Integer and Combinatorial Optimization
整數與組合最佳化

Instructor: Kwei-Long Huang 黃奎隆
國立臺灣大學
工業工程研究所

Email: craighuang@ntu.edu.tw

Textbook

- Textbook
  - H.M. Salkin and K. Mathur, *Foundations of Integer Programming*, North-Holland, New York, 1989
- Reference
  - D.-S. Chen, R. G. Batson, Y. Dang, *Applied Integer Programming: Modeling and Solution*, Wiley 2010
  - L.A. Wolsey, *Integer Programming*, Wiley 1998.
  - G.L. Nemhauser and L.A. Wolsey, *Integer and Combinatorial Optimization*, Wiley 1988.

Outlines

- Introduction
- Integer programming problems
- Integer programming modeling
- Cutting planes

What is an integer program?

- An integer linear program is a mathematical (optimization) model where the objective function and constraints are linear, and all the variables or a subset of the variables are restricted to take integer values.
- The theorems and solution approaches (e.g., simplex method) for linear programming models can not directly apply to an integer linear programming model.

Integer Program

If all variables are integer: we have an Integer Program (IP):

$$\max_{x \in R^n} \{c^T x : Ax \le b, x \ge 0 \text{ and integer}\}.$$

$$\begin{aligned}
&\text{maximize } z = \sum_{j=1}^{n} c_j x_j \\
&\text{subject to} \quad \sum_{j=1}^{n} a_{ij} x_j \le b_i, \qquad i = 1, \ldots, m, \\
&\qquad\qquad\quad x_j \ge 0 \text{ , integer}, \qquad j = 1, \ldots, n.
\end{aligned}$$

Mixed Integer Program

Some variables are integer and the rest are continuous and we have an Mixed (Linear) Integer Program (MIP/MILP).

$$
\begin{aligned}
\text{maximize } z &= \sum_{j=1}^{n} c_j x_j + \sum_{k=1}^{p} d_k y_k \\
\text{subject to } & \sum_{j=1}^{n} a_{ij} x_j + \sum_{k=1}^{p} g_{ik} y_k \le b_i, & i &= 1, \ldots, m, \\
& x_j \ge 0 \text{ , integer ,} & j &= 1, \ldots, n, \\
& y_k \ge 0 \text{ ,} & k &= 1, \ldots, p.
\end{aligned}
$$

Binary Integer Program

If each variable is only on values 0 or 1, i.e., all variables are binary integer: we have a Binary Integer Program (BIP):

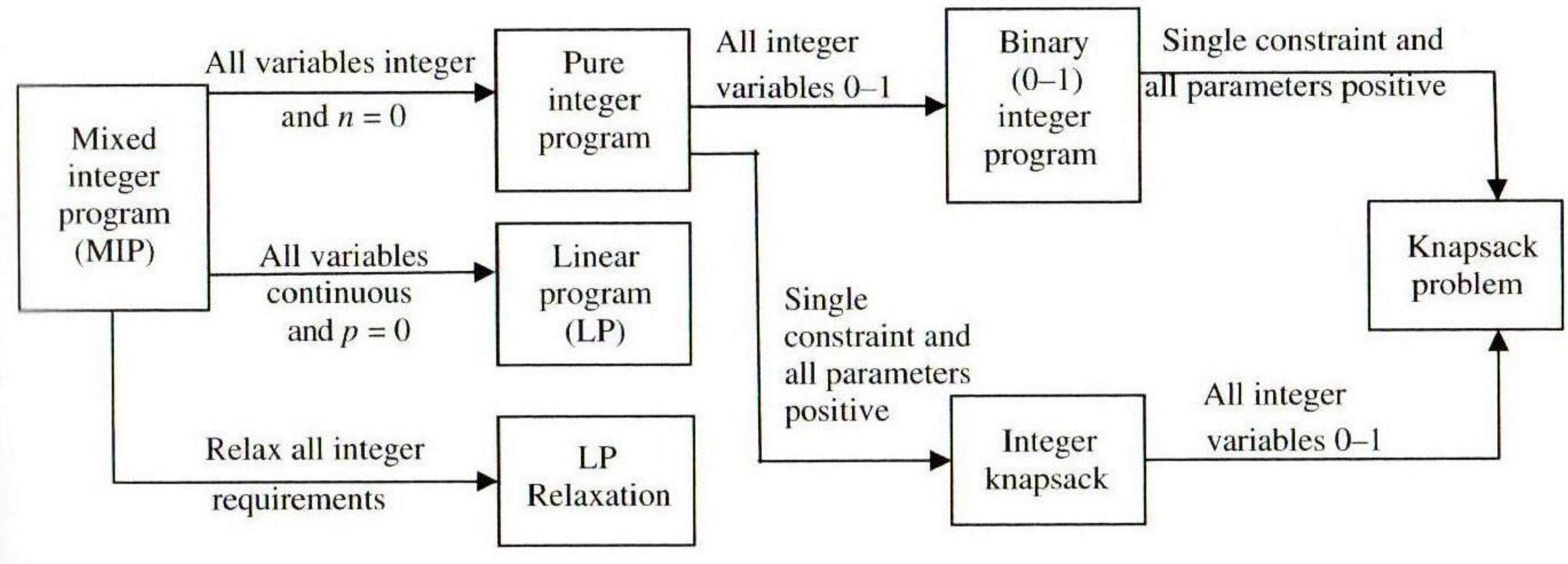
$$\underset{x \in R^n}{\text{Max}}\{c^T x : Ax \le b, x \in \{0,1\}\}.$$

$$\text{maximize } z = \sum_{j=1}^{n} c_j x_j$$

$$\text{subject to} \quad \sum_{j=1}^{n} a_{ij} x_j \le b_i, \qquad i = 1, \ldots, m,$$

$$x_j \in \{0,1\}, \qquad j = 1, \ldots, n.$$

A Simple Classification of Integer Programs



n: number of continuous variables
p: number of integer variables

Source: Applied Integer Programming: Modeling and Solution, WILEY

ILP and MILP

- Some ILP and MILP models have all integer extreme points e.g., assignment problems, transportation problems with integer supplies and demands, min-cost network flow problems with integer supplies and demands, etc. In these cases, there always exist extreme points that are optimal and linear programming (LP) techniques can be used to find optimal solutions.

Example: Quality Furniture Corporation

The Quality Furniture Corporation manufactures two products: **benches** and **tables**. They employ **three** carpenters. During the next week, 150 hours of labor are available at $8 per hour.

Material and Resource

- 300 pounds of wood is available at a cost of $5 per pound.
- Each bench requires 3 labor hours and 12 pounds of wood. Each table requires 6 labor hours and 38 pounds of wood.
- Completed benches sell for $80 each, and tables sell for $200 each.

Quantity Discount

- Up to 150 pounds of wood can be purchased at \$4 per pound when 300 lb of wood has been purchased.
- Up to 100 pounds of wood can be purchased at \$3.5 per pound when 450 lb of wood has been purchased.

Question: How many benches and how many tables should be produced?

Linear Model

$$\text{Max } 85x_1 + 200x_2 - 8(3x_1 + 6x_2) - 5w_1 - 4w_2 - 3.5w_3$$

Where

x_1 is the number of benches produced,

x_2 is the number of tables produced,

w_1 is the number of wood purchased at \$5,

w_2 is the number of wood purchased at \$4,

w_3 is the number of wood purchased at \$3.5.

Linear Model (con't)

- Labor:
  - $3x_1 + 6x_2 \leq 150,\ x_1,\ x_2 \geq 0$
- Wood:
  - $12x_1 + 38x_2 \leq w_1 + w_2 + w_3$
  - $0 \leq w_1 \leq 300$
  - $0 \leq w_2 \leq 150$
  - $0 \leq w_3 \leq 100$

Linear Model (con't)

- Discount constraints:
  - Only when $w_1=300$, w_2 can be ≥ 0.
  - Only when $w_2=150$, w_3 can be ≥ 0.
  - y_1, y_2, and y_3 are binary

Linear Model (con't)

Max $85x_1 + 200x_2 - 8(3x_1 + 6x_2) - 5w_1 - 4w_2 - 3.5w_3$

S.t.

$3x_1 + 6x_2 \leq 150,\ x_1,\ x_2 \geq 0$

$12x_1 + 38x_2 \leq w_1 + w_2 + w_3$

$w_1 \leq 300y_1,\ w_1 \geq 300y_2$

$w_2 \leq 150y_2,\ w_2 \geq 150y_3$

$w_3 \leq 100y_3,$

$w_1,\ w_2,\ w_3 \geq 0$

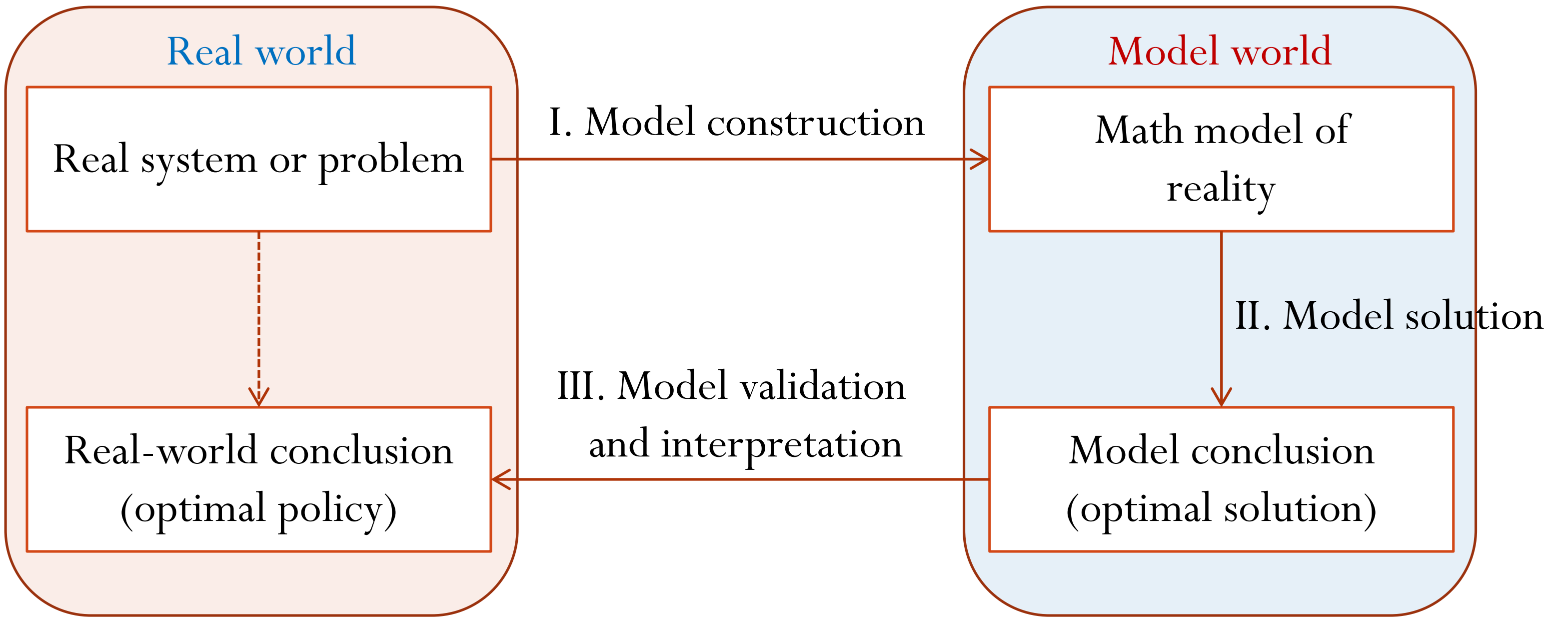
y_1, y_2, and y_3 are binary

Outlines

- Introduction
- Integer programming problems
- Integer programming modeling
- Cutting planes

Modeling Process

- Construct of the model
- Solution of the model
- Validation and interpretation



Source: Applied Integer Programming: Modeling and Solution, WILEY

Model Construction Process

- Step 1: verbally identify and define decision variables, parameters, constraints and the objective from the problem description. Assign appropriate symbols.
- Step 2: translate the verbal description into functions, equations, and inequalities.
- Step 3: check whether the non-MIP factors can be transformed into equivalent mathematical expressions. If yes, an MIP is obtained.

Project Selection

- A single time period
  - Knapsack problem (Cargo loading problem)
- Multiple time periods
  - Capital budgeting problem

0-1 Knapsack Problem

- Suppose that a plane has cargo weight capacity b and is to be loaded with items each with weight a_j and relative value c_j.
- The problem is to load the plane so as to maximize its total relative value.

$$\begin{aligned} &\text{Maximize} \quad && z = \sum_j c_j y_j \\ &\text{subject to} \quad && \sum_j a_j y_j \le b \\ & && y_j = 0 \text{ or } 1 \qquad j = 1, 2, \ldots, n \end{aligned}$$

Capital Budgeting Problem

- Assume project j has a present value of c_j dollars and requires an investment of a_{tj} dollars in time period t (t=1,…, T). The capital available in time period t is b_t dollars.
- The objective of this problem is to maximize the total present value subject to the budgetary constraint in each time period over a prescribed planning horizon T.

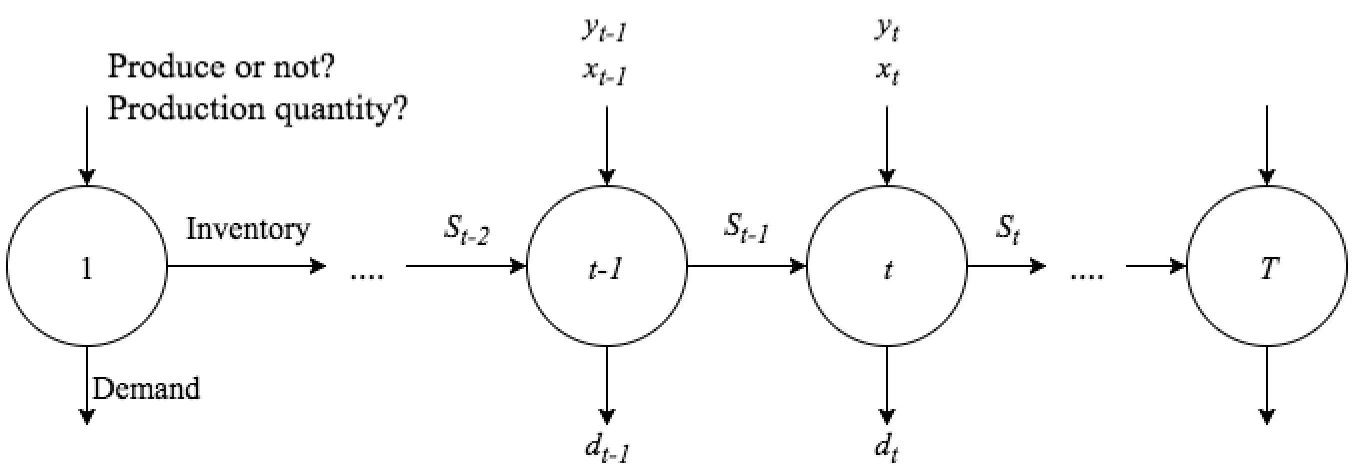
Maximize $\quad z = \sum_j c_j y_j$

subject to $\qquad\qquad t = 1, \ldots, T$

$y_j = 0 \text{ or } 1 \qquad j = 1, 2, \ldots, n$

Production Planning Problems

- Uncapacitated production size
- Capacitated production size
- Just-in-Time production planning



Source: Applied Integer Programming: Modeling and Solution, WILEY

Uncapacitated Production Size (1/2)

Input parameters:	number of periods (T), demand in each period (d_t), setup cost for each period (f_t), unit production cost (c_t), unit holding cost (h_t)
Decision variables:	whether or not to produce in each time period $(y_t = 1$ or $0)$ and how much if the decision is to produce (x_t)
Constraints:	satisfy the demand in each period t
State variables:	inventory level at the end of each period (S_t), assuming the beginning inventory level $s_0 = 0$
Objective:	minimize the total production and inventory costs

Uncapacitated Production Size (2/2)

Let M be a "sufficiently" large number (say, M ______). Note that $y_t = 1$ if and only if ______. The problem can be formulated as follows: Find values of x_t and y_t $(t = 1, \ldots, T)$ so as to

Minimize $\sum_t (c_t x_t + f_t y_t + h_t s_t)$

subject to for all t

for all t

$x_t \geq 0$ for all t

$s_t \geq 0$ $t = 0, 1, \ldots, T$

$y_t = 0$ or 1 for all t

Capacitated Production Sizing

- Each facility has its own capacity limitation (demoted as u).
- We simply replace the ______ in the uncapacitated production size model with a capacity upper limit u.

$$
\begin{array}{lll}
\text{Minimize} & \sum_t (c_t x_t + f_t y_t + h_t s_t) & \\
\text{subject to} & s_{t-1} + x_t - s_t = d_t & \text{for all } t \\
 & & \text{for all } t \\
 & x_t \geq 0 & \text{for all } t \\
 & s_t \geq 0 & \text{for all } t \\
 & y_t = 0 \text{ or } 1 & \text{for all } t
\end{array}
$$

Just-in-Time Production Planning (1/4)

- Multiple products.
- This type of planning seeks to determine a production level for each product in each time period with the right quantity at the right time.
- Try to maintain zero inventory level.
- Penalty on shortage or surplus.

Just-in-Time Production Planning (2/4)

Input parameters:	number of product types (n), number of periods (T), demand of product j in each period (d_{jt}), prescribed production lot size for each product (l_{jt}), unit penalty of earliness (p_j), unit penalty of lateness (q_j)
Decision variables:	production level of each product in each period $(x_{jt} \geq 0)$, number of production runs in each period t for each product (y_{jt})
Constraints:	satisfy demand of each product j in each period and constraints relating to prescribed lot size, number of production runs per period, and production level
State variables:	surplus and shortage inventory levels for each product in each time period $(d_{jt}^{+}$ and $d_{jt}^{-})$, ending inventory level of each product (s_{jt})
Objective:	minimize total penalty cost of all products due to earliness and lateness over all periods

Just-in-Time Production Planning (3/4)

- Recall the inventory balancing equation that relates the beginning inventory level, production level, demand level, and the ending level given below:

$$s_{j,t-1} + x_{jt} - d_{jt} = s_{jt} \qquad \text{for all } j,t$$

$$\text{or} \quad s_{j,t-1} + x_{jt} - s_{jt} = d_{jt}$$

- Let d_{jt}^{+} and d_{jt}^{-}, respectively, be a nonnegative amount of surplus and shortage for each period t and each product j.

$$s_{jt} =$$

Just-in-Time Production Planning (4/4)

- Integer lot size number.
- For example, $l_{jt} = 150$ and $x_{jt} = 700$. Then, the number of lots is 700/150=4.67 which is not an integer.
- A pair of constraints required:

and for all j and t

where $y_{jt} \geq 0$ and integer for all j and t.

- The objective is to

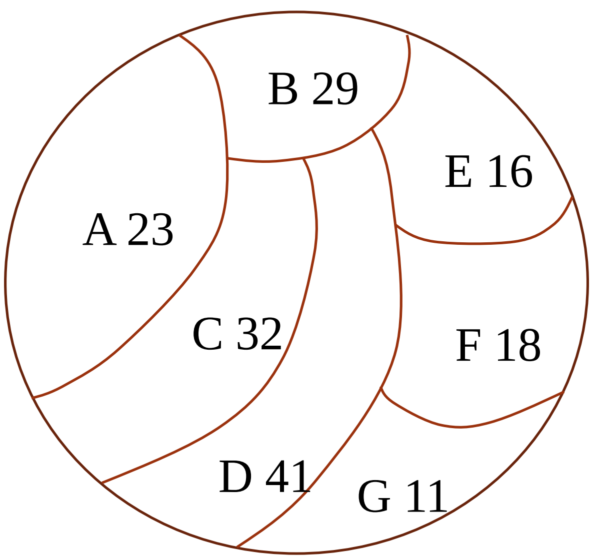
Minimize $z =$

Modeling Techniques

- How about $|x|$?
  - Let $x=$
- How about $c \leq \min\{c_1, c_2, c_3\}$?
  - Three constraints:
- How about $c \leq \min\{d_1 x_1, d_2 x_2, d_3 x_3\}$, x_1, x_2, x_3 are binary?
  -
  -
  -
  -

Set Covering Problem

- The problem is to decide where to install a set of service centers, so that a number of given regions can be covered.
- Let c_j be the cost to install a service center in region j, and a_{ij} be a indicator which shows whether region i is within the service range of the center j.



Minimize $\quad z = \sum_j c_j y_j$

subject to $\qquad\qquad i = 1, \dots, n$

$$y_j = 0 \text{ or } 1 \qquad j = 1, \dots, n$$

Workforce Scheduling Problems (1/3)

- Workforce time windows

Time Window	Shift 1	2	3	4	Workers Required
6 a.m.- 9 a.m.	X			X	55
9 a.m.- 12 noon	X				46
12 noon- 3 p.m.	X	X			59
3 p.m.-6 p.m.		X			23
6 p.m.-9 p.m.		X	X		60
9 p.m.-12 a.m.			X		38
12 a.m.-3 a.m.			X	X	20
3 a.m. -6 a.m.				X	30
Wage rate per 9h shift	$135	$140	$190	$188	

Source: Applied Integer Programming: Modeling and Solution, WILEY

Workforce Scheduling Problems (2/3)

Input parameters: number of shifts (n), number of time windows (T), number of workers required during each time window $(d_{t,}\ t = 1, 2, \ldots, T)$, wage rate per shift for a full-time worker (w_j), wage rate per time window per part-time worker (c_t)

Decision variables: number of full-time workers needed for each work shift (y_j), number of part-time workers needed for each time window (x_t)

Constraints: demand within each time window t must be satisfied, restriction on using part-time workers (can be used only if one or more full-time workers are available in the same time window)

Objective: minimize the total wages paid to all workers

Workforce Scheduling Problems (3/3)

- Let parameter a_{jt}=1 if shift j covers time window t, 0 otherwise.

Minimize $\quad \sum_j w_j y_j + \sum_t c_t x_t$

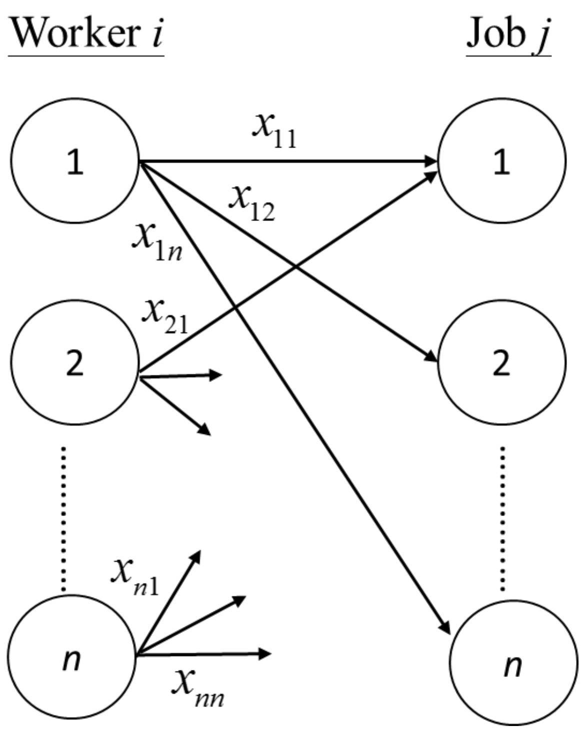
subject to $\qquad t = 1, \ldots, T$

$t = 1, \ldots, T$

$x_t, y_j \geq 0$ and integer $\quad j = 1, \ldots, n; \; t = 1, \ldots, T$

Assignment Problem

- There are n workers available to carry out n jobs.
- One-to-one assignment
- Each worker has his own expertise, so that each assignment has different costs (denoted as c_{ij}).



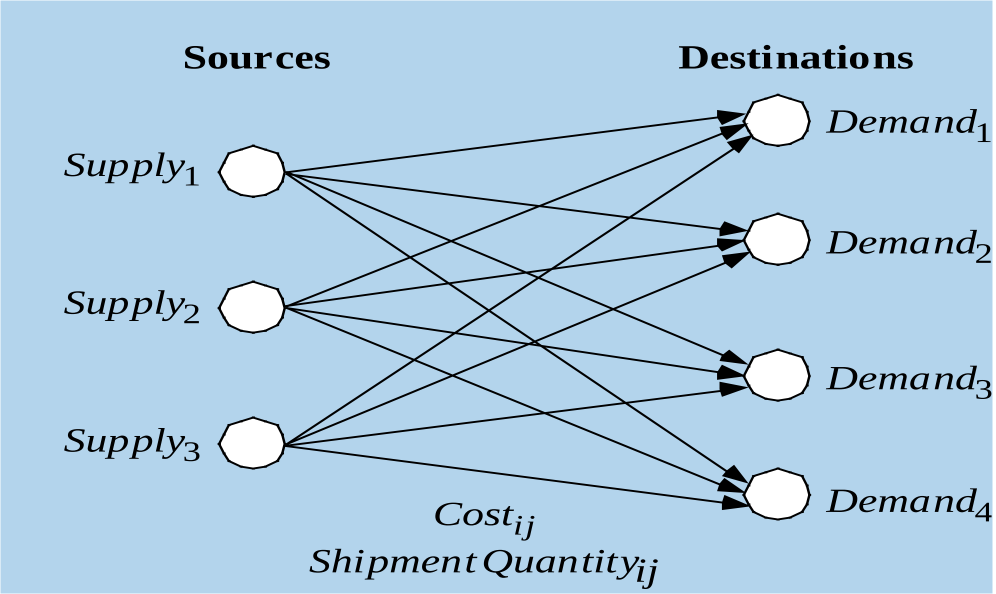
Minimize $\quad z = \sum_i \sum_j c_{ij} x_{ij}$

subject to $\qquad i = 1, \ldots, n$

$\qquad j = 1, \ldots, n$

$x_{ij} = 0$ or 1

Transportation and Distribution Problems



- Fixed-charge transportation
- Uncapacitated facility location
- Capacitated facility location

Fixed-Charge Transportation (1/2)

- In addition to shipping costs, a fixed cost associated with a route is charged when the route is used.

Decision variables:	whether or not source i will supply destination $j (y_{ij} = 1 \text{ or } 0)$. If yes, how much (x_{ij})
Input parameters:	unit shipping cost (c_{ij}), fixed cost (f_{ij}) from source i to destination j, demand at destination $j (d_j)$
Constraints:	demand at each destination must be satisfied (assuming unlimited product availability at each source node)
Objective:	minimize sum of fixed and variable costs

Fixed-Charge Transportation (2/2)

Let M be a "sufficiently" large number (we can let $M = \sum_j d_j$). Note that $y_{ij} = 1$ if and only if $x_{ij} > 0$. The transportation model can be formulated as

$$
\begin{array}{lll}
\text{Minimize} & \sum_i \sum_j (c_{ij} x_{ij} + f_{ij} y_{ij}) & \\
\text{subject to} & \sum x_{ij} = d_j & j = 1, \dots, n \\
 & & i = 1, \dots, m; \; j = 1, \dots, n \\
 & x_{ij} \geq 0 & i = 1, \dots, m; \; j = 1, \dots, n \\
 & \mathrm{y}_{ij} = 0 \text{ or } 1 & i = 1, \dots, m; \; j = 1, \dots, n
\end{array}
$$

Uncapacitated Facility Location (1/2)

- Determine a set of sources to be open for supply.

Decision variables:	whether or not distribution center i should be opened $(y_i = 1 \text{ or } 0)$. If opened, how much should be shipped from distribution center to retail store (x_{ij})
Input parameters:	unit shipping cost from center i to retail $j (c_{ij})$, fixed cost for opening distribution center (f_i)
Constraints:	all demands are to be met at all retail stores
Objective:	minimize total cost of opening and transportation cost

Uncapacitated Facility Location (2/2)

Let M be a "sufficiently" large number (we can let $M = \sum_j d_j$). Note that $y_{ij} = 1$ if and only if $\sum_j x_{ij} > 0$. The uncapacitated facility location problem can be formulated as

$$
\begin{array}{lll}
\text{Minimize} & \displaystyle\sum_i \sum_j c_{ij} x_{ij} + \sum_i f_i y_i & \\
\text{subject to} & \displaystyle\sum_i x_{ij} = d_j & j = 1, \ldots, n \\
& \displaystyle\sum_j x_{ij} \le M y_i & i = 1, \ldots, m \\
& x_{ij} \ge 0 \text{ and integer} & i = 1, \ldots, m;\ j = 1, \ldots, n
\end{array}
$$

Capacitated Facility Location

- Replace "M" with the supply bound u.

$$
\begin{array}{lll}
\text{Minimize} & \sum_i \sum_j c_{ij} x_{ij} + \sum_i f_i y_i & \\
\text{subject to} & \sum_i x_{ij} = d_j & j = 1, \dots, n \\
 & \sum_j x_{ij} \le u_i y_i & i = 1, \dots, m \\
 & x_{ij} \ge 0 \text{ and integer} & i = 1, \dots, m;\ j = 1, \dots, n \\
 & y_i = 0 \text{ or } 1 & i = 1, \dots, m
\end{array}
$$

Outlines

- Introduction
- Integer programming problems
- Integer programming modeling
- Cutting planes

Basic Logical Operations on Variable

- To select a subset of n projects in a manner that maximizes the total present value while satisfying the budget limitation.
- Let y_j=1 if project j is selected, and 0 otherwise.
  - Statement ***A***: project A is selected (y_A=1) or not selected (y_A=0)
  - Statement ***B***: project A is selected (y_B=1) or not selected (y_B=0)
- To obtain a correct MIP model,
  - Only linear equations/ inequalities are allowed.
  - If more than one constraint is required, these constraints have to be satisfied simultaneously.
  - Only the true value (= 1) is of interest.

Conjunction (A and B, $A \cap B$)

- The conjunction of two statements, ***A*** and ***B***, implies that both projects ***A*** and ***B*** are selected, or symbolically

$$\boxed{y_A = 1 \text{ and } y_B = 1}$$

- An alternate formulation is ________ .

Disjunction (*A* or *B*, *A* ∪ *B*)

- The disjunction relation of two statements, ***A*** or ***B*,** implies that either ***A*** or ***B*** or both are true.
- At least one of the projects A or B must be selected.

	y_A	y_B
Project A is selected but not B	1	0
Project B is selected but not A	0	1
Both are selected	1	1

Simple Implication (If *A* Then *B*, *A*→*B*)

- If statement ***A*** is true, statement ***B*** must be true.
- If statement ***A*** is not true, statement ***B*** can be either true or false.

	y_A	y_B
Project A is selected and B is selected	1	1
Project A is not selected and project B is selected	0	1
Project A is not selected and project B is not selected	0	0

Double Implication (A If and only If B)

- Statement ***A*** implies ***B***, and ***B*** also implies ***A***.
- Project ***A*** is selected if and only if project ***B*** is selected.

	y_A	y_B
Project A is selected and B is selected	1	1
Project A is not selected and project B is not selected	0	0

Linear Expressions for Boolean Relations

Logical Relation	Linear Inequality/Equation
$y_C = y_A \cap y_B$	$y_C \leq y_A$ $y_C \leq y_B$ $y_C \geq y_A + y_B - 1$
$y_C = y_A \cup y_B$	$y_C \geq y_A$ $y_C \geq y_B$ $y_C \leq y_A + y_B$
$y_A \rightarrow y_C$	$y_A \leq y_C$
$y_C = \sim y_A$	$y_C = 1 - y_A$

Source: Applied Integer Programming: Modeling and Solution, WILEY

Either/Or Constraints

- A decision variable may be defined by two disjunctive regions.
- For instance, either $x \leq 3$ or $x \geq 10$.
- Constraints transformation:

$$\boxed{\begin{array}{l} \qquad x - 3 \leq My \\ \text{and} \quad -x + 10 \leq M(1 - y) \end{array}}$$

- When y=0, constraint $x \geq 10$ is always true.
- When y=1, constraint $x \leq 3$ is always true.

One-Machine Scheduling Problem

- Let x_i and x_j respectively denote the start time of job *i* and job *j* to be scheduled.
- Let t_i and t_j respectively represent the known machine processing time of job *i* and job *j*.

$$\text{Either} \quad x_i + t_i \le x_j \quad \text{or} \quad x_j + t_j \le x_i$$

- Sequence constraint (if job *i* before job *j*, variable y_{ij}=0):

p Out of m Constraints Must Hold

- Consider the case where the model has a set of m constraints but in addition requires only some p out of m (assuming $p<m$) constraints to hold.
- Let $y_i = 1$ for constraint i is relaxed, and 0 otherwise.

$$f_i(x) - b_i \le My_i \qquad \text{for } i = 1, 2, \ldots, m$$

and y_i is binary for all i.

If/Then Constraints

- If constraint ***A*** holds, constraint ***B*** must hold.
- If constraint ***A*** **doesn't** hold, constraint ***B*** can be either true or false (be relaxed).
- If ***A*** then ***B*** is equivalent to the logical statement $\sim A \cup B$.

y_A	y_B	$y_A \rightarrow y_B$	$\sim y_A \cup y_B$
1	1	1	1
0	1	1	1
0	0	1	1
1	0	0	0

If/Then Constraints (con't)

- We have two constraints: $f_1(x) - b_1 < 0$ and $f_2(x) - b_2 \leq 0$.
- $\sim \boldsymbol{A} \cup \boldsymbol{B}$**:** either or $f_2(x) - b_2 \leq 0$.
- Constraint ***A*** is satisfied, only when y is 1. That is, constraint ***B*** must be satisfied.
- When y = 0, constraint A can't be satisfied.
- Thus, A=1 and B=0 never be happened.

Example - If/Then

- If $x_1=1$, then $x_2=x_3=x_4=0$
- Because all variables are binary, the following can be obtained

- $\sim A \cup B$: $x_1 \le 0$ and $x_2+x_3+x_4 \le 0$
- **Then,** $\boxed{\begin{aligned} & x_1 \le My \\ & x_2 + x_3 + x_4 \le M(1-y) \end{aligned}}$

- $x_1=1$ then $y=1$, $\boxed{x_1 \le 3, x_2 + x_3 + x_4 \le 0}$
- $x_1=0$, x_2,x_3,x_4 are unrestricted. $\boxed{x_1 \le 0, x_2 + x_3 + x_4 \le 3}$

Outlines

- Introduction
- Integer programming problems
- Integer programming modeling
- Cutting planes
  - Cut for IP

LP vs MILP

Graphical Solutions

$$\text{Maximize} \quad 2x - y = z$$

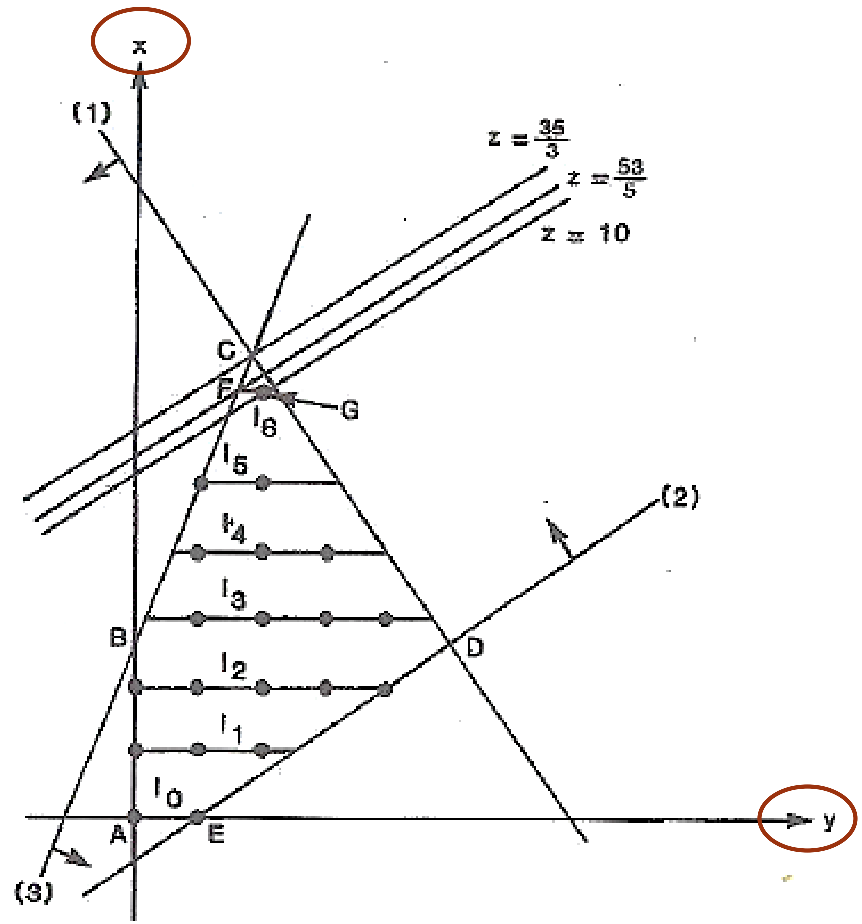
$$\text{subject to} \quad 5x + 7y \leq 45,$$

$$-2x + y \leq 1,$$

$$2x - 5y \leq 5,$$

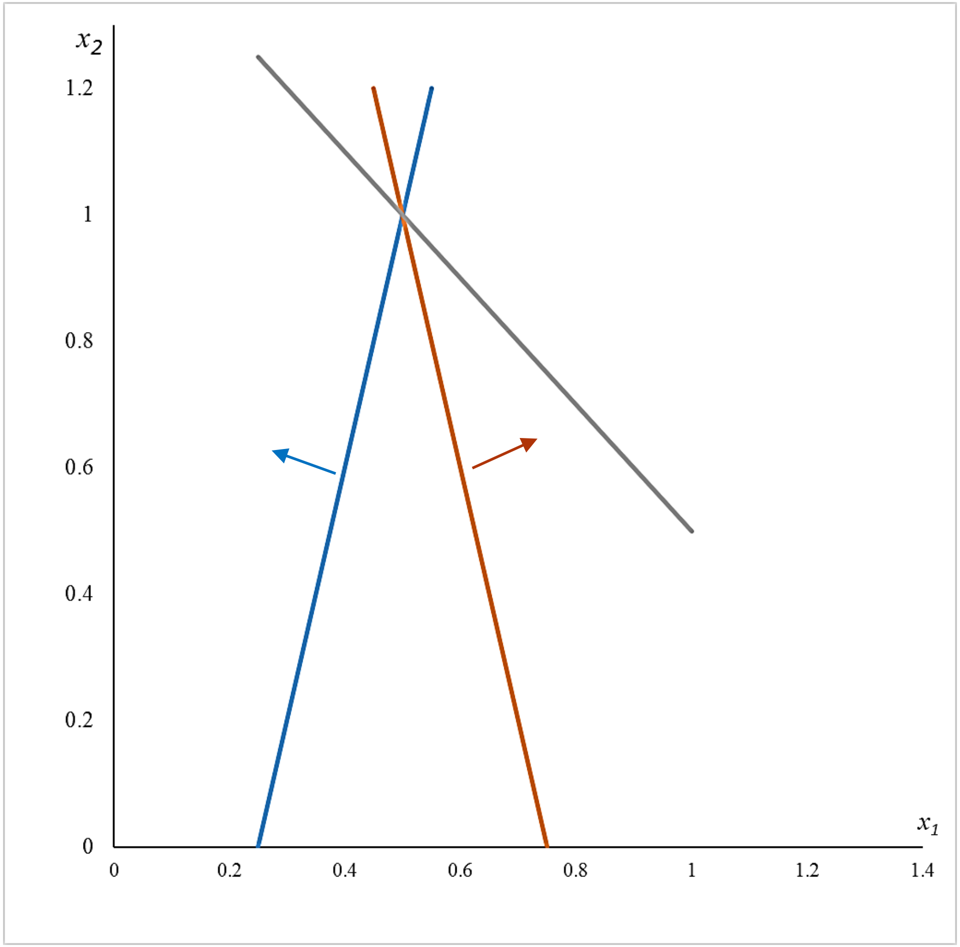
$$x, \quad y \geq 0,$$

$$\text{and} \quad x \ \text{integer}.$$



No Solution

$$
\begin{aligned}
\text{Maximize} \quad & x_1 + x_2 = z \\
\text{subject to} \quad & -4x_1 + x_2 \le -1, \\
& 4x_1 + x_2 \le 3, \\
& x_1,\ \ x_2 \ge 0 \text{ and integer}
\end{aligned}
$$



Some Insights

- The maximal value of the objective function to the MIP (IP) solved as a LP is ____________ on the value of any feasible solution to MIP (IP).
- If the optimal solution to the MIP solved as a LP is integer in its integer constrained variables, it solves the MIP
- If the MIP solved as a linear one is infeasible, so is the MIP.

Rounding LP Solutions

- Example: facility location (assume all produced items have to be shipped out)

		Shopping costs Customers (n)					Production Capacity M_i
		1	2	3	4	5	
Sources (m)	1	93	70	48	68	81	**2**
	2	45	89	97	85	96	**3**
	3	92	93	58	37	99	**2**
	4	55	103	55	57	38	**3**
	5	74	60	78	54	52	**2**
	Demands d_j	**1**	**1**	**1**	**1**	**1**	

LP Solution

LP optimal solution, cost = 228

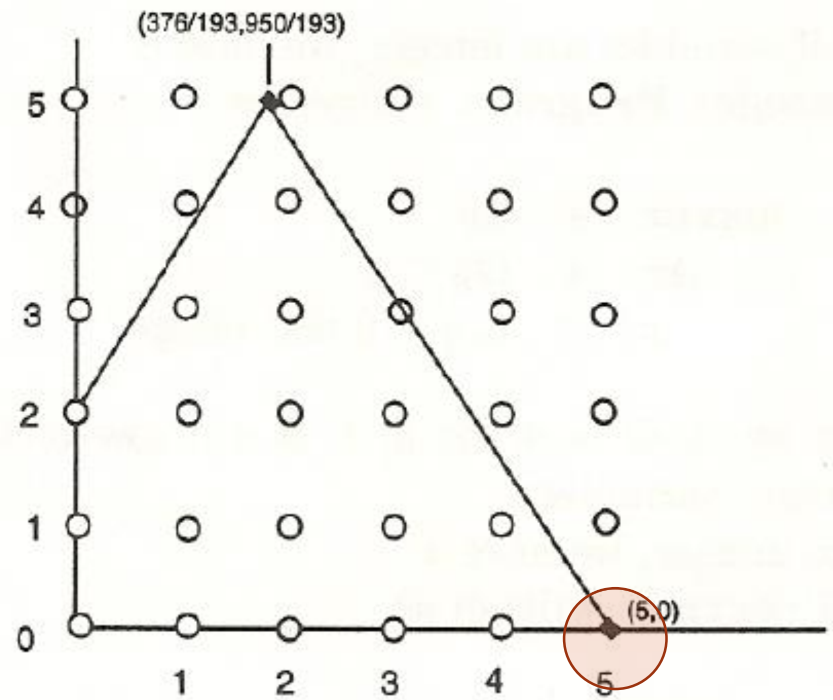
$$x_1 = x_3 = x_5 = 1/2, \quad x_2 = x_4 = 1/3,$$

$$z_{13} = z_{21} = z_{34} = z_{45} = z_{52} = 1$$

Facility variables	Shipping variables	Total cost
$x_1 = x_2 = 1$	$z_{12}=z_{13}=z_{21}=z_{24}=z_{25}=1$	344
$x_1 = x_4 = 1$	$z_{12}=z_{13}=z_{41}=z_{44}=z_{45}=1$	268
$x_3 = x_2 = 1$	$z_{21}=z_{22}=z_{25}=z_{33}=z_{34}=1$	325
$x_3 = x_4 = 1$	$z_{32}=z_{34}=z_{41}=z_{43}=z_{45}=1$	278
$x_5 = x_2 = 1$	$z_{21}=z_{22}=z_{23}=z_{54}=z_{55}=1$	337
$x_5 = x_4 = 1$	$z_{41}=z_{43}=z_{45}=z_{52}=z_{54}=1$	**262**

Example 2

$$\max 1.00x_1 + 0.64x_2$$
$$50x_1 + 31x_2 \leq 250$$
$$3x_1 - 2x_2 \geq -4$$
$$x_1, x_2 \geq 0 \text{ and integer.}$$



Source: L.A. Wolsey, *Integer Programming*, Wiley 1998.

Concept of cuts

- ABCDE is the feasible region of Linear Programming

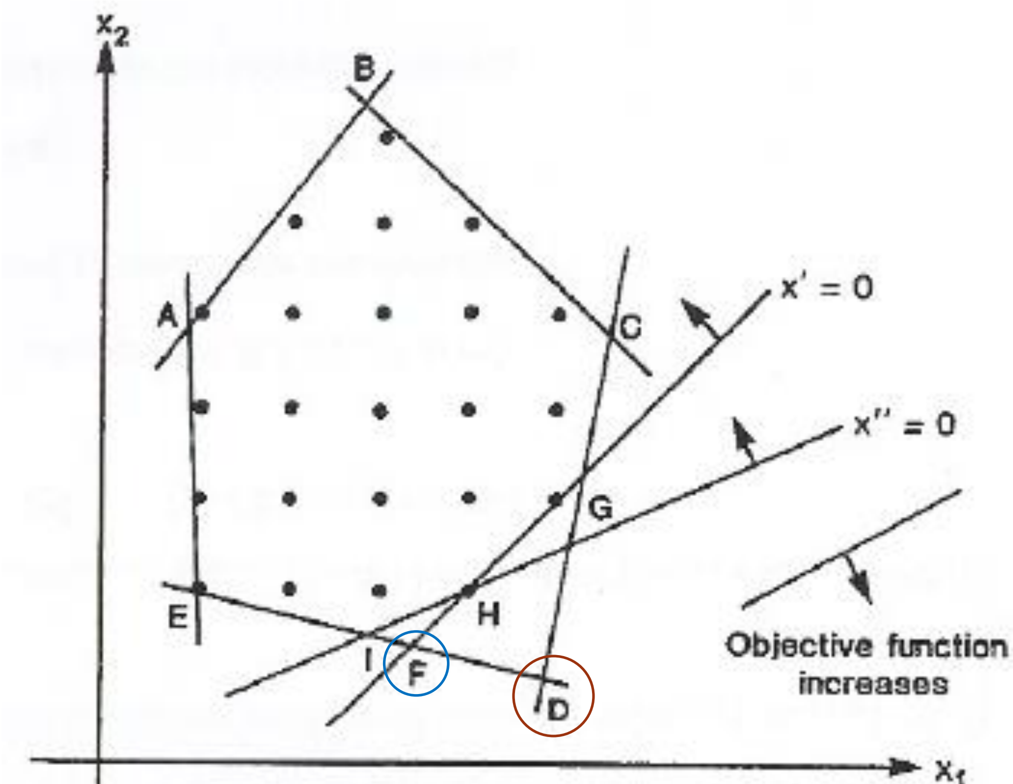


Figure 4.1: The Basic Approach

Cutting Plane: Procedures

The there main steps are:

1. Solve the LP relaxation with the simplex method. If the problem is an ILP, start with an all-integer tableau (or with tableau of rational numbers). If the problem is infeasible or has an integer solution, stop. Otherwise, go to step 2.
2. Whenever the solution is noninteger, the integrality constraints imply new additional constraints (or "cuts"), which cut off the current optimal point. Add a new constraint to the tableau, which will produce primal infeasibility.
3. Re-optimize with the dual simplex method. If the new problem is infeasible or has an integer solution, stop. Otherwise, go to 2.

Two Keys of the Cut

- The current LP optimal solution will become infeasible due to the cut constraint.
- No integer solutions are cut off by the constraint.

Fractional Cutting Plane Method

- For a pure integer program
- The very beginning form contains all integer coefficients

Form of a Gomory Cut

There is a row v,

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

k^{th} Gomory cut,

$$\sum_{j=1}^{n}(-f_{v,j})\left(x_{J(j)}\right) + x_{n+m+k} = -f_{v,0}\ , \quad \text{(Gomory, 1958)}$$

where x_{n+m+k} is called Gomory slack variable,

$$f_{v,j} = a_{v,j} - \lfloor a_{v,j} \rfloor, j = 0, \dots n.$$

Note that $0 \leq f_{v,j} < 1, j = 0, \dots n, 0 < f_{v,0} < 1.$

Ex: 2.6 = 2 + 0.6 or = 3 – 0.4
–1.8 = –2 + 0.2 or = –1 – 0.8

Example

$$\begin{aligned} \text{Maximize} \quad & 5x_1 - 2x_2 = z \\ \text{subject to} \quad & -x_1 + 2x_2 + s_1 = 5 \\ & 3x_1 + 2x_2 + s_2 = 19 \\ & -x_1 - 3x_2 + s_3 = -9 \\ & x_1, x_2, s_1, s_2, s_3 \geq 0 \text{ and integer.} \end{aligned}$$

- A pure IP
- All coefficients are integer including the slack variables

Optimal Tableau for LP

	z	x_1	x_2	s_1	s_2	s_3	**RHS**
z	1	0	0	0	17/7	16/7	**179/7**
x_1	0	1	0	0	3/7	2/7	**39/7**
x_2	0	0	1	0	−1/7	−3/7	**8/7**
s_1	0	0	0	1	5/7	8/7	**58/7**

- Not an IP solution
- Source row: x_1 (x_2 and s_1 are also valid)
- The cut?

Add the Cut

	z	x_1	x_2	s_1	s_2	s_3	x_3	**RHS**
z	1	0	0	0	17/7	16/7	0	**179/7**
x_1	0	1	0	0	$3/7$	$2/7$	0	**39/7**
x_2	0	0	1	0	$-1/7$	$-3/7$	0	**8/7**
s_1	0	0	0	1	$5/7$	$8/7$	0	**58/7**
x_3	0	0	0	0	$-3/7$	$-2/7$	1	**-4/7**

- The solution becomes infeasible
- This optimal solution is cut off
- Apply dual simplex method

Cut 2

	z	x_1	x_2	s_1	s_2	s_3	x_3	**RHS**
z	1	0	0	0	0	2/3	17/3	**67/3**
x_1	0	1	0	0	0	0	1	**5**
x_2	0	0	1	0	0	$-1/3$	$-1/3$	**4/3**
s_1	0	0	0	1	0	2/3	5/3	**22/3**
s_2	0	0	0	0	1	2/3	$-7/3$	**4/3**
x_4								

- Source row: s_1

Cut 3

	z	x_1	x_2	s_1	s_2	s_3	x_3	x_4	**RHS**
z	1	0	0	0	0	0	5	1	**22**
x_1	0	1	0	0	0	0	1	0	**5**
x_2	0	0	1	0	0	0	0	$-1/2$	**3/2**
s_1	0	0	0	1	0	0	1	1	**3**
s_2	0	0	0	0	1	0	-3	1	**1**
s_3	0	0	0	0	0	1	1	$-3/2$	**1/2**
x_5									

- Source row: x_2

Optimal IP Solution

	z	x_1	x_2	s_1	s_2	s_3	x_3	x_4	x_5	**RHS**
z	1	0	0	0	0	0	5	0	2	**21**
x_1	0	1	0	0	0	0	1	0	0	**5**
x_2	0	0	1	0	0	0	0	0	1	**2**
s_1	0	0	0	1	0	0	1	0	2	**2**
s_2	0	0	0	0	1	0	-3	0	2	**0**
s_3	0	0	0	0	0	1	1	0	-3	**2**
x_4	0	0	0	0	0	0	0	1	-2	**1**

- An IP solution is obtained and it is optimal!!

Derive the Cut

There is a row v,

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

Let $f_{v,j} = a_{v,j} - \lfloor a_{v,j} \rfloor$,

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

All decision variables are integer,

$$x_v + \sum_{j=1}^{n} (\lfloor a_{v,j} \rfloor x_{J(j)} + f_{v,j} x_{J(j)}) = \lfloor a_{v,0} \rfloor + f_{v,0}$$

Derive the Cut #2

Remove the integer parts,

Thus,

$$\sum_{j=1}^{n}(f_{v,j})\left(x_{J(j)}\right) - f_{v,0} \geq 0 \text{ or } \sum_{j=1}^{n}(f_{v,j})\left(x_{J(j)}\right) \geq f_{v,0}$$

The cut,

$$\sum_{j=1}^{n}(-f_{v,j})\left(x_{J(j)}\right) \leq -f_{v,0}$$

$\sum_{j=1}^{n}(-f_{v,j})\left(x_{J(j)}\right) + x' = -f_{v,0}$ Note that x' is !!

Questions?

- Next class of IP will be on 9:10 12/27.