} &= \min cx \\ s.t.\ \ Ax &\geq b \\ x &\in \mathbb{Z}^n, \end{aligned} \tag{3}$$

- For any $u \in \mathbb{R}^m$, $uAx \geq ub$ is valid for the LP relaxation.
- If $uA \in \mathbb{Z}^n$, then $uAx \geq \lceil ub \rceil$ is valid for the IP – C-G cut.
- Let $\hat{\pi} = uA, \hat{\pi}_0 = \lceil ub \rceil - 1$ and consider the disjunction $(\hat{\pi}, \hat{\pi}_0)$.
- $\mathcal{P}_1 = \{x | Ax \geq b, \hat{\pi}x \leq \hat{\pi}_0\} = \phi$
- $\hat{\pi}x \geq \hat{\pi}_0 + 1$ is a valid inequality.
- It is trivially the best inequality obtained from the disjunction $(\hat{\pi}, \hat{\pi}_0)$.

Finding C-G Cuts

$$
\begin{aligned}
z_{IP} &= \min cx \\
s.t. \;\; Ax &\geq b \qquad \text{(IP)} \\
x &\in \mathbb{Z}^n,
\end{aligned}
\qquad\qquad
\begin{aligned}
z_{LP} &= \min cx \\
s.t. \;\; Ax &\geq b \qquad \text{(LP)} \\
x &\in \mathbb{R}^n,
\end{aligned}
$$

$(\hat{\pi}, \hat{\pi}_0) \in \mathbb{Z}^{n+1}$ is a disjunction that gives a C-G inequality that raises the bound to a given K.

Finding C-G Cuts

$$
\begin{aligned}
z_{IP} &= \min cx \\
s.t.\ \ Ax &\geq b \qquad \text{(IP)} \\
x &\in \mathbb{Z}^n,
\end{aligned}
\qquad\qquad
\begin{aligned}
z_{LP} &= \min cx \\
s.t.\ \ Ax &\geq b \qquad \text{(LP)} \\
x &\in \mathbb{R}^n,
\end{aligned}
$$

$(\hat{\pi}, \hat{\pi}_0) \in \mathbb{Z}^{n+1}$ is a disjunction that gives a C-G inequality that raises the bound to a given K.

$\iff$ both the following LPs in x are infeasible.

$$
\begin{aligned}
Ax &\geq b \\
\hat{\pi}x &\leq \hat{\pi}_0 \\
x &\in \mathbb{R}^n.
\end{aligned}
\qquad\qquad
\begin{aligned}
Ax &\geq b \\
cx &\leq K \\
\hat{\pi}x &\geq \hat{\pi}_0 + 1 \\
x &\in \mathbb{R}^n.
\end{aligned}
$$

Finding C-G Cuts

$$\begin{aligned} z_{IP} &= \min cx \\ s.t. \quad Ax &\geq b \qquad \text{(IP)} \\ x &\in \mathbb{Z}^n, \end{aligned} \qquad\qquad \begin{aligned} z_{LP} &= \min cx \\ s.t. \quad Ax &\geq b \qquad \text{(LP)} \\ x &\in \mathbb{R}^n, \end{aligned}$$

$(\hat{\pi}, \hat{\pi}_0) \in \mathbb{Z}^{n+1}$ is a disjunction that gives a C-G inequality that raises the bound to a given K.

$\iff$ both the following LPs in x are infeasible.

$$\begin{aligned} Ax &\geq b \\ \hat{\pi}x &\leq \hat{\pi}_0 \\ x &\in \mathbb{R}^n. \end{aligned} \qquad\qquad \begin{aligned} Ax &\geq b \\ cx &\leq K \\ \hat{\pi}x &\geq \hat{\pi}_0 + 1 \\ x &\in \mathbb{R}^n. \end{aligned}$$

$\iff$ both the following LPs are feasible.

$$\begin{aligned} pA - \hat{\pi} &= 0 \\ pb - \hat{\pi}_0 &> 0 \\ p &\in \mathbb{R}^m_+ \end{aligned} \qquad\qquad \begin{aligned} qA - sc + \hat{\pi} &= 0 \\ qb - sK + \hat{\pi}_0 &> -1 \\ q \in \mathbb{R}^m_+, s &\in \mathbb{R}_+ \end{aligned}$$

Finding C-G Cuts

We can obtain a C-G inequality that increases the lower bound to K if and only if the following MIP is feasible:

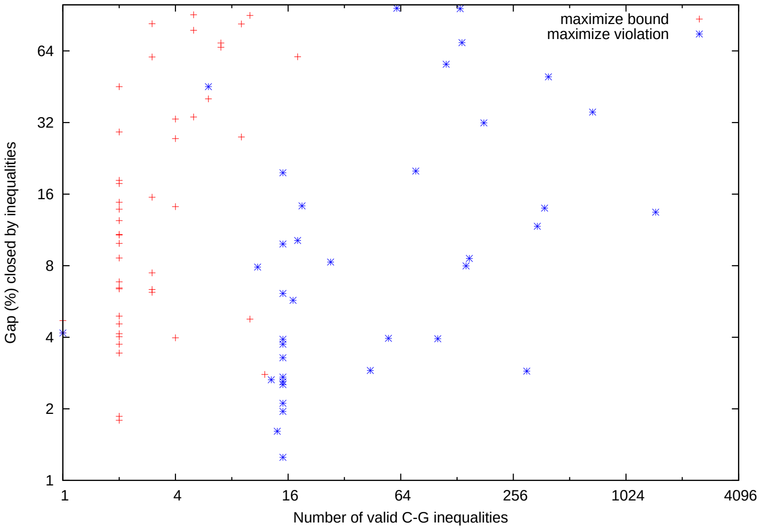
$$\begin{aligned} pA - \pi &= 0 \\ pb - \pi_0 &> 0 \\ qA - sc + \pi &= 0 \\ qb - sK + \pi_0 &> -1 \\ p \in \mathbb{R}^m_+ q \in \mathbb{R}^m_+&, s \in \mathbb{R}_+ \\ \pi \in \mathbb{Z}^n, \pi_0 &\in \mathbb{Z} \end{aligned} \tag{4}$$

- Similar to formulation of (Fischetti and Lodi, 2005) for optimizing over C-G closure.
- They select the maximum violated C-G inequality.
- A computational experiment to compare the two formulations.

Computational Experiment: C-G Cuts

- We need to solve (4) for different values of K.
- Set a time limit of 1000s for all iterations of (4) (200s for each run).
- Set a time limit of 1000s for the MIP formulation to find the maximum violation C-G cut.
- 177 instances from MIPLIB-3, MIPLIB-2003, Mittlemann-Set
- CPLEX-10.2, Coin-Utils.
- 2GB RAM, 4MB Cache, 1.86GHz, 64bit-LINUX.
- Time limit for each instance: 20 hours.

Computational Experiment: C-G Cuts



Finding Split Cuts

Using the same approach, one can derive split-inequalities

We can obtain split-inequalities from a single disjunction such that the lower bound increases to K if and only if the following MIP is feasible:

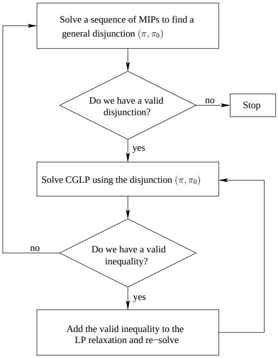
$$
\begin{aligned}
pA - s_L c - \pi &= 0 \\
pb - s_L K - \pi_0 &> 0 \\
qA - s_R c + \pi &= 0 \\
qb - s_R K + \pi_0 &> -1 \\
p \in \mathbb{R}^m_+ q \in \mathbb{R}^m_+, s_L, s_R &\in \mathbb{R}_+ \\
\pi \in \mathbb{Z}^n, \pi_0 &\in \mathbb{Z}
\end{aligned} \tag{5}
$$

Important Difference from C-G cuts

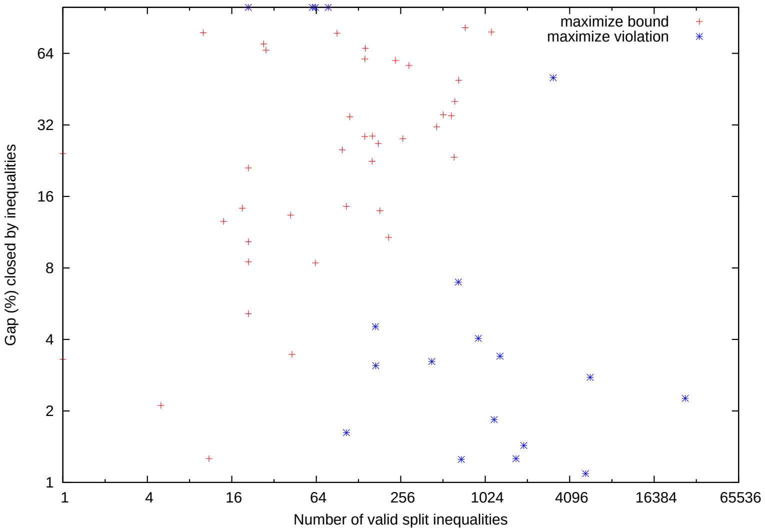
We can generate many split-inequalities using a given disjunction. We use CGLP to generate such inequalities.

Finding Split Cuts

- (Balas and Saxena, 2008) use a similar approach to find maximally violated split inequalities.
- Both methods require solution to a parametric MIP.
- After a disjunction is obtained, we get inequalities by solving a CGLP in both cases.



Computational Experiment: Split Cuts



Conclusions

- Set up experiments to use branching-disjunctions to generate valid inequalities for the cutting-plane algorithm
- Results suggest that disjunctions selected for maximum bound improvement might be more useful than those selected for maximum violation
- Might be another piece of evidence suggesting branching and valid inequalities should be viewed more holistically.
- Cuts derived from such disjunctions may be selected by different criteria.
- Need to devise new methods of identifying good disjunctions.
- Should we cut or branch at a given stage of the branch-and-cut algorithm.

To Branch or to Cut.

Or, what am I going to do with this disjunction?

Ashutosh Mahajan[1] and Ted Ralphs[2]

[1]Mathematics and Computer Science Division
Argonne National Laboratory

[2]Industrial and Systems Engineering
Lehigh University

August 27, 2009

COR@L
COMPUTATIONAL OPTIMIZATION RESEARCH AT LEHIGH