UNIVERSIDAD PONTIFICIA ICAI ICADE COMILLAS MADRID

upcomillas.es

Mixed integer linear modeling

Andrés Ramos (Andres.Ramos@comillas.edu)
Pedro Sánchez (Pedro.Sanchez@comillas.edu)
Sonja Wogrin (Sonja.Wogrin@comillas.edu)

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA
DEPARTAMENTO DE ORGANIZACIÓN INDUSTRIAL

CONTENTS

- **PROBLEM CLASSIFICATION**
- SEVERAL CHARACTERISTIC PROBLEMS
- FIXED COST PROBLEM
- LOGICAL PROPOSITIONS
- MINIMUM, MAXIMUM AND ABSOLUTE VALUE
- PIECEWISE LINEAR (master)
- CONVEX AND NONCONVEX REGION (master)
- SPECIAL ORDERED SETS (master)
- REFORMULATION (master)

1

Problem Classification

IP Problem classification

- *Linear* problems where several or all the variables are *integer*. A particular case of integer variables are *binary* variables (0/1).

1. PIP (*pure integer programming*) all integer
2. BIP (*binary integer programming*) all binary
3. MIP (*mixed integer programming*) some integer o binary

Justification of optimization problem with integer variables

- Investments are discrete variables (generation or transmission expansion planning, singular equipment acquisition, people hiring)

- Decisions are binary variables (plant or store location)

Binary representation of discrete variables

- *x* integer variable
- y_i binary variable (0/1)

$$x = \sum_{i=0}^{N} 2^i y_i \qquad 0 \le x \le u \qquad 2^N \le u \le 2^{N+1}$$

2

Several Characteristic Problems

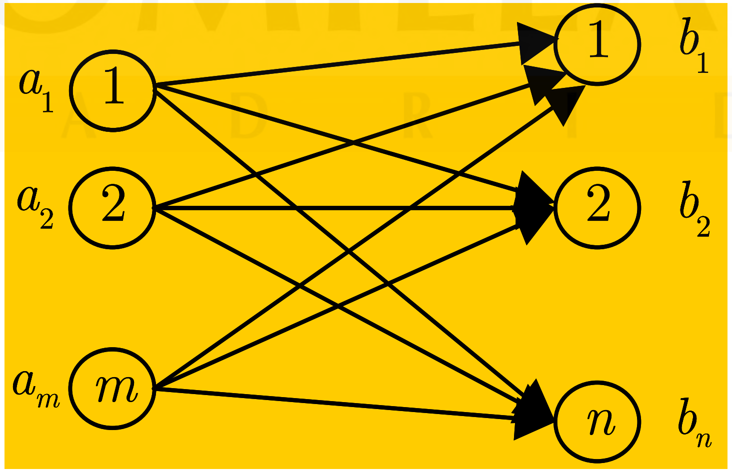
Several LP and BIP characteristic problems

- They have been exhaustively studied. Limited importance in practice, but they may appear as part of other problems.
- *Linear Programming LP*
  - Transportation
  - Transshipment
  - Assignment
- *Binary Integer Programming BIP*
  - Knapsack
  - Covering
  - Packing
  - Partitioning
  - Traveling salesman

Transportation problem

- Minimize the total transportation cost of a certain product from origin to destination, satisfying the destination demand without exceeding the origin offer.

- a_i product offer in origin *i* — *m* origins
- b_j product demand in destination *j* — *n* destinations
- c_{ij} unitary transportation cost from *i* to *j*



Transportation problem formulation

$$\min_{x_{ij}} \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$$

- Offer available in each origin *i* $\sum_{j=1}^{n} x_{ij} = a_i \quad \forall i = 1,\ldots,m$
- Demand in each destination *j* $\sum_{i=1}^{m} x_{ij} = b_j \quad \forall j = 1,\ldots,n$
- $x_{ij} \geq 0$ units of product transported from *i* to *j* $\forall i, j$
- Hypothesis: offer equals demand of the product

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$$

- If $\sum_{i=1}^{m} a_i > \sum_{j=1}^{n} b_j$ add universal sink with zero cost
- If $\sum_{i=1}^{m} a_i < \sum_{j=1}^{n} b_j$ add universal source with very high cost

Transportation problem structure

	x_{11}	x_{12}	…	x_{1n}	x_{21}	x_{22}	…	x_{2n}	…	x_{m1}	x_{m2}	…	x_{mn}
	1	1	…	1									
m restricciones de oferta					1	1	…	1					
									⋱				
										1	1	…	1
	1				1				…	1			
n restricciones de demanda		1				1			…		1		
			⋱				⋱		…			⋱	
				1				1	…				1

If a_i and b_j are integer $\Rightarrow$ x_{ij} are integer because the matrix is totally unimodular (i.e., every square submatrix has determinant 0, 1 or –1)

Transshipment problem

- Determine in a network of n nodes the cheapest routes to carry product units from their origins to their destinations through intermediate transshipment locations.

- Each *origin* generates $b_i > 0$ units.
- Each *destination* consumes $b_i < 0$ units.
- Each *transshipment* neither generates nor consumes units $b_i = 0$.
- c_{ij} transportation unit cost from i to j in this direction.

Transshipment problem formulation

$$\min_{x_{ij}} \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij} x_{ij}$$

- Balance or flow conservation in each node *i*

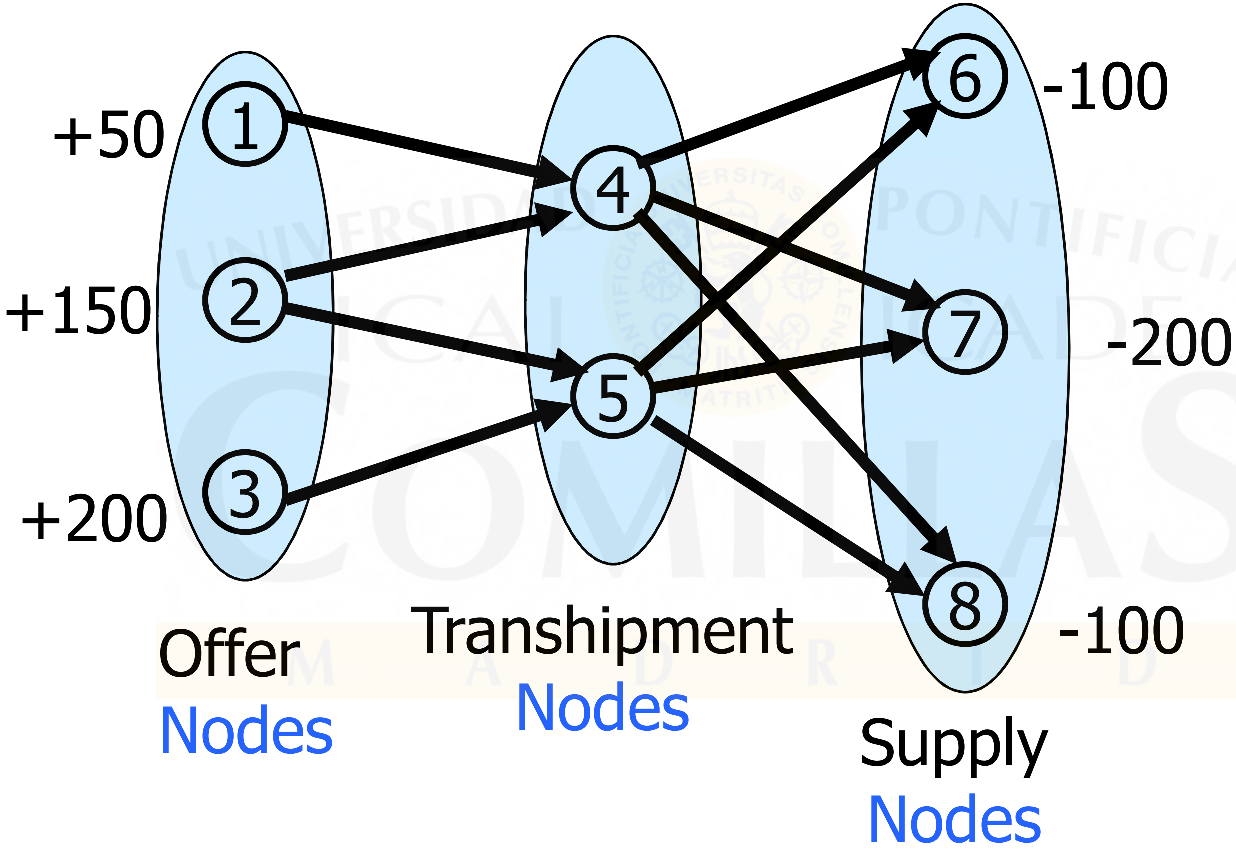
$$\sum_{j=1}^{n} x_{ij} - \sum_{k=1}^{n} x_{ki} = b_i \quad \forall i = 1,\ldots,n$$

$\forall i, j$

- $x_{ij} \geq 0$ units of product transported from *i* to *j*
- Hypothesis: offer equals demand

$$\sum_{i=1}^{n} b_i = 0$$

Transhipment Example



$b_i \geq 0$

$b_i \leq 0$

Task assignment problem

- n tasks

  n persons (machines, etc.) to do them
- It is particular case of a transportation problem.
- Minimize the total cost of doing the tasks knowing that each person does 1 task and each task is done by 1 person.
- c_{ij} cost of doing task i by person j

$$x_{ij} = \begin{cases} 1 & \text{if task } \boldsymbol{i} \text{ is done by person } \boldsymbol{j} \\ 0 & \text{otherwise} \end{cases}$$

- *Although it is not necessary to declare them as binary variables*.

Task assignment problem formulation

$$\min_{x_{ij}} \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$$

- Each task *i* is done by a person

$$\sum_{j=1}^{n} x_{ij} = 1 \quad \forall i = 1, \ldots, n$$

- Each person *j* does one task

$$\sum_{i=1}^{n} x_{ij} = 1 \quad \forall j = 1, \ldots, n$$

$$x_{ij} \geq 0 \quad \forall i, j$$

Task sequencing in one machine

- Given several tasks to do, their duration and an estimated problem due date, state the mixed integer programming problem to find the sequence that minimizes the mean delay of the tasks, with the following data:

Task	A	B	C	D
Processing time	9	12	7	14
Due date	15	19	23	31

Task sequencing in one machine

- Let d_j be the processing time of task *j* and r_j the due date of task *j*.
- Define the problem variables as

$$x_{ij} = \begin{cases} 1 & \text{if task } j \text{ is done in step } i \\ 0 & \text{otherwise} \end{cases}$$

- The objective function will minimize the mean delay

$$\min \frac{1}{4} \sum_i p_i$$

- Subject to these constraints:
  - Each task is done once

$$\sum_i x_{ij} = 1 \quad \forall j$$

  - In each step only one task

$$\sum_j x_{ij} = 1 \quad \forall i$$

Task sequencing in one machine

- For each step i a task is done and its due date is $\sum_j r_j x_{ij}$
- On the other hand, task j done in this step end at time $\sum_j d_j \sum_{k \le i} x_{kj}$

- Variables n_i and p_i, take into account if the task ends before due date (prompted) or after (delayed), therefore p_i , is the delay, that appears in the objective function

$$\sum_j d_j \left(\sum_{k \le i} x_{kj} \right) + n_i - p_i = \sum_j r_j x_{ij} \quad \forall i$$

$$n_i, p_i \ge 0 \quad x_{ij} \in \{0,1\}$$

Knapsack problem

- n projects
- Maximize the total value of selecting a set of projects without exceeding the available budget.
- c_j cost of project j
- v_j value of project j
- b available budget

$$\boldsymbol{x_j} = \begin{cases} 1 & \text{if project } \boldsymbol{j} \text{ is done} \\ 0 & \text{otherwise} \end{cases}$$

Knapsack problem formulation

$$\max_{x_j} \sum_{j=1}^{n} v_j x_j$$

- Limit on the available budget

$$\sum_{j=1}^{n} c_j x_j \le b$$

$$x_j \in \{0,1\} \quad \forall j$$

Set Covering Problem

- m characteristics (flights)
- n set of characteristics (sequences of flights). If a set is selected, then all characteristics of this set should be done.
- Minimize the total cost of the selected sets in such a way that all characteristics are covered at least once.
- c_j cost of selected set j
- Membership matrix

$$a_{ij} = \begin{cases} 1 & \text{if characteristic } \boldsymbol{i} \text{ belongs to set } \boldsymbol{j} \\ 0 & \text{otherwise} \end{cases}$$

- Decision variables

$$x_j = \begin{cases} 1 & \text{if set } \boldsymbol{j} \text{ is selected} \\ 0 & \text{otherwise} \end{cases}$$

Set Covering Formulation

$$\min_{x_j} \sum_{j=1}^{n} c_j x_j$$

- Each characteristic *i* should be selected at least once.

$$\sum_{j=1}^{n} a_{ij} x_j \geq 1 \quad i = 1, \ldots, m$$

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

Set Covering Example: Crew Assignment

- An airline company needs to assign crews to cover all its flights. Specifically, it requires to solve the set covering problem of three crews whose origin airport is San Francisco for all flights shown in the first column of the table. The rest of columns show 12 feasible flight sequences (sets) for any crew.
- It is necessary to choose three flight sequences (one for each crew) to cover all flights. It is possible to have more than one crew in the same flight (the extra crew is considered as passengers although the personnel is paid normally). The assignment cost of a crew to a flight sequence is given in thousands of Euros at the last row.
- The objective is to minimize the total assignment cost of the three crews to cover all flights.

Feasible Flight Sequences

	1	2	3	4	5	6	7	8	9	10	11	12
SF - LA	1			1			1			1		
SF - Denver		1			1			1			1	
SF - Seattle			1			1			1			1
LA - Chicago				2			2		3	2		3
LA - SF	2					3				5	5	
Chicago - Denver				3	3				4			
Chicago - Seattle							3	3		3	3	4
Denver - SF		2		4	4				5			
Denver - Chicago					2			2			2	
Seattle - SF			2				4	4				5
Seattle - LA						2			2	4	4	2
Cost (M€)	2	3	4	6	7	5	7	8	9	9	8	9

Crew Assignment Formulation

$$\min \; 2x_1 + 3x_2 + 4x_3 + 6x_4 + 7x_5 + 5x_6 + 7x_7 + 8x_8 + 9x_9 + 9x_{10} + 8x_{11} + 9x_{12}$$

- Flight Covering

$$\begin{aligned} x_1 + x_4 + x_7 + x_{10} &\geq 1 \\ x_2 + x_5 + x_8 + x_{11} &\geq 1 \\ x_3 + x_6 + x_9 + x_{12} &\geq 1 \\ &\vdots \end{aligned}$$

- Three crews Assignment

$$\sum_{j=1}^{12} x_j = 3 \qquad x_j \in \{0,1\} \quad j = 1,\ldots 12 \qquad x_j = \begin{cases} 1 & \text{if the set } j \text{ is chosen} \\ 0 & \text{otherwise} \end{cases}$$

- Solution
  - $x_3 = x_4 = x_{11} = 1$ $\quad x_j = 0 \quad j \neq 3, 4, 11$ $\quad$ cost = 18 M€
  - $x_1 = x_5 = x_{12} = 1$ $\quad x_j = 0 \quad j \neq 1, 5, 12$ $\quad$ cost = 18 M€

Set Packing Problem

- *m* projects
- *n* project sets. If a set is selected all projects of this set are done.
- Maximize the total benefit without doing one project more than once.
- c_j benefit of selecting the set *j*

$$a_{ij} = \begin{cases} 1 & \text{if the project } \boldsymbol{i} \text{ is in the set } \boldsymbol{j} \\ 0 & \text{otherwise} \end{cases}$$

$$x_j = \begin{cases} 1 & \text{if set } \boldsymbol{j} \text{ is selected} \\ 0 & \text{otherwise} \end{cases}$$

Set Packing Formulation

$$\max_{x_j} \sum_{j=1}^{n} c_j x_j$$

- Each project *i* of all sets cannot be selected more than once

$$\sum_{j=1}^{n} a_{ij} x_j \leq 1 \quad i = 1, \ldots, m$$

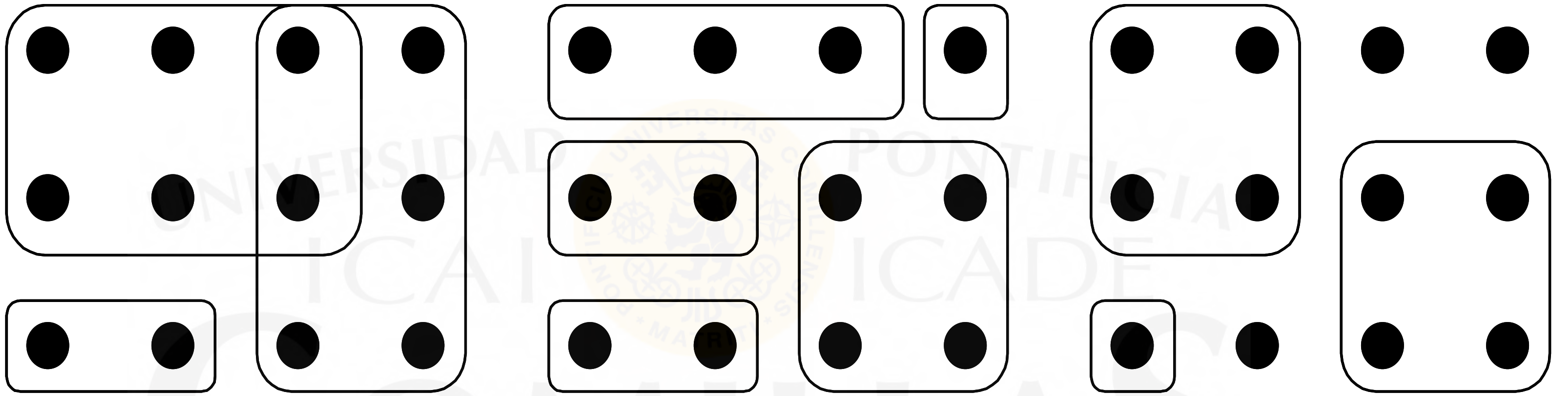
$$x_j \in \{0,1\} \quad j = 1, \ldots, n$$

Set Partitioning Problem

- Exactly each characteristic (project) of all sets should be chosen only once

$$\sum_{j=1}^{n} a_{ij} x_j = 1 \quad i = 1, \ldots, m$$

Dot Graphs of Covering, Partitioning and Packing Problems



COVERING PARTITIONING PACKING

Traveling Salesman Problem (TSP)

- Given a list of cities and the distances between each pair of cities, this problem consists of finding what is the shortest possible route that visits each city exactly once and returns to the origin city
- *Formulation 1:* $x_{ij} = \begin{cases} 1 & \text{if the path between } i \text{ and } j \text{ is included in the route} \\ 0 & \text{otherwise} \end{cases}$

$$\min_{x_{ij}} \sum_i \sum_j c_{ij} x_{ij}$$

$$\sum_i x_{ij} = 1 \quad \forall j$$

$$\sum_j x_{ij} = 1 \quad \forall i$$

$$\sum_{i,j \in U} x_{ij} \le \text{Card}(U) - 1 \qquad \forall U \subset \{1,\ldots,n\} / 2 \le \text{Card}(U) \le n-2$$

$$x_{ij} \in \{0,1\}$$

Traveling Salesman Problem (TSP)

- *Formulation 2:*

$$x_{ijk} = \begin{cases} 1 & \text{if the path between } \boldsymbol{i} \text{ and } \boldsymbol{j} \text{ is included in the route at stage } \boldsymbol{k} \\ 0 & \text{otherwise} \end{cases}$$

$$\min_{x_{ijk}} \sum_{i,j,k} c_{ij} x_{ijk}$$

$$\sum_{j,k} x_{ijk} = 1 \quad \forall i$$

$$\sum_{i,k} x_{ijk} = 1 \quad \forall j$$

$$\sum_{i,j} x_{ijk} = 1 \quad \forall k$$

$$\sum_{i} x_{ijk} = \sum_{r} x_{jrk+1} \quad \forall j,k$$

$$x_{ijk} \in \{0,1\}$$

3

Fixed Cost Problem

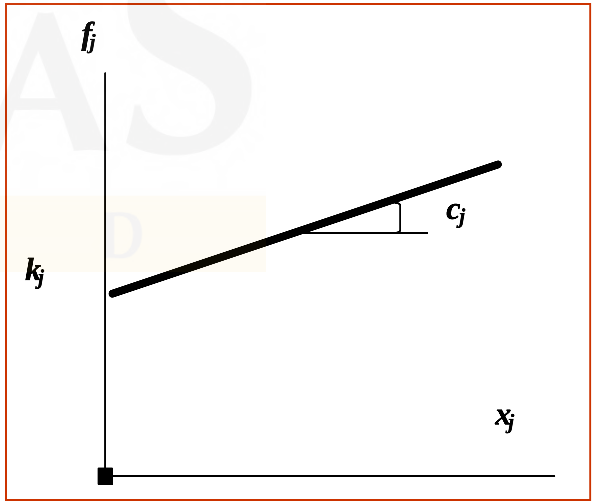
Fixed Cost Problem

- Objective Cost Function: $f_j(x_j)=\begin{cases} 0 & x_j=0 \\ k_j+c_jx_j & x_j>0 \end{cases}$
- A binary variable y_j represents the binary decision on an activity realization x_j

$$y_j=\begin{cases} 1 & x_j>0 \\ 0 & x_j=0 \end{cases}$$

- Mathematical formulation:

$$\min \sum_{j=1}^{n} f_j(x_j)=\sum_{j=1}^{n}\left(k_jy_j+c_jx_j\right)$$
$$x_j \le M_jy_j$$
$$x_j \ge 0 \quad j=1,\dots,n$$
$$y_j \in \{0,1\} \quad j=1,\dots,n$$



- M_j should have the lowest possible value

Fixed Cost Problem: Unit Commitment on Electric Systems

- Determine thermal generation units should be connected to the electric network each hour of the day (or week) in such a way that:
  - Variable Generation Costs are minimized (including fuel costs and **startup/shutdown costs**).
  - Demand is supplied each hour
  - A specific level of spinning reserve is given
  - Technical limits are fulfilled (minimum/maximum outputs, ramp up/down)

Unit Commitment Problem. Data and Variables

DATA

- D_h demand during hour *h* [MW]
- R spinning reserve ratio related with demand [p.u.]
- a_t **linear coefficient of fuel variable cost of unit *t* [€/MWh]**
- b_t **fixed coefficient of fuel variable cost of unit *t* [€/h]**
- ca_t startup cost of unit *t* [€]
- cp_t shutdown cost of unit *t* [€]
- $\overline{P}_t$ maximum output of unit *t* [MW]
- $\underline{P}_t$ minimum output of unit *t* [MW]
- rs_t ramp up of unit *t* [MW/h]
- rb_t ramp down of unit *t* [MW/h]

VARIABLES

- P_{ht} output of unit *t* during hour *h* [MW]
- A_{ht} commitment of unit *t* during hour *h* {0,1}
- AR_{ht} start up of unit *t* during hour *h* {0,1}
- PR_{ht} shut down of unit *t* during hour *h* {0,1}

Unit Commitment Problem. Formulation

$$\min \sum_{h=1}^{H}\sum_{t=1}^{T}\left(a_t P_{ht} + b_t A_{ht} + ca_t AR_{ht} + cp_t PR_{ht}\right)$$

$$\sum_{t=1}^{T} P_{ht} = D_h \qquad H$$

$$\sum_{t=1}^{T}\left(\bar{P}_t A_{ht} - P_{ht}\right) = RD_h \qquad H$$

$$\underline{P}_t A_{ht} \le P_{ht} \le \bar{P}_t A_{ht} \qquad 2HT$$

$$A_{ht} - A_{h-1t} = AR_{ht} - PR_{ht} \qquad (H-1)T$$

$$P_{ht} - P_{h-1t} \le rs_t \qquad (H-1)T$$

$$P_{h-1t} - P_{ht} \le rb_t \qquad (H-1)T$$

$$P_{ht} \ge 0 \qquad A_{ht}, AR_{ht}, PR_{ht} \in \{0,1\}$$

4

Logical Propositions

Logical Propositions

- How to model the proposition: “if the product A is produced then the product B should be produced too”. The condition of the production of product *j* is modelled as $x_j \geq 1$. Then this proposition is mathematically written as $x_A \geq 1 \Rightarrow x_B \geq 1$

- This proposition cannot be included directly (with arrows) into the linear problem. In this example, one constraint $x_B \geq 1$ is included or not depending on the value of variable $x_A \geq 1$ (**endogenous problem**), modifying the structure of the problem.

Disjunctive Propositions (i)

- A couple of constraints where only one (either of the two) should be met, while the other one is not neccesary. Then, it should meet one constraint but not neccesarily both.

$$\boldsymbol{f}(\boldsymbol{x}) \leq 0 \quad \text{or} \quad \boldsymbol{g}(\boldsymbol{x}) \leq 0$$

Example of Disjunctive Propositions (ii)

- One of these two constraints should be met

$$3x_1 + 2x_2 \le 18 \quad \text{or} \quad x_1 + 4x_2 \le 16$$

- Adding *M* (high value constant) is equivalent to relaxing the constraint (for positive variables with positive coefficients)
  - Relax the constraint 1 and fulfill the 2

$$3x_1 + 2x_2 \le 18 + M$$
$$x_1 + 4x_2 \le 16$$

  - Relax the constraint 2 and fulfill the 1

$$3x_1 + 2x_2 \le 18$$
$$x_1 + 4x_2 \le 16 + M$$

- Using an auxiliary binary variable one of both is fulfilled and the other one is relaxed

$$3x_1 + 2x_2 \le 18 + M\delta$$
$$x_1 + 4x_2 \le 16 + M(1 - \delta)$$

$$\delta = \begin{cases} 1 & \text{if constraint 1 is relaxed} \\ 0 & \text{if constraint 2 is relaxed} \end{cases}$$

Fulfillment at least k of N constraints

- At least k of N ($k < N$) constraints should be met

$$f_1(x_1,\ldots,x_n) \le d_1$$
$$f_2(x_1,\ldots,x_n) \le d_2$$
$$\vdots$$
$$f_N(x_1,\ldots,x_n) \le d_N$$

- Using k = 1 and N = 2 is the disjunctive case
- Formulation:

$$f_1(x_1,\ldots,x_n) \le d_1 + M\delta_1$$
$$f_2(x_1,\ldots,x_n) \le d_2 + M\delta_2$$
$$\vdots$$
$$f_N(x_1,\ldots,x_n) \le d_N + M\delta_N$$
$$\sum_{i=1}^{N} \delta_i = N - k$$
$$\delta_i \in \{0,1\} \quad i = 1,\ldots,N$$

Selecting one from *N* values

- The equation should fulfill one of the possible values

$$f(x_1,\dots,x_n) = \begin{cases} d_1 \\ d_2 \\ \vdots \\ d_N \end{cases}$$

- Formulation:

$$f(x_1,\dots,x_n) = \sum_{i=1}^{N} d_i \delta_i$$

$$\sum_{i=1}^{N} \delta_i = 1$$

$$\delta_i \in \{0,1\} \quad i = 1,\dots,N$$

Simple Propositions (i)

- Using the previous constraint of the fixed cost $\boldsymbol{x} \le \boldsymbol{M}\delta$
- *M* is an upper positive bound of $\boldsymbol{x}$ and δ its associated binary variable.
  - If $\delta = 1$ the constraint is relaxed and is met by default $\boldsymbol{x} \le \boldsymbol{M}$
  - If $\delta = 0$ then $\boldsymbol{x} \le 0$
- So this constraint allows to model the proposition $\delta = 0 \Rightarrow \boldsymbol{x} \le 0$
- On the other hand, if $\boldsymbol{x} > 0$ then $\delta = 1$. If $\boldsymbol{x} \le 0$ the constraint does not imply anything $\boldsymbol{x} > 0 \Rightarrow \delta = 1$
- Both are equivalent propositions because $\boldsymbol{P} \rightarrow \boldsymbol{Q}$ is equivalent to $\text{No } \boldsymbol{Q} \rightarrow \text{No } \boldsymbol{P}$

$$\left.\begin{array}{l} \delta = 0 \Rightarrow \boldsymbol{x} \le 0 \\ \boldsymbol{x} > 0 \Rightarrow \delta = 1 \end{array}\right\} \boldsymbol{x} \le \boldsymbol{M}\delta$$

Simple Propositions (ii)

- Analogously the constraint $x \geq m\delta$ being *m* a negative lower bound of x and δ the binary variable.
  - If $\delta = 1$ the constraint does not imply anything as $x \geq m$ is fulfilled by default.
  - If $\delta = 0$ then $x \geq 0$. So, this constraint allows to model the proposition $\delta = 0 \Rightarrow x \geq 0$
- On the other hand, if $x < 0$ then $\delta = 1$. If $x \geq 0$ the constraint does not imply anything. $x < 0 \Rightarrow \delta = 1$
- Both propositions are equivalent as $P \rightarrow Q$ is equivalent to $\text{No } Q \rightarrow \text{No } P$

$$\left.\begin{array}{l} \delta = 0 \Rightarrow x \geq 0 \\ x < 0 \Rightarrow \delta = 1 \end{array}\right\} x \geq m\delta$$

Proposition of $\leq$ constraint (i)

- The proposition $\delta = 1 \rightarrow \sum_j a_j x_j \leq b$

  is modeled as $\sum_j a_j x_j \leq b + M(1-\delta)$

  being *M* a upper bound of the constraint for any value of x_j

  $$\sum_j a_j x_j - b \leq M$$

If $\delta = 1$ the original constraint is formulated and if $\delta = 0$ the original constraint is relaxed.

- Analogously to the previous case this constraint models the next proposition

  $$\sum_j a_j x_j > b \rightarrow \delta = 0$$

Proposition of $\leq$ constraint (ii)

- The proposition $$\sum_j a_j x_j \leq b \rightarrow \delta = 1$$

  can be replaced by $$\delta = 0 \rightarrow \sum_j a_j x_j > b$$

  or also by $$\delta = 0 \rightarrow \sum_j a_j x_j \geq b + \varepsilon$$

  Both are equivalent to $$\sum_j a_j x_j \geq b + \varepsilon + (m - \varepsilon)\delta$$

  Being *m* a lower bound of the constraint for any value of x_j

$$\sum_j a_j x_j - b \geq m$$

Proposition of $\geq$ constraint (i)

- Symmetrically propositions of upper or equal to can be modeled.
- The proposition $\delta = 1 \rightarrow \sum_j a_j x_j \geq b$

  is equivalent to $\sum_j a_j x_j \geq b + m(1-\delta)$

  being *m* a lower bound of the constraint for any value of. x_j

$$\sum_j a_j x_j - b \geq m$$

  If $\delta = 1$ the original constraint is formulated and if $\delta = 0$ the original constraint is relaxed.
- Analogously to the previous case this constraint models the next proposition

$$\sum_j a_j x_j < b \rightarrow \delta = 0$$

Proposition of $\geq$ constraint (ii)

- The proposition $\sum_j a_j x_j \geq b \rightarrow \delta = 1$

  can be replaced by $\delta = 0 \rightarrow \sum_j a_j x_j < b$

  or also by $\delta = 0 \rightarrow \sum_j a_j x_j \leq b - \varepsilon$

  Both are equivalent to $\sum_j a_j x_j \leq b - \varepsilon + (M + \varepsilon)\delta$

  Being *M* a upper bound of the constraint for any value of x_j

  $$\sum_j a_j x_j - b \leq M$$

Proposition of $=$ constraint (i)

- These equality constraints are replaced by constraints of upper than or equal to constraints and lower than and equal to constraints simultaneously.
- The proposition $\delta = 1 \rightarrow \sum_j a_j x_j = b$

  is equivalent to $\delta = 1 \rightarrow \sum_j a_j x_j \le b$

  $\delta = 1 \rightarrow \sum_j a_j x_j \ge b$
- The next two constraints model this equality

$$\sum_j a_j x_j \le b + M(1-\delta)$$

$$\sum_j a_j x_j \ge b + m(1-\delta)$$

- Effectively when $\delta = 1$ both constraints are fulfilled and when $\delta = 0$ both constraints are relaxed.

Proposition of $=$ constraint (ii)

- The proposition $\sum_j a_j x_j = b \rightarrow \delta = 1$

  is a combination of the previous cases
  $$\sum_j a_j x_j \leq b \rightarrow \delta' = 1$$
  $$\sum_j a_j x_j \geq b \rightarrow \delta'' = 1$$

  and besides $\delta' = 1 \text{ y } \delta'' = 1 \rightarrow \delta = 1$

  The resulting formulation:
  $$\sum_j a_j x_j \geq b + \varepsilon + (m - \varepsilon)\delta'$$
  $$\sum_j a_j x_j \leq b - \varepsilon + (M + \varepsilon)\delta''$$

  and additional constraint that models the fulfillment of previous constraints $\delta' + \delta'' - \delta \leq 1$

Double propositions

- Double propositions are split into two simple propositions.

$$\delta = 1 \leftrightarrow \sum_j \boldsymbol{a}_j \boldsymbol{x}_j \le \boldsymbol{b}$$ is equivalent to $$\begin{cases} \delta = 1 \rightarrow \sum_j \boldsymbol{a}_j \boldsymbol{x}_j \le \boldsymbol{b} \\ \sum_j \boldsymbol{a}_j \boldsymbol{x}_j \le \boldsymbol{b} \rightarrow \delta = 1 \end{cases}$$

and the same for other types of double propositions

Proposition Modeling Tables (i)

$\delta = 1 \rightarrow \sum_j a_j x_j \leq b$	$\sum_j a_j x_j \leq b + M(1-\delta)$
$\sum_j a_j x_j \leq b \rightarrow \delta = 1$	$\sum_j a_j x_j \geq b + \varepsilon + (m - \varepsilon)\delta$
$\delta = 1 \rightarrow \sum_j a_j x_j \geq b$	$\sum_j a_j x_j \geq b + m(1-\delta)$
$\sum_j a_j x_j \geq b \rightarrow \delta = 1$	$\sum_j a_j x_j \leq b - \varepsilon + (M + \varepsilon)\delta$
$\delta = 1 \rightarrow \sum_j a_j x_j = b$	$\sum_j a_j x_j \leq b + M(1-\delta)$ $\sum_j a_j x_j \geq b + m(1-\delta)$
$\sum_j a_j x_j = b \rightarrow \delta = 1$	$\sum_j a_j x_j \geq b + \varepsilon + (m - \varepsilon)\delta'$ $\sum_j a_j x_j \leq b - \varepsilon + (M + \varepsilon)\delta''$ $\delta' + \delta'' - \delta \leq 1$

Double Proposition Modeling Tables (ii)

$\delta = 1 \leftrightarrow \sum_j a_j x_j \le b$	$\sum_j a_j x_j \le b + M(1-\delta)$ $\sum_j a_j x_j \ge b + \varepsilon + (m - \varepsilon)\delta$
$\delta = 1 \leftrightarrow \sum_j a_j x_j \ge b$	$\sum_j a_j x_j \ge b + m(1-\delta)$ $\sum_j a_j x_j \le b - \varepsilon + (M + \varepsilon)\delta$
$\delta = 1 \leftrightarrow \sum_j a_j x_j = b$	$\sum_j a_j x_j \le b + M(1-\delta)$ $\sum_j a_j x_j \ge b + m(1-\delta)$ $\sum_j a_j x_j \ge b + \varepsilon + (m - \varepsilon)\delta'$ $\sum_j a_j x_j \le b - \varepsilon + (M + \varepsilon)\delta''$ $\delta' + \delta'' - \delta \le 1$

Equivalences between conditional and/or compounded propositions

- These equivalences are useful to transform implications before converting them into linear constraints

P → Q	not P or Q
P → (Q and R)	(P → Q) and (P → R)
P → (Q or R)	(P → Q) or (P → R)
(P and Q) → R	(P→ R) or (Q → R)
(P or Q) → R	(P→ R) and (Q → R)
not (P or Q)	not P and no Q
not (P and Q)	not P or not Q

Proposition equivalence

- A proposition can be formulated using disjunctive constraints

$$\boldsymbol{f}(\boldsymbol{x}) > 0 \Rightarrow \boldsymbol{g}(\boldsymbol{x}) \leq 0$$

$f(x) > 0$	$g(x) <= 0$	
T	**T**	**T**
T	F	F
F	**T**	**T**
F	F	**T**

- This is equivalent to

$$\boldsymbol{f}(\boldsymbol{x}) \leq 0 \quad \text{or} \quad \boldsymbol{g}(\boldsymbol{x}) \leq 0$$

$f(x) <= 0$	$g(x) <= 0$	
F	**T**	**T**
F	F	F
T	**T**	**T**
T	F	**T**

Simple Conditional and/or compounded Propositions

- X_i constraint *i*, δ_i binary variable that shows that constraint *i* is met
- The first row shows that constraint 1 or 2 (or both) should be met. So, at least one of the two variables δ_1 y δ_2 should be equal to 1. The linear equation is $\delta_1 + \delta_2 \geq 1$

X_1 o X_2	$\delta_1 + \delta_2 \geq 1$
X_1 y X_2	$\delta_1 = 1, \delta_2 = 1$
no X_1	$\delta_1 = 0$
$X_1 \rightarrow X_2$	$\delta_1 - \delta_2 \leq 0$
$X_1 \leftrightarrow X_2$	$\delta_1 - \delta_2 = 0$

- Besides there must be a constraint that says $\delta_i = 1$ that if the constraint *i* is met then $x_i > 0 \rightarrow \delta_i = 1$
- This proposition has been modeled for the fixed cost problem and its modeling equation was $\delta_i = 1$

Complex Conditional and/or compounded Propositions

- Complex Propositions are split into a double proposition to obtain linear constraints directly
- For example, $(X_A \text{ o } X_B) \rightarrow (X_C \text{ o } X_D \text{ o } X_E)$
  it is modeled as $\delta_A + \delta_B \geq 1 \rightarrow \delta_C + \delta_D + \delta_E \geq 1$

  and is transformed into a double proposition
  $$\delta_A + \delta_B \geq 1 \rightarrow \delta = 1 \rightarrow \delta_C + \delta_D + \delta_E \geq 1$$
  which is equivalent to
  $$\delta_A + \delta_B \geq 1 \rightarrow \delta = 1$$
  $$\delta = 1 \rightarrow \delta_C + \delta_D + \delta_E \geq 1$$
- Next, these equivalences are mathematically formulated

Equivalence Formulation (Example)

- If the product A is produced or the product B (or both) then at least one of the products C, D or E should be produced.
- X_i constraint of production of product *i*
- δ_i=1 binary variable to satisfy the constraint *i*

$$(\boldsymbol{X_A} \text{ o } \boldsymbol{X_B}) \rightarrow (\boldsymbol{X_C} \text{ o } \boldsymbol{X_D} \text{ o } \boldsymbol{X_E})$$

$$\delta_A + \delta_B \geq 1 \rightarrow \delta = 1$$
$$\delta = 1 \rightarrow \delta_C + \delta_D + \delta_E \geq 1$$

$$\delta_A + \delta_B - 2\delta \leq 0$$
$$-\delta_C - \delta_D - \delta_E + \delta \leq 0$$

- Formulation

$$\delta_A + \delta_B - 2\delta \leq 0$$
$$-\delta_C - \delta_D - \delta_E + \delta \leq 0$$

Alternative Formulation (Example)

$$(X_A \text{ o } X_B) \rightarrow (X_C \text{ o } X_D \text{ o } X_E)$$

equivalent to $[X_A \rightarrow (X_C \text{ o } X_D \text{ o } X_E)] \text{ y } [X_B \rightarrow (X_C \text{ o } X_D \text{ o } X_E$

$$\delta_A \geq 1 \rightarrow \delta_C + \delta_D + \delta_E \geq 1$$
$$\delta_B \geq 1 \rightarrow \delta_C + \delta_D + \delta_E \geq 1$$

$$\delta_A \geq 1 \rightarrow \delta = 1 \rightarrow \delta_C + \delta_D + \delta_E \geq 1$$
$$\delta_B \geq 1 \rightarrow \delta = 1 \rightarrow \delta_C + \delta_D + \delta_E \geq 1$$

- Formulation:

$$X_i \leq M\delta_i$$
$$\delta_A - \delta \leq 0$$
$$\delta_B - \delta \leq 0$$
$$-\delta_C - \delta_D - \delta_E + \delta \leq 0$$
$$\delta_i \in \{0,1\}, \delta \in \{0,1\}$$

Basket Team Problem

- A basket coach has 9 players that are ranked from 1 to 3 based on their skills on ball handle, shot, rebound and defense

Player	Positions	Handle	Shot	Rebound	Defense
1	Pivot	2	1	3	3
2	Guard	3	3	1	2
3	Pivot, Forward	2	3	2	2
4	Forward, Guard	1	3	3	1
5	Pivot, Forward	1	3	1	2
6	Forward,Guard	3	1	2	3
7	Pivot, Forward	3	2	2	1
8	Pivot	2	1	3	2
9	Forward	3	3	1	3

Basket Team (Logical Constraints)

- The team should be composed by 5 players that should have the maximum defense value satisfying the following conditions:
  1. At least two players should be able to play as pivot, two as forward and one as guard. Each player only plays in one position.
  2. Their average value of ball handle, shot and rebound should be upper or equal to 2.
  3. If the player 3 is selected, then the player 6 cannot be selected.
  4. If the player 1 is selected, the player 4 or the player 5 should be selected but not both. If the player 1 is not selected, players 4 and 5 may be selected.
  5. The player 8 or 9, but not both, should be selected.

- Formulate a linear problem to optimize the basket team.

$$\max 3x_1 + 2x_2 + 2x_3 + x_4 + 2x_5 + 3x_6 + x_7 + 2x_8 + 3x_9$$

$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 + x_9 = 5$$

$$x_1 + y_{3p} + y_{5p} + y_{7p} + x_8 \geq 2$$

$$y_{3a} + y_{4a} + y_{5a} + y_{6a} + y_{7a} + x_9 \geq 2$$

$$x_2 + y_{4b} + y_{6b} \geq 1$$

$$2x_1 + 3x_2 + 2x_3 + x_4 + x_5 + 3x_6 + 3x_7 + 2x_8 + 3x_9 \geq 10$$

$$x_1 + 3x_2 + 3x_3 + 3x_4 + 3x_5 + x_6 + 2x_7 + x_8 + 3x_9 \geq 10$$

$$3x_1 + x_2 + 2x_3 + 3x_4 + x_5 + 2x_6 + 2x_7 + 3x_8 + x_9 \geq 10$$

$x_3 = 1 \Rightarrow x_6 = 0$ is equivalent to $x_3 \leq 0$ or $x_6 \leq 0$

$$x_3 \leq y_1$$
$$x_6 \leq 1 - y_1$$

or alternatively $x_3 + x_6 \leq 1$

$x_1 \geq 1 \rightarrow x_4 + x_5 = 1$ is equivalent to $x_1 \leq 0$ ó $(x_4 + x_5 \leq 1$ y $x_4 + x_5 \geq 1)$

$$x_1 \leq y_2$$
$$x_4 + x_5 - 1 \leq (1 - y_2)$$
$$x_4 + x_5 - 1 \geq -1(1 - y_2)$$

or alternatively

$$x_4 + x_5 \leq 2 - x_1$$
$$x_4 + x_5 \geq x_1$$

$$x_8 + x_9 = 1$$

$$y_{3p} + y_{3a} - x_3 = 0$$

$$y_{4a} + y_{4b} - x_4 = 0$$

$$y_{5p} + y_{5a} - x_5 = 0$$

$$y_{6a} + y_{6b} - x_6 = 0$$

$$x_{7p} + x_{7a} - x_7 = 0$$

$$x_i, y_i \in \{0,1\}$$

- Two optimal solutions

$$x_1 = x_2 = x_3 = x_5 = x_8 = 1$$

$$x_1 = x_4 = x_6 = x_7 = x_9 = 1$$

and the rest of players = 0

Notation equivalence:

p:Pivot
a: Forward (alero)
b: Point Guard (base)

Product of Binary Variables

$\delta_1 \delta_2 = 0$ $\delta_i \in \{0,1\}$	$\delta_1 = 0$ o $\delta_2 = 0$	$\delta'_1 + \delta'_2 \geq 1$ $\delta_1 + \delta'_1 = 1$ $\delta_2 + \delta'_2 = 1$ $\delta_i, \delta'_i \in \{0,1\}$
$\delta_1 \delta_2$ $\delta_i \in \{0,1\}$	Reemplazar $\delta_1 \delta_2$ por δ_3 $\delta_3 = 1 \leftrightarrow \delta_1 = 1$ y $\delta_2 = 1$	$\delta_3 \leq \delta_1$ $\delta_3 \leq \delta_2$ $\delta_1 + \delta_2 \leq 1 + \delta_3$ $\delta_i \in \{0,1\}$
$\boldsymbol{x}\delta$ $\boldsymbol{x} \geq 0$ $\delta \in \{0,1\}$	Reemplazar $\boldsymbol{x}\delta$ por $\boldsymbol{y}$ $\delta = 0 \rightarrow \boldsymbol{y} = 0$ $\delta = 1 \rightarrow \boldsymbol{y} = \boldsymbol{x}$	$\boldsymbol{y} \geq 0$ $\boldsymbol{y} \leq \boldsymbol{M}\delta$ $-\boldsymbol{x} + \boldsymbol{y} \leq 0$ $\boldsymbol{x} - \boldsymbol{y} + \boldsymbol{M}\delta \leq \boldsymbol{M}$ $\boldsymbol{x} \leq \boldsymbol{M}$

- PROBLEM CLASSIFICATION
- SEVERAL CHARACTERISTIC PROBLEMS
- FIXED COST PROBLEM
- LOGICAL PROPOSITIONS
- **MINIMUM, MAXIMUM AND ABSOLUTE VALUE**
- PIECEWISE LINEAR (master)
- CONVEX AND NONCONVEX REGION (master)
- SPECIAL ORDERED SETS (master)
- REFORMULATION (master)

5

Minimum, Maximum and Absolute Value

Minimum or maximum of variables

$$\min z \qquad z = \max(x, y) \quad \Rightarrow \quad \begin{aligned} &\min z \\ &z \geq x \\ &z \geq y \end{aligned}$$

$$\max z \qquad z = \min(x, y) \quad \Rightarrow \quad \begin{aligned} &\max z \\ &z \leq x \\ &z \leq y \end{aligned}$$

Modeling the absolute value

$$z \le |x| \;\Rightarrow\; -x \le z \le x$$

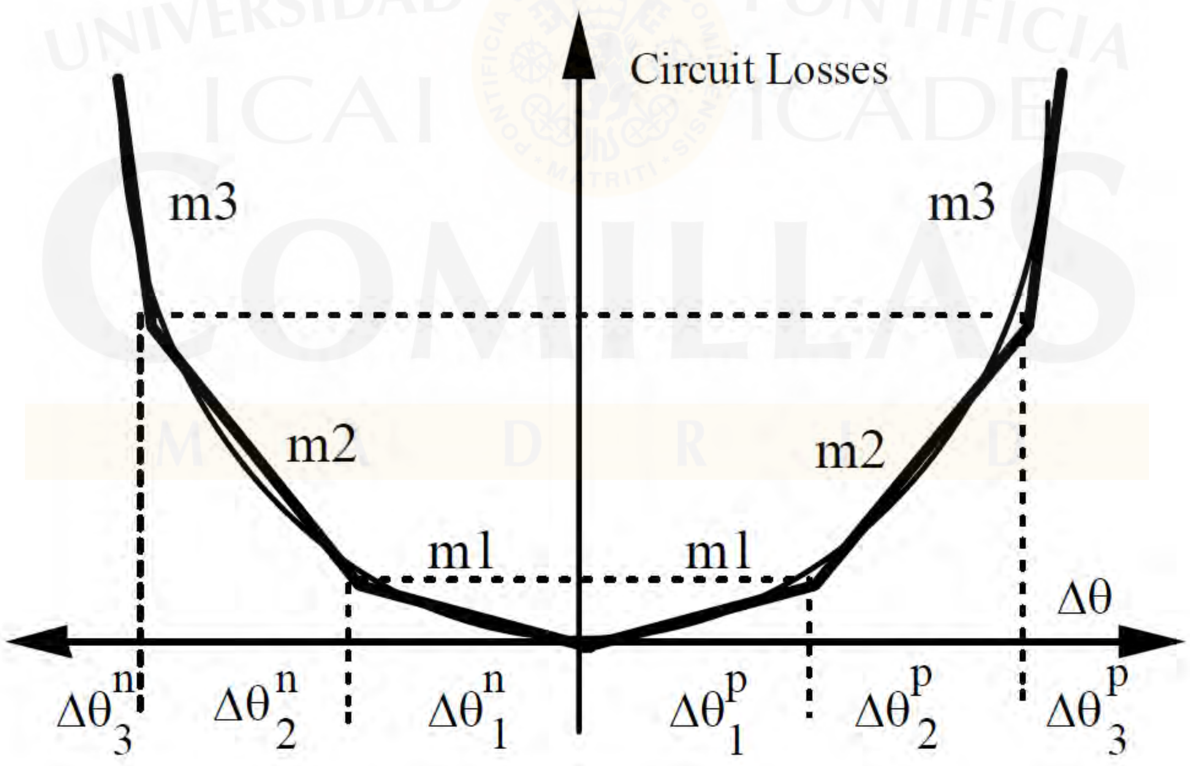
$$\begin{array}{ll} \min |x| & \Rightarrow \quad \min(x^+ + x^-) \\ s.t. & \quad s.t. \\ x \in X & \quad x = x^+ - x^- \\ & \quad x \in X \\ & \quad x^+, x^- \ge 0 \end{array}$$

6

Piecewise Linear (Master)

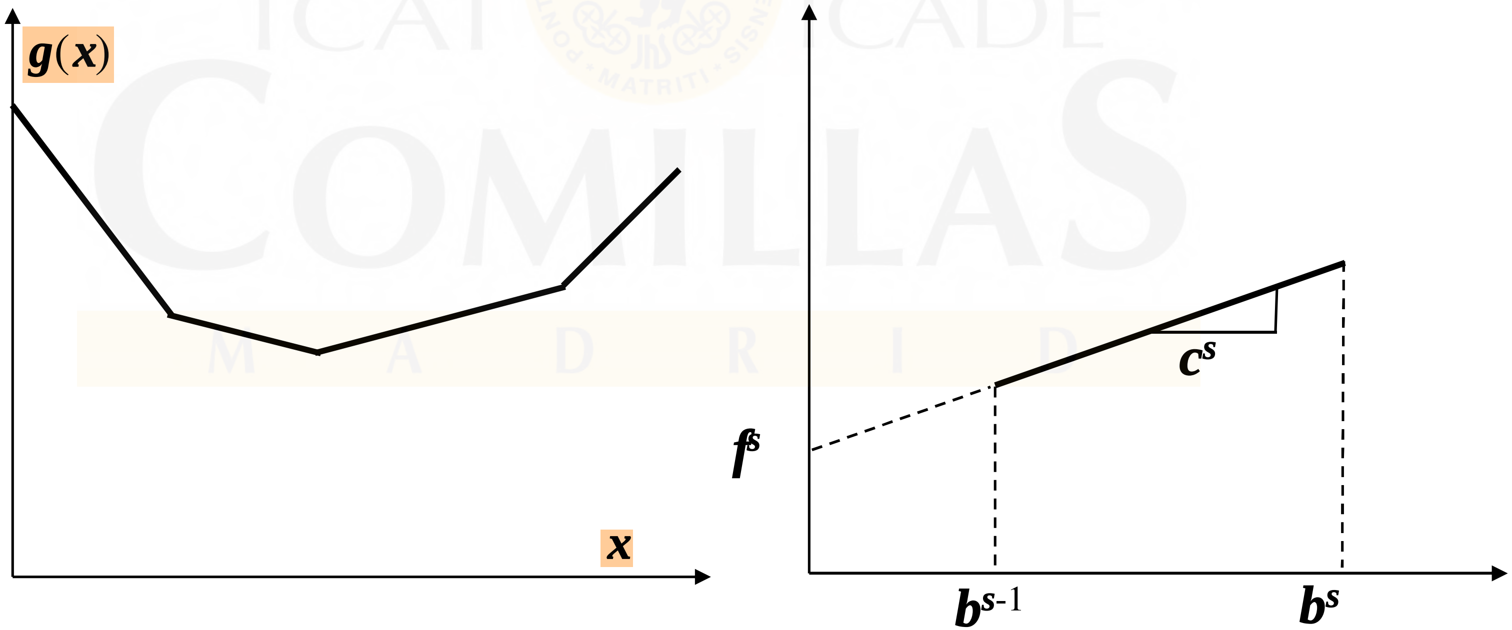
Piecewise linear function

- This function can appear when linearizing a nonlinear function
- Needed when it is a concave/convex function in a minimization/maximization problem



Modeling a piecewise linear function

- Piecewise linear function defined as a set of segments *s*
- Point is over the piecewise function $g(x)$ (equality constraint)
- It is assumed that the abscise of the first segment is the origin $b^0=0$



Three possible approaches

- Approaches
  1. Incremental
  2. Multiple selection
  3. Convex combination

- LP relaxations of the three approaches **are equivalent**
  - Any feasible solution of a relaxation corresponds to a feasible solution of the other ones with the same cost

Incremental modeling

- Define z^s as the load or usage of each segment
- Total load or value will be $x = \sum_s z^s$
- Segment *s*+1 has load 0 unless the previous one is full. If and only if $z^{s+1} > 0$ then $z^s = b^s - b^{s-1}$
- Introduce binary variables

$$y^s = \begin{cases} 1 & \text{if } z^s > 0 \\ 0 & \text{otherwise} \end{cases}$$

- Problem formulation

$$g(x) = \sum_s \left(c^s z^s + \hat{f}^s y^s \right)$$

$$x = \sum_s z^s$$

$$\left(b^s - b^{s-1}\right) y^{s+1} \le z^s \le \left(b^s - b^{s-1}\right) y^s$$

$$y^s \in \{0,1\}, y^{S+1} = 0$$

being $\hat{f}^s = \left(f^s + c^s b^{s-1} \right) - \left(f^{s-1} + c^{s-1} b^{s-1} \right)$ difference in cost in the intersection point of segments *s*-1 and *s*

Multiple selection modeling

- Define z^s as the total load if x is in this segment
- Total load or value will be $x = \sum_s z^s$
- If total load is in a segment, for this segment $z^s = x$ and for the other ones $z^s = 0$
- Introduce binary variables

$$y^s = \begin{cases} 1 & \text{if } z^s > 0 \\ 0 & \text{otherwise} \end{cases}$$

- Problem formulation

$$g(x) = \sum_s \left(c^s z^s + f^s y^s \right)$$

$$x = \sum_s z^s$$

$$b^{s-1} y^s \le z^s \le b^s y^s$$

$$\sum_s y^s \le 1$$

$$y^s \in \{0,1\}$$

Convex combination modeling

- Any point of the segment is a convex combination of their extremes with weights μ^s, λ^s
- Problem formulation

$$\boldsymbol{g}(\boldsymbol{x}) = \sum_{s} \left[\mu^s \left(\boldsymbol{c}^s \boldsymbol{b}^{s-1} + \boldsymbol{f}^s \right) + \lambda^s \left(\boldsymbol{c}^s \boldsymbol{b}^s + \boldsymbol{f}^s \right) \right]$$

$$\boldsymbol{x} = \sum_{s} \left(\mu^s \boldsymbol{b}^{s-1} + \lambda^s \boldsymbol{b}^s \right)$$

$$\mu^s + \lambda^s = \boldsymbol{y}^s$$

$$\sum_{s} \boldsymbol{y}^s \le 1$$

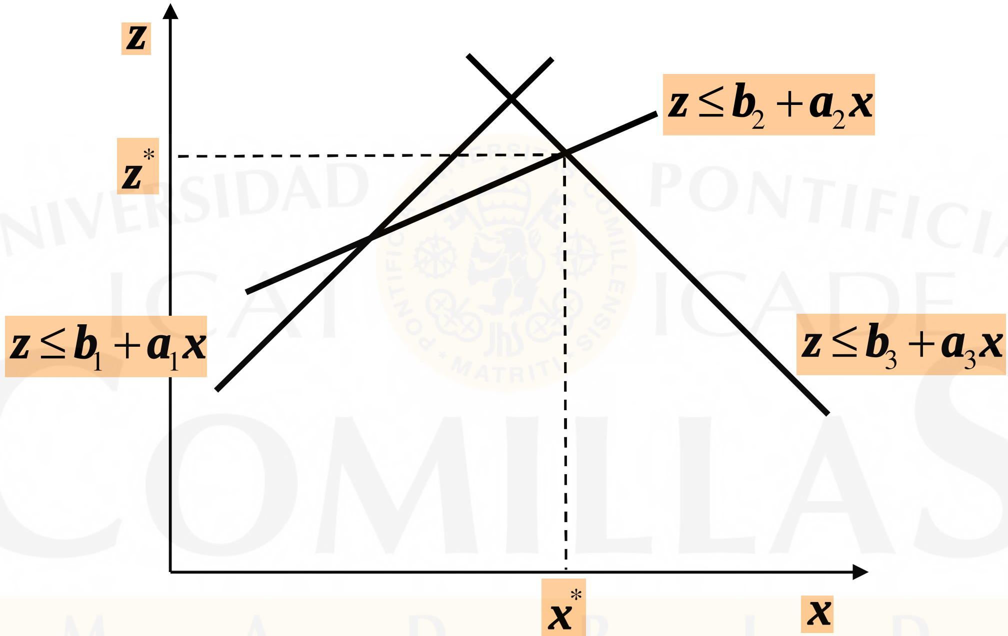
$$\mu^s, \lambda^s \ge 0, \boldsymbol{y}^s \in \{0,1\}$$

7

Convex and Non-Convex Regions (Master)

Maximizing an objective function. Concave region

- Optimization problem with a concave feasible region

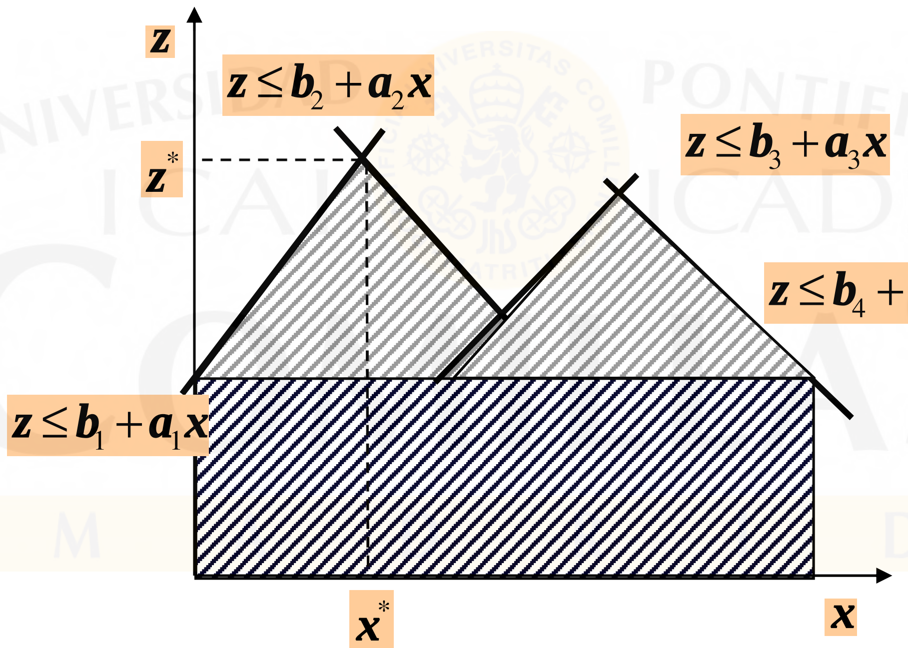


❑ LP Formulation

$$\max z$$
$$z \le b_1 + a_1 x$$
$$z \le b_2 + a_2 x$$
$$z \le b_3 + a_3 x$$
$$x, z \ge 0$$

Maximizing an objective function. Nonconcave region (i)

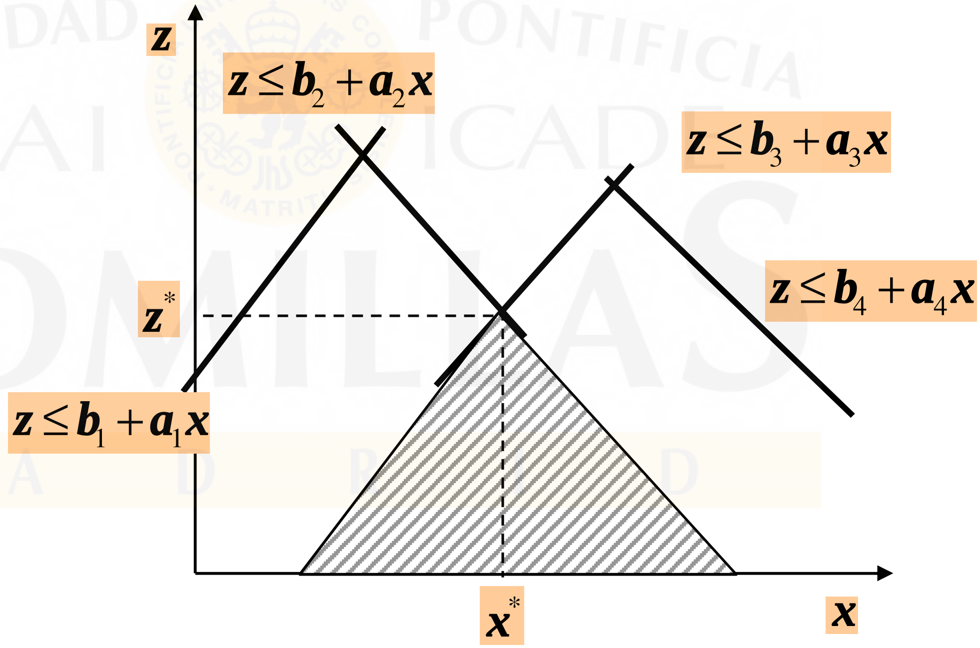
- Optimization problem with a nonconcave feasible region



Maximizing an objective function. Nonconcave region (ii)

- LP Formulation

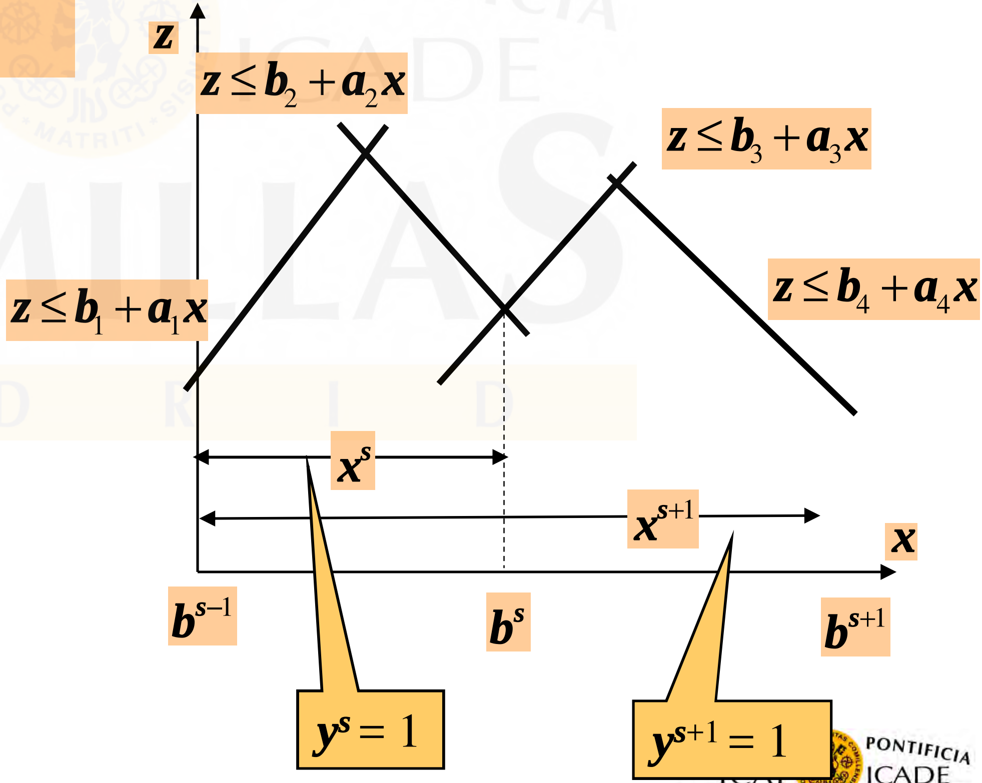
$$\max z$$
$$z \le b_1 + a_1 x$$
$$z \le b_2 + a_2 x$$
$$z \le b_3 + a_3 x$$
$$z \le b_4 + a_4 x$$
$$x, z \ge 0$$



Maximizing an objective function. Nonconcave region (iii)

- Divide nonconcave feasible region in concave feasible sub-regions and use a binary variable (multiple selection) to choose the feasible sub-region

$$y^s = \begin{cases} 1 & \text{if we are in sub-region } s \\ 0 & \text{otherwise} \end{cases}$$



Maximizing an objective function. Nonconcave region (iv)

$$\max z$$
$$z \le b_1 y^s + a_1 x^s + b_3 y^{s+1} + a_3 x^{s+1}$$
$$z \le b_2 y^s + a_2 x^s + b_4 y^{s+1} + a_4 x^{s+1}$$
$$x = \sum_s x^s$$
$$b^{s-1} y^s \le x^s \le b^s y^s \quad \forall s$$
$$\sum_s y^s \le 1$$
$$x, z, x^s \ge 0, y^s \in \{0,1\}$$

Sub-region s

Sub-region s+1

For every sub-region

- If we are in s

$$y^s = 1, y^{s+1} = 0$$

$$\max z$$
$$z \le b_1 + a_1 x^s$$
$$z \le b_2 + a_2 x^s$$
$$x = x^s$$
$$x^{s+1} = 0$$
$$b^{s-1} \le x^s \le b^s$$
$$x, z, x^s \ge 0$$

If we are in s+1

$$y^s = 0, y^{s+1} = 1$$

$$\max z$$
$$z \le b_3 + a_3 x^{s+1}$$
$$z \le b_4 + a_4 x^{s+1}$$
$$x = x^{s+1}$$
$$x^s = 0$$
$$b^s \le x^{s+1} \le b^{s+1}$$
$$x, z, x^{s+1} \ge 0$$

8

Special Ordered Sets (Master)

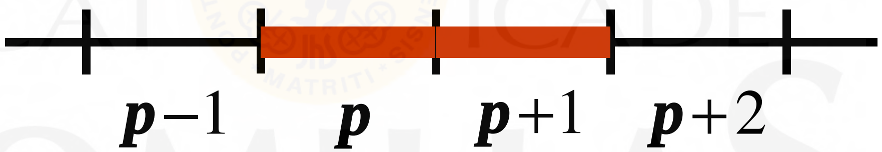
SOS1 and SOS2

- SOS1: set of variables in which a single variable must be different from 0

- SOS2: set of variables in which at most two variables must be different from 0 and must be consecutive
  - Example: maintenance scheduling of thermal units

Maintenance scheduling (i)

- **Assumption:**
  - Scheduled maintenance of each unit lasts for an integer number of periods.



- Maintenance scheduling involves **inter-period** variables and constraints:
  - Decisions taken in period *p* affect adjacent periods.

Maintenance scheduling (ii)

- **Relevant information** to decide the scheduled maintenance:
  - Maintenance duration for each thermal unit *t*: M_t (expressed in number of periods). Must be consecutive

Maintenance scheduling (iii)

- **Inter-period variables:**
  - Unit unavailable due to maintenance:

$$u_{pt} = \begin{cases} 1 & \text{unit } \boldsymbol{t} \text{ unavailable due to maintenance in period } \boldsymbol{p} \\ 0 & \text{otherwise} \end{cases}$$

  - Startup (beginning) and shutdown (end) of the maintenance period:

$$su_{pt} = \begin{cases} 1 & \text{maintenance of unit } \boldsymbol{t} \text{ begins in } \boldsymbol{p} \\ 0 & \text{otherwise} \end{cases}$$

$$sd_{pt} = \begin{cases} 1 & \text{maintenance of unit } \boldsymbol{t} \text{ ends in } \boldsymbol{p} \\ 0 & \text{otherwise} \end{cases}$$

Maintenance scheduling (iv)

- **Contiguity of maintenance periods:**
  - Formulation I:

$$\begin{cases} \sum_{p \le q < p+M_t} u_{qt} \ge M_t su_{pt} & \forall p,t \\ u_{pt} - u_{p-1t} \le su_{pt} & \forall p,t \\ \sum_{p} su_{pt} \le 1 & \forall t \end{cases}$$

Example: maintenance begins in p = 3 and lasts M_t = 4 periods:

$$su_{3t} = 1$$

$$\sum_{3 \le q \le 6} u_{qt} = u_{3t} + u_{4t} + u_{5t} + u_{6t} \ge 4 \Rightarrow u_{3t} = u_{4t} = u_{5t} = u_{6t} = 1$$

$$u_{2t} \le su_{2t} + u_{1t} = 0 + 0 = 0 \Rightarrow u_{2t} = 0$$

$$u_{3t} \le su_{3t} + u_{2t} = 1 + 0 = 1 \Rightarrow u_{3t} \le 1$$

Maintenance scheduling (v)

- **Contiguity of maintenance periods:**
  - Formulation II:

$$\begin{cases} su_{pt} = sd_{p+M_t} & \forall p,t \\ u_{pt} - u_{p-1t} = su_{pt} - sd_{pt} & \forall p,t \\ \sum_p \left(su_{pt} + sd_{pt} \right) \le 2 & \forall t \end{cases}$$

Example: maintenance begins in p = 3 and lasts M_t = 4 periods

$$su_{3t} = 1$$

$$su_{3t} - sd_{7t} = 0 \Rightarrow 1 - sd_{7t} = 0 \Rightarrow sd_{7t} = 1$$

$$u_{2t} - u_{3t} + su_{3t} - sd_{3t} = 0 \Rightarrow u_{2t} - u_{3t} + 1 - 0 = 0 \Rightarrow \begin{cases} u_{2t} = 0 \\ u_{3t} = 1 \end{cases}$$

Minimum uptime and downtime constraints

$$u_{pt} = \begin{cases} 1 & \text{unit } \boldsymbol{t} \text{ comitted in period } \boldsymbol{p} \\ 0 & \text{otherwise} \end{cases}$$

$$su_{pt} = \begin{cases} 1 & \text{stratup of unit } \boldsymbol{t} \text{ begins in } \boldsymbol{p} \\ 0 & \text{otherwise} \end{cases}$$

$$sd_{pt} = \begin{cases} 1 & \text{shutdown of unit } \boldsymbol{t} \text{ begins in } \boldsymbol{p} \\ 0 & \text{otherwise} \end{cases}$$

$$\sum_{p-UT_t+1 \le q \le p} su_{qt} \le u_{pt} \qquad \forall p,t$$

$$\sum_{p-DT_t+1 \le q \le p} su_{qt} \le 1 - u_{pt} - DT_t \qquad \forall p,t$$

- PROBLEM CLASSIFICATION
- SEVERAL CHARACTERISTIC PROBLEMS
- FIXED COST PROBLEM
- LOGICAL PROPOSITIONS
- MINIMUM, MAXIMUM AND ABSOLUTE VALUE
- PIECEWISE LINEAR (master)
- CONVEX AND NONCONVEX REGION (master)
- SPECIAL ORDERED SETS (master)
- **REFORMULATION (master)**

9

Reformulation (Master)

Reformulation

- Most MIP problems can be formulated in different ways
- In MIP problems, a good formulation is crucial to solve the model
- Good MIP formulation measure
  - Integrality gap difference between the objective function of the MIP and LP-relaxation solutions
- Given two equivalent MIP formulations, one is ***stronger*** (*better*) than the other, if the feasible region of the linear relaxation is strictly contained in the feasible region of the other one. Integrality gap is lower.

Warehouse location problem (no limits) (i)

- Choose where to locate warehouses among a set of locations and assign clients to the warehouses minimizing the total cost. No limits means that there are no limit in the number of clients assigned to a warehouse.
- Data

j locations

i clients

c_j localization cost in j

h_{ij} cost of satisfying the demand of client i from j

- Variables

$$y_j = \begin{cases} 1 & \text{warehouse located in } j \\ 0 & \text{otherwise} \end{cases}$$

x_{ij} fraction of demand of client i met from j

Warehouse location problem (no limits) (ii)

Formulation I

$$\min \sum_j c_j y_j + \sum_{ij} h_{ij} x_{ij}$$
$$\sum_j x_{ij} = 1 \quad \forall i$$
$$x_{ij} \le y_j \quad \forall ij$$
$$y_j \in \{0,1\}, x_{ij} \in [0,1]$$

Number of constraints: *I+IJ*

Formulation II

$$\min \sum_j c_j y_j + \sum_{ij} h_{ij} x_{ij}$$
$$\sum_j x_{ij} = 1 \quad \forall i$$
$$\sum_i x_{ij} \le M y_j \quad \forall j$$
$$y_j \in \{0,1\}, x_{ij} \in [0,1]$$

Number of constraints: *I+J*

- Both formulations are MIP equivalent. However, formulation I is stronger
- Intuitively as many constraints the worse. That's true in LP. However, in many MIP problems the more constraints the better.

Production problem with fixed and inventory costs (i)

- Data

  t time period

  c_t fixed cost, p_t variable cost, h_t inventory cost

  d_t demand

- Variables

  $$y_t = \begin{cases} 1 \text{ to produce} \\ 0 \text{ not produce} \end{cases}$$

  x_t amount produced

  s_t inventory at the end of the period

- Formulation I

$$\min \sum_t (c_t y_t + p_t x_t + h_t s_t)$$
$$s_{t-1} + x_t = d_t + s_t \quad \forall t$$
$$x_t \le M y_t \quad \forall t$$
$$s_0 = s_T = 0$$
$$x_t, s_t \ge 0, y_t \in \{0,1\}$$

Number of constraints: $2T$

Number of variables: $3T$

Production problem with fixed and inventory costs (ii)

- Variables

$$y_t = \begin{cases} 1 \text{ to produce} \\ 0 \text{ not produce} \end{cases}$$

q_{it} quantity produced in period i to met the demand in period $t \geq i$

- Formulation II

$$\min \sum_{t=1}^{T} \sum_{i=1}^{t} (p_i + h_i + h_{i+1} + \cdots + h_{t-1}) q_{it} + \sum_{t=1}^{T} c_t y_t$$

$$\sum_{i=1}^{t} q_{it} = d_t \quad \forall t$$

$$q_{it} \leq d_t y_i \quad \forall it$$

$$q_{it} \geq 0, y_t \in \{0,1\}$$

Number of constraints: $T+T^2/2$

Number of variables: $T+T^2/2$

- Formulation II is better. However, it has greater number of constraints and variables.

Reformulation criteria

- It can be interesting increase the number of variables if they can be used in the branching strategy of B&B. For example, "artificial" division of a zone in regions N, S, E and W to branch first in these zonal regions.
- Alternatively, introduce lazy constraints
- Avoid the use of big *M* parameters or put tight (lowest upper bound) values for the big *M*
- Alternative formulation for the Fixed Cost problem using SOS1 variable (at most one of the variables can be ≠ 0)
  - GAMS/CPLEX supports the use of an **indicator variable** $x \leq My$

$$\begin{gathered} \min Fy + Vx \\ x \leq My \\ x \geq 0 \\ y \in \{0,1\} \end{gathered}$$

$$\begin{gathered} \min Fy + Vx \\ y + z = 1 \\ x, y \in SOS1 \\ y, z \in \{0,1\} \end{gathered}$$

Some tips for MIP

- It can be interesting increase the number of variables if they can be used in the branching strategy of B&B. For example, "artificial" division of a zone in regions N, S, E and W to branch first in these zonal regions.
- Alternatively, introduce **lazy constraints** (only in GAMS/GUROBI)
- Avoid the use of big M parameters or put tight (lowest upper bound) values for the big M
- GAMS/CPLEX/GUROBI supports the use of an **indicator constraint** $x \le My$

$$\begin{gathered} \min Fy + Vx \\ x \le My \\ x \ge 0 \\ y \in \{0,1\} \end{gathered} \quad \longrightarrow \quad \begin{gathered} \min Fy + Vx \\ x \le 0 \\ x \ge 0 \\ y \in \{0,1\} \end{gathered}$$

Write in the file cplex.opt

```
indic constraint$y 0
```

Tight and compact unit commitment

- G. Gentile, G. Morales-España and A. Ramos *A Tight MIP Formulation of the Unit Commitment Problem with Start-up and Shut-down Constraints* EURO Journal on Computational Optimization 5 (1), 177–201 March 2017 10.1007/s13675-016-0066-y
- G. Morales-España, C.M. Correa-Posada, A. Ramos *Tight and Compact MIP Formulation of Configuration-Based Combined-Cycle Units* IEEE Transactions on Power Systems 31 (2), 1350-1359, March 2016 10.1109/TPWRS.2015.2425833
- G. Morales-España, J.M. Latorre, and A. Ramos *Tight and Compact MILP Formulation for the Thermal Unit Commitment Problem* IEEE Transactions on Power Systems 28 (4): 4897–4908, Nov 2013 10.1109/TPWRS.2012.2222938
- G. Morales-España, J.M. Latorre, and A. Ramos *Tight and Compact MILP Formulation of Start-Up and Shut-Down Ramping in Unit Commitment* IEEE Transactions on Power Systems 28 (2): 1288-1296, May 2013 10.1109/TPWRS.2012.2222938

Andrés Ramos

http://www.iit.comillas.edu/aramos/

Andres.Ramos@comillas.edu

Pedro Sánchez

Pedro.Sanchez@comillas.edu

Sonja Wogrin

Sonja.Wogrin@comillas.edu

Departamento de Organización Industrial
Alberto Aguilera, 23
28015 Madrid, Spain
Tel +34 91 542 28 00
Fax + 34 91 542 11 32
info-doi@doi.icai.upcomillas.es