Dual All-Integer Programming

Kwei-Long Huang

Agenda

- Basic Approach
- The Form of the Cut
- The Derivation
  - The Cut
  - The Pivot Column
  - The λ Selection
- Properties of the Added Cuts
- Algorithm Strategies
- Finiteness

Dual All-Integer Programming

- This cutting plane algorithm for ILP problems was developed by Gomory in 1960.
- This method is a direct extension of the dual simplex method. It starts with an all-integer tableau and maintains the integrality of the tableaus in subsequent iterations.

Basic Procedures

- 1. Start with an all-integer simplex tableau which contains a lexicographic dual feasible solution.
- 2. Select a primal infeasible row (i.e., $a_{v,0} < 0$). If none exists, the current basic solution is optimal integer and terminate.
- 3. Select the pivot column α_k to be the lexicographically smallest column having $a_{vj} < 0$. If none exist, there is no integer feasible solution and terminate.
- 4. Derive an all-integer inequality from row *v* which is not satisfied at the current primal solution and has a **-1** coefficient in column α_p. Append it to the bottom of the current tableau and label it the pivot row.
- 5. Perform a dual simplex pivot iteration to obtain an updated all-integer tableau and go back to step 2.

The Form of the Cut

- Let v be the generating row with $a_{v,0} < 0$,

$$x_v = a_{v,0} + \sum_{j=1}^{n} a_{v,j} \left(-x_{J(j)} \right).$$

- Then, the all-integer cut is

$$x' = \left\lfloor \frac{a_{v,0}}{\lambda} \right\rfloor + \sum_{j=1}^{n} \left\lfloor \frac{a_{v,j}}{\lambda} \right\rfloor \left(-x_{J(j)} \right).$$

- Where x' is the Gomory slack variable and λ is a positive number computed as follows:

The Form of the Cut (con't)

- Let α_p be the lexicographically smallest column with $a_{v,j} < 0$, $j = 1, \ldots, n$.
- Let $u_p = 1$ and, for $a_{v,j} < 0$, $j = 1, \ldots, n$ and $j \neq p$, let u_j be the largest integer number such that

$$\alpha_p \prec \left(\frac{1}{u_j}\right)\alpha_j. \text{ Note that } u_j \geq 1 \text{ and } \alpha_p \succ 0.$$

- For $a_{v,j} < 0$, $j=1, \ldots, n$, compute $\lambda_j = -\frac{a_{vj}}{u_j}$.
- Let $\lambda = \max\{\lambda_j\}$. Note that $\lambda \geq \lambda_p = -\frac{a_{vp}}{u_p} \geq 1$.

Example (1/4)

$$
\begin{array}{llr}
\text{Maximize} & -4x_1 - 5x_2 = x_0 & \\
\text{subject to} & -x_1 - 4x_2 \le -5, & (x_3) \\
 & -3x_1 - 2x_2 \le -7, & (x_4) \\
\text{and} & x_1,\ \ x_2 \ \ge 0, & \text{integer.}
\end{array}
$$

Example (2/4)

#1	1	$(-x_1)$	$(-x_2)$
x_0	0	4	5
x_1	0	-1	0
x_2	0	0	-1
x_3	-5	-1	-4
$\rightarrow x_4$	-7	-3	-2

$(x_1 = x_2 = 0)$
$p = 1$
$u_1 = 1, u_2 = 1$
$\lambda_1 = 3, \lambda_2 = 2$
$\lambda = \max(\lambda_1, \lambda_2) = 3$

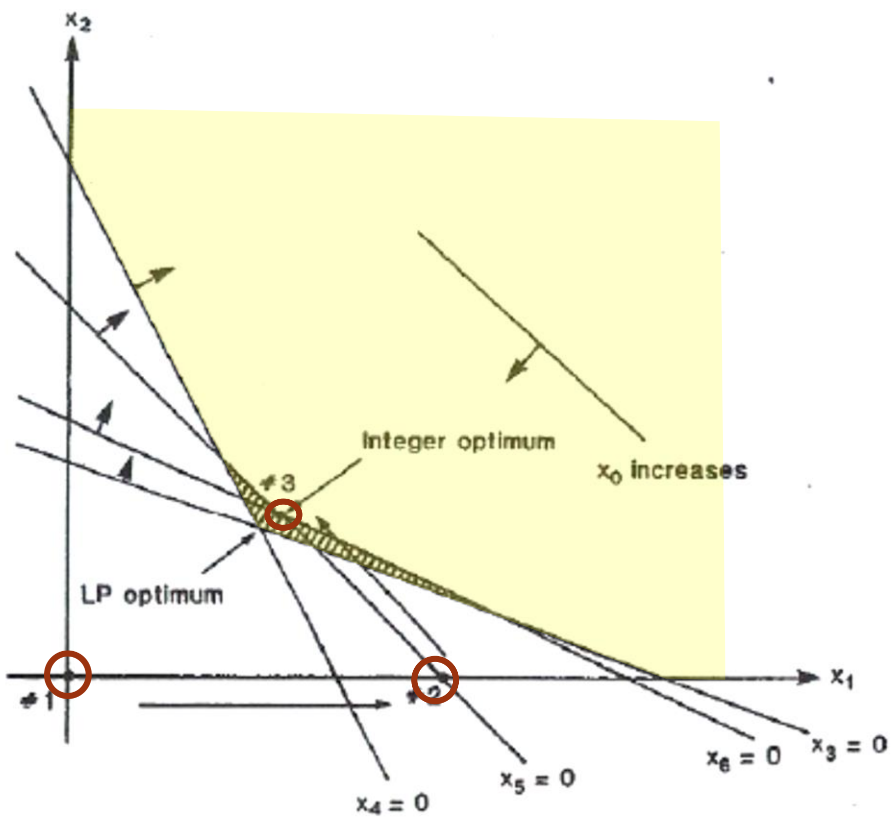
Example (3/4)

#2	1	$(-x_5)$	$(-x_2)$
x_0	-12	4	1
x_1	3	-1	1
x_2	0	0	-1
x_3	-2	-1	-3
x_4	2	-3	1
x_5	0	-1	0

$(x_1 = 3, x_2 = 0)$
$p = 2$
$u_1 = 3, u_2 = 1$
$\lambda_1 = {}^1/_3, \lambda_2 = 3$
$\lambda = \max(\lambda_1, \lambda_2) = 3$

Example (4/4)

#3	1	$(-x_5)$	$(-x_6)$
x_0	-13	3	1
x_1	2	-2	1
x_2	1	1	-1
x_3	1	2	-3
x_4	1	-4	1
x_5	0	-1	0
x_6	0	0	-1



Example 2 (1/2)

$$\begin{array}{llr}
\text{Maximize} & -10x_1 - 14x_2 - 21x_3 = x_0 & \\
\text{subject to} & 8x_1 + 11x_2 + 9x_3 \geq 12, & (x_4) \\
 & 2x_1 + 2x_2 + 7x_3 \geq 14, & (x_5) \\
 & 9x_1 + 6x_2 + 3x_3 \geq 10, & (x_6) \\
\text{and} & x_1,\ x_2,\ x_3 \geq 0, \quad \text{integer.} &
\end{array}$$

Example 2 (2/2)

- If set λ=2 => LDS not satisfied.

#1	1	$(-x_1)$	$(-x_2)$	$(-x_3)$
x_0	0	10	14	21
x_1	0	-1	0	0
x_2	0	0	-1	0
x_3	0	0	0	-1
x_4	-12	-8	-11	-9
x_5	-14	-2	-2	-7
x_6	-10	-9	-6	-3

$p = 1$

$u_1 = 1$

$u_2 = 1, u_3 = 2$

$\lambda_1 = \left({}^{2}/_{u_1}\right) = 2$

$\lambda_2 = \left({}^{2}/_{u_2}\right) = 2$

$\lambda_3 = \left({}^{7}/_{u_3}\right) = {}^{7}/_{2}$

$\lambda = {}^{7}/_{2}$

Agenda

- Basic Approach
- The Form of the Cut
- The Derivation
  - The Cut
  - The Pivot Column
  - The λ Selection
- Properties of the Added Cuts
- Algorithm Strategies
- Finiteness

The Derivation of the Cut (1/6)

- First we need to show that the pivot element is always -1.

Consider a source row

$$x = a_0 + \sum_{j=1}^{n} a_j \left(-x_{J(j)} \right),$$

with $a_0 < 0.$

The Derivation of the Cut (2/6)

Let λ be a positive number. Then,

$$f = \frac{a}{\lambda} - \left\lfloor \frac{a}{\lambda} \right\rfloor ;$$

then $0 \le f < 1$. Defining $r = \lambda f$ yields

$$\frac{r}{\lambda} = \frac{a}{\lambda} - \left\lfloor \frac{a}{\lambda} \right\rfloor , \quad \text{or} \quad a = \left\lfloor \frac{a}{\lambda} \right\rfloor \lambda + r$$

where $0 \le r < \lambda$

The Derivation of the Cut (3/6)

- Rewrite the coefficients in source row

$$a_j = \left\lfloor \frac{a_j}{\lambda} \right\rfloor \lambda + r_j,\ j = 1, \ldots, n,$$

and $1 =$

where $\lambda > 0$, $0 \le r_j < \lambda$, and $0 \le r_0 < \lambda$.

The Derivation of the Cut (4/6)

- Rewrite the source row

$$\left(\left\lfloor\frac{1}{\lambda}\right\rfloor\lambda+r\right)x=\left(\left\lfloor\frac{a_0}{\lambda}\right\rfloor\lambda+r_0\right)+\sum_{j=1}^{n}\left(\left\lfloor\frac{a_j}{\lambda}\right\rfloor\lambda+r_j\right)(-x_{J(j)})$$

$$\Rightarrow \sum_{j=1}^{n} r_j x_{J(j)}+rx=r_0+\lambda\left\{\left\lfloor\frac{a_0}{\lambda}\right\rfloor+\right.$$

$$\boxed{\boxed{x'=\left\lfloor\frac{a_0}{\lambda}\right\rfloor+\sum_{j=1}^{n}\left\lfloor\frac{a_j}{\lambda}\right\rfloor\left(-x_{J(j)}\right)+\left\lfloor\frac{1}{\lambda}\right\rfloor(-x).}}$$

The Derivation of the Cut (5/6)

- We need to show x' is non-negative

$\sum_{j=1}^{n} r_j x_{J(j)} + rx$ is non-negative and $0 \leq r_0 < \lambda$.

$$r_0 + \lambda x' \geq 0$$

In addition, x' is integer. Thus, x' is non-negative.

The Derivation of the Cut (6/6)

- It was shown in the introduction of the cut that $\lambda \geq 1$. If λ turns out to be 1, the source row is the pivot row and a new inequality is not added.

$$x^{'} = \left\lfloor \frac{a_0}{\lambda} \right\rfloor + \sum_{j=1}^{n} \left\lfloor \frac{a_j}{\lambda} \right\rfloor \left(-x_{J(j)} \right) \geq 0.$$

Since $a_0 < 0$ and $\left\lfloor \frac{a_0}{\lambda} \right\rfloor < 0$, thus it may be appended to the bottom of the tableau and used as pivot row. Furthermore, for λ sufficiently large the pivot element will be -1.

The Pivot Column (1/2)

- The pivot column for the LDS method satisfies

$$\frac{\alpha_p}{-b_p} \prec \frac{\alpha_j}{-b_j}$$

- Where b_j, $j=1,\ldots,n$ are the nonbasic coefficients in the pivot row for the all-integer algorithm. Thus,

$$b_p = -1$$

$$b_j = \left\lfloor \frac{a_j}{\lambda} \right\rfloor \quad \Rightarrow \quad \alpha_p \prec \frac{\alpha_j}{-\left\lfloor \frac{a_j}{\lambda} \right\rfloor}, \; j = 1,\ldots,n, \; j \neq p, \frac{a_j}{\lambda} < 0.$$

The Pivot Column (2/2)

$$\alpha_p \prec \frac{\alpha_j}{-\left\lfloor \frac{a_j}{\lambda} \right\rfloor} \prec \alpha_j,\ j = 1, \ldots, n,\ j \neq p, \frac{a_j}{\lambda} < 0.$$

=> The pivot column is the lexicographically smallest column with a negative element in the pivot row.

The λ Selection (1/2)

- The objective is to choose λ so that
  - The pivot element is -1.
  - It produces the greatest lexicographic decrease is column 0 (α_0). The updated column 0 becomes

$$\alpha_0' = \alpha_0 + \left\lfloor \frac{\alpha_0}{\lambda} \right\rfloor \alpha_p'.$$

Since $\left\lfloor \frac{\alpha_0}{\lambda} \right\rfloor$ is a negative integer, $\left\lfloor \frac{\alpha_0}{\lambda} \right\rfloor$ decreases as λ decreases.

Then, we must choose the smallest possible λ.

$$\alpha_p \prec \frac{\alpha_j}{-\left\lfloor \frac{\alpha_j}{\lambda} \right\rfloor}$$

The λ Selection (2/2)

- Let $u_p = 1$ and let u_j, $j \neq p$ be the largest integer such that

$$\alpha_p \prec \frac{\alpha_j}{u_j}$$

$$\Rightarrow -\left\lfloor \frac{\alpha_j}{\lambda} \right\rfloor \le u_j.$$

- Then, the smallest satisfying the above inequality is

$$\lambda = \max\{\lambda_j\}, \text{ where } \lambda_j = -\frac{a_j}{u_j}.$$

Agenda

- Basic Approach
- The Form of the Cut
- The Derivation
  - The Cut
  - The Pivot Column
  - The λ Selection
- Properties of the Added Cuts
- Algorithm Strategies
- Finiteness

Stronger Cut

- Source row

$$x = -4 - 3(-x_1) - 5(-x_2) \geq 0$$

- $\lambda = 2$

$$x' = -2 - 2(-x_1) - 3(-x_2) \geq 0$$

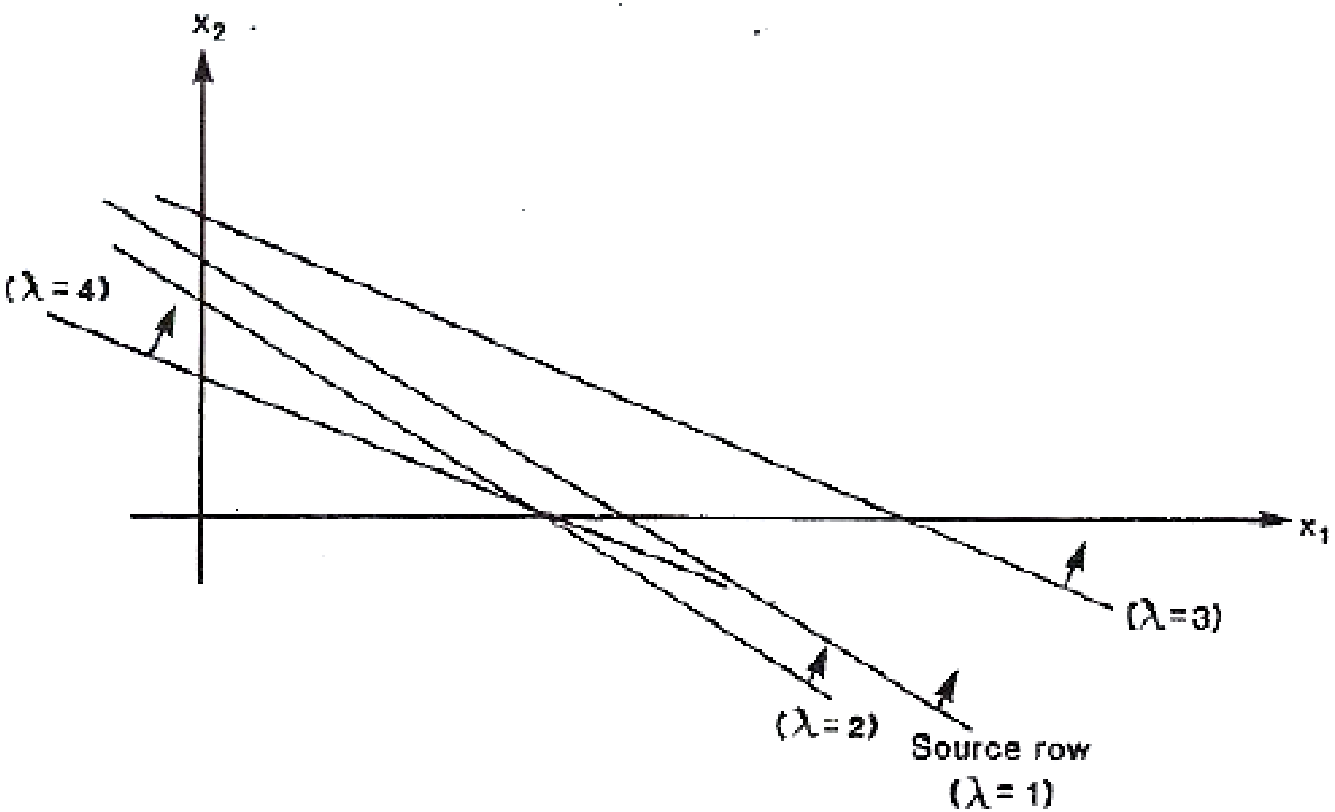
- $\lambda = 3$

$$x' = -2 - 1(-x_1) - 2(-x_2) \geq 0$$

- $\lambda = 4$

$$x' = -1 - 1(-x_1) - 2(-x_2) \geq 0$$

Strength of the Inequalities



Derive a Stronger Cut

- If increase λ so that $\left\lfloor {a_0}/{\lambda} \right\rfloor$ and $\left\lfloor {a_j}/{\lambda} \right\rfloor$, for $a_j > 0$, remain the same, a stronger inequality can be obtained when some or all $\left\lfloor {a_j}/{\lambda} \right\rfloor$ ($a_j < 0$) change.

- Since $x_{J(j)} = \left\lfloor {a_0}/{\lambda} \right\rfloor \Big/ \left\lfloor {a_j}/{\lambda} \right\rfloor$ increases as λ increases. (the numerator remains the same.)

- Also, the zero column has __ ___ amount of change.

Example

- Consider a partial tableau where *x* is the source row.

	1	$-x_1$	$-x_2$	$-x_3$	$-x_4$
x_0	20	1	3	4	4
x	-20	-7	-8	-15	18

- The Gomory cut

$$x' = -3 - 1(-x_1) - 2(-x_2) - 3(-x_3) + 2(-x_4) \geq 0 \qquad (\lambda=7)$$

- What is the possible increment of λ?
- The stronger cut

$$x' =$$

Gomory Fractional Cut

- Note that when λ=1

$$x' = \left\lfloor \frac{a_0}{\lambda} \right\rfloor + \sum_{j=1}^{n} \left\lfloor \frac{a_j}{\lambda} \right\rfloor \left(-x_{J(j)}\right) + \left\lfloor \frac{1}{\lambda} \right\rfloor (-x),$$

where the source row is

$$x = a_0 + \sum_{j=1}^{n} a_j \left(-x_{J(j)}\right);$$

Agenda

- Basic Approach
- The Form of the Cut
- The Derivation
  - The Cut
  - The Pivot Column
  - The λ Selection
- Properties of the Added Cuts
- Algorithm Strategies
- Finiteness

Choose and Drop

- Selection rule for finite convergence: if the constant term of a row is negative and remains negative, the row will eventually be selected.

- Examples of rules that satisfy the above condition:
  - Select the first row with a negative constant term.
  - Select a row with a negative constant term.
  - Randomly select a row with a negative constant term.

- Drop derived cuts when their slack variable becomes basic. This rule limits the size of the tableau.

Agenda

- Basic Approach
- The Form of the Cut
- The Derivation
  - The Cut
  - The Pivot Column
  - The λ Selection
- Properties of the Added Cuts
- Algorithm Strategies
- Finiteness

Finiteness Proof (1/3)

- To show finiteness x_0 must be bounded from below.
- The proof is similar to that of the dual fractional ILP algorithm. That is, we assume the algorithm is not finite. Then, using rule 1 for row selection, there must exist an infinite sequence of tableaus such that $\alpha_0^k \succ \alpha_0^{k+1} \succ \alpha_0^{k+2} \succ \ldots$
- Since each component of α_0 is integer, its first component $a_{0,0}$ must decrease by integer amounts until it remains fixed.
- Then, the second component of α_0, $a_{1,0}$, must also decrease by integer amounts until it remains fixed to a nonnegative integer. Eventually, all components of α_0 must remain fixed.

Finiteness Proof (2/3)

- Using contradiction, it can be shown that, if $a_{0,0}$, $a_{1,0}, \dots$, $a_{v\text{-}1,0}$ remain fixed above their lower bounds, then $a_{v,0}$ cannot become negative.

- Suppose $a_{v,0}$ falls below 0; then row v becomes an eligible source row. Using rule 1, row v is selected to generate the cut and the pivot row becomes

$$x' = \left\lfloor \frac{a_{v,0}}{\lambda} \right\rfloor + \sum_{j=1}^{n} \left\lfloor \frac{a_{v,j}}{\lambda} \right\rfloor \left(-x_{J(j)} \right) > 0,$$

$$\text{where } \left\lfloor \frac{a_{v,0}}{\lambda} \right\rfloor < 0.$$

Finiteness Proof (3/3)

- If the problem is feasible, there exists an index p such that $a_{v,p}<0$.
- Then ,choose λ so that the pivot $\left\lfloor \frac{a_{v,p}}{\lambda} \right\rfloor = \boxed{}$
- In this case, $a_{0,0}$ will to

$$a'_{0,0} = a_{0,0} + a_{0,p}\left\lfloor \frac{a_{v,0}}{\lambda} \right\rfloor, \text{where } a_{0,p} > 0 \text{ and } \left\lfloor \frac{a_{v,0}}{\lambda} \right\rfloor < 0,$$

which is a contradiction.

- Thus, $a_{v,0}$ must eventually remains fixed at a nonnegative integer.

Reminder

- Midterm: 9:20~11:50 on 4/20
- One page cheat sheet in A4-size
- No laptops or smart phones. (Calculator is allowed.)
- Final project
  - Form a group with 5~6 students.
  - Submit one page to describe your project on May 4.
  - The project can be a literature review for an application area or methodology, a research problem, or a computational study.