AM 121: Intro to Optimization Models and Methods
Fall 2016

Lecture 16: More cuts, Branch and Cut, other tricks…

David C. Parkes
SEAS

Lesson Plan

• Review Chvátal-Gomory cuts
• Cover cuts
• Branch-and-cut
• Other tricks
  – Preprocessing
  – Primal heuristics
  – SOS branching

Jensen & Bard: 8.5
• **Definition.** An inequality \( a^T x \leq b \) is **valid** for set \( X \subseteq \mathbb{R}^n \) if \( a^T x \leq b \) for all \( x \in X \).

• **Defn.** A **cut** is a valid inequality that **separates** the current fractional solution \( x^* \).

**Chvátal-Gomory inequalities**

• Integer, non-negative decision variables
• Consider \( X = P \cap \mathbb{Z}^2 \), where \( P \) is given by
  
  \[
  \begin{align*}
  7x_1 - 2x_2 & \leq 14 \\
  x_2 & \leq 3 \\
  2x_1 - 2x_2 & \leq 3 \\
  x & \geq 0
  \end{align*}
  \]

• Valid to combine with non-neg weights \( u = (2/7, 37/63, 0) \):
  
  \[
  2x_1 + \frac{1}{63}x_2 \leq \frac{121}{21}
  \]

• Valid to use non-neg of \( x \) to round coefficients on LHS down to nearest integer:
  
  \[
  2x_1 + 0x_2 \leq \frac{121}{21}
  \]

• Valid to use integrality of LHS to round down RHS:
  
  \[
  2x_1 \leq 5
  \]
Example (Gomory’s algorithm)

• In addition to integer, non-negative decision variables, needs integer coefficients and integer RHs.

• Consider the IP
  \[
  z = \text{max } 4x_1 - x_2 \\
  7x_1 - 2x_2 \leq 14 \\
  x_2 \leq 3 \\
  2x_1 - 2x_2 \leq 3 \\
  x_1, x_2 \geq 0, \text{ integer}
  \]

• For row \(i\) with fractional RHS, the CG cut is
  \[
  \sum_{j \in B'} (\bar{a}_{ij} - [\bar{a}_{ij}]) x_j \geq \bar{b}_i - [\bar{b}_i]
  \]

• \(z + \frac{4}{7} x_3 + \frac{1}{7} x_4 = \frac{59}{7}
  \]
  \[
  x_1 + \frac{1}{7} x_3 + \frac{2}{7} x_4 = \frac{20}{7}
  \]
  \[
  x_2 + x_4 = 3
  \]
  \[
  -\frac{2}{7} x_3 + \frac{10}{7} x_4 + x_5 = \frac{23}{7}
  \]

• \(B=\{1, 2, 5\}\)
• For row \( i \) with fractional RHS, the CG cut is \( \sum_{j \in B'} (\overline{a}_{ij} - [\overline{a}_{ij}]) x_j \geq \overline{b}_i - \overline{b}_i \)

\[
\begin{align*}
\text{z} & \quad + \quad \frac{4}{7} x_3 \quad + \quad \frac{1}{7} x_4 \quad = \quad \frac{59}{7} \\
\text{x}_1 & \quad + \quad \frac{1}{7}x_3 \quad + \quad \frac{2}{7} x_4 \quad = \quad \frac{20}{7} \\
\text{x}_2 & \quad + \quad x_4 \quad = \quad 3 \\
\quad - \frac{2}{7} x_3 \quad + \quad 10/7 x_4 \quad + \quad x_5 \quad = \quad \frac{23}{7}
\end{align*}
\]

• \( B=\{1, 2, 5\} \)
• Add cut \( \frac{1}{7}x_3 + \frac{2}{7}x_4 \geq \frac{6}{7} \)
• Add excess variable \( x_6 \)

---

• Re-optimize. New optimal tableau:
\[
\begin{align*}
\text{z} & \quad + \quad \frac{1}{2} x_5 \quad + \quad 3x_6 \quad = \quad \frac{15}{2} \\
\text{x}_1 & \quad + \quad x_6 \quad = \quad 2 \\
\text{x}_2 & \quad - \frac{1}{2} x_5 \quad + \quad x_6 \quad = \quad \frac{1}{2} \\
\text{x}_3 & \quad - x_5 \quad - \quad 5x_6 \quad = \quad 1 \\
\text{x}_4 & \quad + \frac{1}{2} x_5 \quad + \quad 6x_6 \quad = \quad \frac{5}{2}
\end{align*}
\]

• \( B=\{1,2,3,4\} \) and \( x^*=(2,1/2,1, 5/2, 0, 0) \)
• Add cut \( \frac{1}{2}x_5 \geq \frac{1}{2} \)
• Add excess variable \( x_7 \)
• Re-optimize. New optimal tableau:

\[
\begin{align*}
z &+ 3x_6 + x_7 = 7 \\
x_1 &+ x_6 = 2 \\
x_2 &+ x_6 - x_7 = 1 \\
x_3 &- 5x_6 - 2x_7 = 2 \\
x_4 &+ 6x_6 + x_7 = 2 \\
x_5 &- x_7 = 1
\end{align*}
\]

• \( x^*=(2, 1, 2, 2, 1, 0, 0) \)
Cover inequalities

• Defined for problems in which the feasible space is a “0-1 Knapsack set”. Example:
  \[ X = \{ x \in \{0,1\}^7 : 11x_1 + 6x_2 + 6x_3 + 5x_4 + 5x_5 + 4x_6 + x_7 \leq 19 \} \]

  \[ \text{0/1 variables} \quad \text{positive coefficients} \quad \text{positive RHS} \]

• In general, we allow binary IPs with multiple such “knapsack rows” but throughout our discussion we will assume a single such row.

Cover inequalities

• Defined for problems in which the feasible space is a “0-1 Knapsack set”. Example:
  \[ X = \{ x \in \{0,1\}^7 : 11x_1 + 6x_2 + 6x_3 + 5x_4 + 5x_5 + 4x_6 + x_7 \leq 19 \} \]

  \[ \text{0/1 variables} \quad \text{positive coefficients} \quad \text{positive RHS} \]

• Example cover inequalities for \( X \) are:
  \[
  \begin{align*}
  x_1 + x_2 + x_3 & \leq 2 \\
  x_1 + x_2 + x_6 & \leq 2 \\
  x_1 + x_5 + x_6 & \leq 2 \\
  x_3 + x_4 + x_5 + x_6 & \leq 3
  \end{align*}
  \]

  Using \{0,1\} property!
Cover inequalities

- For $X = \{x \in \{0,1\}^n : \sum_j a_j x_j \leq b\}$, with $a_j \geq 0$, $b \geq 0$.
- $N = \{1, \ldots, n\}$
- **Defn.** A set $C \subseteq N$ is a **cover** if $\sum_{j \in C} a_j > b$. A cover is **minimal** if $C \setminus \{j\}$ is not a cover for any $j \in C$.
- **Proposition.** If $C$ is a cover, the **cover inequality** $\sum_{j \in C} x_j \leq |C| - 1$ is valid.
- **Proof.** Consider $x \in \{0,1\}^n$ with $\sum_{j \in C} x_j = |C|$. Have $\sum_j a_j x_j \geq \sum_{j \in C} a_j x_j = \sum_{j \in C} a_j > b$. Not in $X$.

Any $x$ that violates cover is infeasible, not in $X$. So, cover inequality is valid.

Back to the examples

- $X = \{x \in \{0,1\}^7 : 11x_1 + 6x_2 + 6x_3 + 5x_4 + 5x_5 + 4x_6 + x_7 \leq 19\}$

  0/1 variables positive coefficients positive RHS

- Example cover inequalities for $X$ are:
  - $C = \{1,2,3\}$: $x_1 + x_2 + x_3 \leq |C| - 1 = 2$
  - $C = \{1,2,6\}$: $x_1 + x_2 + x_6 \leq |C| - 1 = 2$
Review: Cover inequalities

- For $X = \{x \in \{0,1\}^n : \sum_j a_j x_j \leq b\}$, with $a_j \geq 0$, $b \geq 0$.
- Given a cover set $C$, the **cover inequality** is $\sum_{j \in C} x_j \leq |C| - 1$.
- Equivalently, can write a cover inequality as $\sum_{j \in C} (1 - x_j) \geq 1$

- A cover set $C$ has the property that $\sum_{j \in C} a_j > b$.

How can we use Cover inequalities?
• **Definition.** An inequality $a^T x \leq b$ is **valid** for set $X \subseteq \mathbb{R}^n$ if $a^T x \leq b$ for all $x \in X$.

• **Defn.** A **cut** is a valid inequality that **separates** the current fractional solution $x^*$.

![Fractional solution and members of set X](image)

Separation with Cover inequalities

• Can write cover inequality as 
  $$\sum_{j \in C} (1 - x_j) \geq 1$$
  ("at least one not used")

• Given frac $x^*$, find valid $C$ with $\sum_{j \in C}(1-x^*_j)<1$

• Formulate an IP! Let $z_j$ denote whether $j \in C$.
  
  \[
  \beta = \min \sum_{j \in N} (1- x^*_j) z_j \\
  \text{s.t.} \quad \sum_{j \in N} a_j z_j > b \quad \text{(valid cover)} \\
  z \in \{0,1\}^n
  \]

• If $\beta \geq 1$, then $x^*$ satisfies all cover inequalities.
• If $\beta < 1$, then we find a cut.

• Note: this IP is nonstandard (*strict inequality*) But can replace with $\sum_{j \in N} a_j z_j \geq b+1$ when coeffs are integral.
Example use of a Cover Inequality

- $X=\{x \in \{0,1\}^6: 45x_1+46x_2+79x_3+54x_4+53x_5+125x_6 \leq 178\}$, fractional $x^*=(0, 0, 3/4, 1/2, 1, 0)$
- Solve:
  \[ \beta = \min z_1 + z_2 + \frac{1}{4} z_3 + \frac{1}{2} z_4 + 0z_5 + z_6 \]
  s.t. $45z_1 + 46z_2 + 79z_3 + 54z_4 + 53z_5 + 125z_6 > 178$
  $z \in \{0,1\}^6$
- $z^*=(0,0,1,1,1,0)$ with $\beta=3/4$
- Conclude that $x_3 + x_4 + x_5 \leq 2$ is a cut for $x^*$.

*Why is this useful?* Because we can try to solve the “auxiliary problem” heuristically; give up if too hard.

Two Variations on Cover Inequalities

- Extended Cover Inequalities
- Lifted Cover Inequalities
Extended Cover Inequalities

• $X=\{x \in \{0,1\}^7 : 11x_1+6x_2+6x_3+5x_4+5x_5+4x_6+x_7 \leq 19\}$
• Cover inequality: $x_3 + x_4 + x_5 + x_6 \leq 3$
• Extended cover ineq: $x_1 + x_2 + x_3 + x_4 + x_5 + x_6 \leq 3$
• Still valid. Why? Any four variables have sum coefficients > 19. (And so $x \in X$ must satisfy)

• **Rule**: Can extend cover set $C$ to extended cover set $E(C)$, where $E(C) = C \cup \{j : a_j \geq a_i \text{ for all } i \in C\}$.
• Extended cover inequality: $\sum_{j \in E(C)} x_j \leq |C|-1$ is valid, and stronger than cover inequality.

Two Variations on Cover Inequalities

• Extended Cover Inequalities
• Lifted Cover Inequalities
Lifted Cover Inequalities

• $X = \{x \in \{0,1\}^7 \mid 11x_1 + 6x_2 + 6x_3 + 5x_4 + 5x_5 + 4x_6 + x_7 \leq 19\}$

• **Lifting**: Given $x_2 + x_3 + x_4 + x_5 + x_6 \leq 3$, find a valid inequality $\alpha_1 x_1 + x_2 + x_3 + x_4 + x_5 + x_6 \leq 3$, for some $\alpha_1 > 0$.

• Valid for any soln with $x_1 = 0$.

• What about solns with $x_1 = 1$? Well, any $x \in X$ with $x_1 = 1$ satisfies: $6x_2 + 6x_3 + 5x_4 + 5x_5 + 4x_6 + x_7 \leq 8$ (*)

• **Idea**: pick maximum $\alpha_1$ s.t. $\alpha_1 + x_2 + x_3 + x_4 + x_5 + x_6 \leq 3$ for all integer solutions satisfying (*)

• Set $\alpha_1 = 3 - \max_x \{ x_2 + x_3 + x_4 + x_5 + x_6 : 6x_2 + 6x_3 + 5x_4 + 5x_5 + 4x_6 + x_7 \leq 8, x \in \{0,1\}^6 \} = 3 - 1 = 2$

• **Lifted cover inequal**: $2x_1 + x_2 + x_3 + x_4 + x_5 + x_6 \leq 3$

• **Theorem.** Lifted cover inequalities are “facet-defining” for $\text{conv}(X)$ when $X$ is $\{0,1\}$-knapsack set.

• (Informal) An inequality is **facet defining** if it defines the face of a polyhedron. Consider:

---

![Diagram showing facet defining and non-facet defining inequalities]
Lifted Cover Inequalities

• Complete for IPs with constraints that are 0-1 knapsack sets.
• But, rather than use them to construct the convex hull we will do something else…

• Let’s see this next. How to use lifted cover inequalities and other approaches?

• We’ve seen two families of cuts:
  – Chvátal-Gomory cuts for IPs with integer coefficients and integer RHS values
  – (Lifted, extended) cover cuts for IPs with \{0,1\} vars and strictly positive coeffs and strictly positive RHS values
• There are many others:
  – Flow cover
  – Mixed integer rounding
  – Odd hole
  – …
• How can we use these?
The Branch-and-Cut method

- Generalizes Branch-and-Bound.
- If cannot prune a node, can try to find a cut.
- Strengthen the formulation; re-solve the LP, may now be able to prune by bound.
- If no cuts are found, then branch.

The Branch-and-Cut method

- Branch-and-Bound + cuts generated throughout the tree.
- Cuts generated at root are “global,” cuts at subproblems “local.” Place into a “cut pool.”
- Goal: reduce the number of search nodes by improving bounds
Example Application

• Winner determination problem in an auction, e.g. selling lots of wine.
• 10 bidders, 5 different wines (≈100 cases each).
• Bidders want either wine 1, or wine 2, or wine 3…
• Goal: maximize total revenue

<table>
<thead>
<tr>
<th>Bidders</th>
<th>Goods</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A</td>
</tr>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>demand</td>
<td>91</td>
</tr>
<tr>
<td>value</td>
<td>110</td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>demand</td>
<td>54</td>
</tr>
<tr>
<td>value</td>
<td>65</td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>demand</td>
<td>3</td>
</tr>
<tr>
<td>value</td>
<td>19</td>
</tr>
</tbody>
</table>

Supply

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>91</td>
<td>87</td>
<td>109</td>
<td>88</td>
<td>64</td>
<td></td>
</tr>
</tbody>
</table>
Winner Determination Problem

- Bidders i, goods j with supply b_j
- c_{ij} value to i for a_{ij} units of good j
- x_{ij}=1 if select bid from i on j as a winner

\[
\begin{align*}
\text{max } & \sum_i \sum_j c_{ij} x_{ij} \\
\text{s.t. } & \sum_j x_{ij} \leq 1 \text{ for all } i = 1,\ldots,m \\
& \sum_i a_{ij}x_{ij} \leq b_j \text{ for all } j = 1,\ldots,n \\
& x_{ij} \in \{0,1\}
\end{align*}
\]

Has “0/1 knapsack set” structure. Can use Cover cuts.

Example Search Tree

- 14 lifted cover cuts generated. LP bound improves from 595.6 to 554.1
- 1 cut generated
- 3 cuts generated. LP integral. Fathom by integrality.
- BnB without cuts requires 4206 search nodes.
- Branch-and-cut: With even more cut generation at root, bound improves to 546 and tree has 3 nodes!
Historical Context

• Use of cutting planes as a practical tool for solving IPs was abandoned in 1960s and ‘70s!!!
• Revisited in early ‘70s following NP-completeness theory (interest in problem structure and facets.)
• Balas, Wolsey, Johnson, Padberg developed cover inequalities and “lifting” in mid ‘70s.
• Branch-and-cut developed in early ‘80s and introduced into commercial solvers in ‘00s.
• Provides order of magnitude improvement in solve times.

Other important components

• Preprocessing
• Primal heuristics
• SOS branching
Preprocessing

• Detect and eliminate redundant constraints and variables, tighten bounds where possible

Preproc.: Isolating and Fixing

• Consider LP: (Wolsey pp.104)
• \( \text{max } 2x_1 + x_2 - x_3 \)
  \[ \begin{align*}
  5x_1 - 2x_2 + 8x_3 & \leq 15 \quad (a) \\
  8x_1 + 3x_2 - x_3 & \geq 9 \quad (b) \\
  x_1 + x_2 + x_3 & \leq 6 \quad (c) \\
  0 \leq x_1 \leq 3; \quad 0 \leq x_2 \leq 1; \quad 1 \leq x_3 
  \end{align*} \]
• Isolate \( x_1 \) in (a) to obtain \( 5x_1 \leq 15 + 2x_2 - 8x_3 \leq 15 + 2 - 8 = 9 \), and so \( x_1 \leq 9/5 \).
• Isolate \( x_3 \) in (a) to obtain \( x_3 \leq 17/8 \). Not useful to isolate \( x_2 \).
• Can also isolate \( x_1 \) in (b); obtain \( x_1 \geq 7/8 \).
• Repeat: improve bound on \( x_3 \) from (a) by using \( x_1 \geq 7/8 \); obtain \( x_3 \leq 101/64 \)
• Find constraint (c) is redundant, using \( x_1 \leq 9/5 \) and
  \( x_3 \leq 101/64 \), together with \( x_2 \leq 1 \).
“Fixing” values of variables

The modified LP is:
• max $2x_1 + x_2 - x_3$
  s.t. $5x_1 - 2x_2 + 8x_3 \leq 15$
  $8x_1 + 3x_2 - x_3 \geq 9$
  $7/8 \leq x_1 \leq 9/5; 0 \leq x_2 \leq 1; 1 \leq x_3 \leq 101/64$

• **Fixing**: In optimal solution, have $x_2 = 1$; and $x_3 = 1$.
• Why? $x_2$ has positive coeff in obj, and increasing makes constraints less tight. Similarly, best to decrease $x_3$.

• Obtain: max{$2x_1 : 7/8 \leq x_1 \leq 9/5$}. Easy to solve!

Other important components

• Preprocessing
• Primal heuristics
• SOS branching
Primal heuristics

• At a fractional search node, use a heuristic approach to try to find an integral solution
• For example, round the solution or do a directed “neighborhood search.”

• May be able to improve incumbent.

Other important components

• Preprocessing
• Primal heuristics
• SOS branching
SOS branching

• Consider constraints $\sum_{j=1}^{n} x_j = 1$, $x_j \in \{0,1\}$
• Typical to branch on fractional $x_k^* :$

- **unbalanced**
  - $x_k \leq 0$
  - $x_k \geq 1$
  - n-1 solutions
  - 1 solution

- **balanced**
  - $\sum_{j=1}^{k} x_j \leq 0$
  - $\sum_{j=1}^{k} x_j \geq 1$

"Special-ordered sets" branching. Pick smallest $k$ s.t. $\sum_{j=1}^{k} x_j^* \geq \frac{1}{2}$

Example: $x^* = (0,0.2,0.4,0,0.4)$. Branch $x_1 + x_2 + x_3 \leq 0$, $x_1 + x_2 + x_3 \geq 1$

Summary: Advanced IP solving

• MIPs are a lot harder to solve than LPs!
• But IP technology is sophisticated:
  - Dual simplex pivots
  - Tight formulations
  - Automated cut generation
  - Primal heuristics
  - Node and variable selection heuristics
• Problems with 100,000s variables and 10,000s constraints can be solved in commercially viable times.