MILP: Why do we need integer variables?

- Sometimes, the variables represent things that, because of its nature, can only take integer values: number of books to buy, number of facilities to be located, number of people to be hired.
- Logic constraints can be modeled via binary/integer variables.
- Some nonlinear models can be approximated using MILP.

Modeling tricks

Dealing with unrestricted variables

If $x_1, \ldots, x_r \in \mathbb{R}$ are unrestricted variables, and our algorithm only works with nonnegative variables, we can change:

$$x_i = x_i^1 - x_i^2, \; x_i^1, x_i^2 \geq 0.$$

This duplicates the number of variables. We can do better and introduce only one additional variable x^* which just move all the variable to the right:

$$x_i = x_i^* - x^*, \; x_i^*, x^* \geq 0.$$

Example

$$\begin{aligned} x_1 + x_2 &\leq 1 \\ 2x_1 - x_2 &\geq 3 \\ x_1, x_2 &\in \mathbb{R} \end{aligned} \quad \text{is equivalent to} \quad \begin{aligned} x_1^* - x^* + x_2^* - x^* &\leq 1 \\ 2(x_1^* - x^*) - (x_2^* - x^*) &\geq 3 \\ x_1^*, x_2^*, x^* &\geq 0 \end{aligned}$$

Modeling tricks

Converting linear equalities into linear inequalities

Using slack and surplus variables we can transform inequalities into equalities. But we can also do the opposite.

$$a_i^t x = b_i, i = 1 \ldots, m$$

can be transformed into

$$\begin{gathered} a_i^t x \leq b_i, i = 1 \ldots, m \\ \left(\sum_{i=1}^m a_i^t\right) x \geq \sum_{i=1}^m b_i \end{gathered}$$

Example

$$\begin{aligned} x_1 + x_2 &= 1 \\ 2x_1 - x_2 &= 3 \end{aligned} \quad \text{is equivalent to} \quad \begin{aligned} x_1 + x_2 &\leq 1 \\ 2x_1 - x_2 &\leq 3 \\ 3x_1 &\geq 4 \end{aligned}$$

Modeling tricks

Converting nonlinear objective functions into linear

$$\begin{array}{ll} \min & f(x) \\ s.t. & x \in X \end{array} \quad \text{is equivalent to} \quad \begin{array}{ll} \min & t \\ s.t. & f(x) \le t \\ & x \in X \end{array}$$

Modeling tricks

Dealing with absolute values

$$\begin{array}{ll} \min & \sum_{i=1}^{m} |f_i(x)| \\ s.t. & x \in X \end{array}$$

is equivalent to

$$\begin{array}{lll} \min & \sum_{i=1}^{m} t_i & \\ s.t. & x \in X & \\ & f_i(x) \leq t_i, & i = 1, \dots, m \\ & -f_i(x) \leq t_i, & i = 1, \dots, m \\ & t_i \geq 0, & i = 1, \dots, m \end{array}$$

Modeling tricks

Dealing with the max function

$$\begin{array}{ll} \min & \max_{i=1}^{m}\{f_i(x)\} \\ s.t. & x \in X \end{array}$$

is equivalent to

$$\begin{array}{ll} \min & t \\ s.t. & x \in X \\ & f_i(x) \leq t, i = 1, \ldots, m \end{array}$$

MILP: modeling tricks

Alternative sets of constraints

Consider two set of constraints

$$f_i^1(x) \leq b_i^1, i = 1, \ldots, m_1$$
$$f_i^2(x) \leq b_i^2, i = 1, \ldots, m_2$$

A set of constraints stating that at least one of the two above sets of constraints must be satisfied can be written as

$$f_i^1(x) - \delta_1 M_i^1 \leq b_i^1, i = 1, \ldots, m_1$$
$$f_i^2(x) - \delta_2 M_i^2 \leq b_i^2, i = 1, \ldots, m_2$$
$$\delta_1 + \delta_2 \leq 1$$
$$\delta_1, \delta_2 \in \{0, 1\}$$

provided that the parameters M_i^j satisfy
$f_i^j(x) \leq b_i^j + M_i^j, i = 1, \ldots, m_j, j = 1, 2$

MILP: modeling tricks

Alternative sets of constraints

Consider two set of constraints

$$f_i^1(x) \leq b_i^1, i = 1, \ldots, m_1$$
$$f_i^2(x) \leq b_i^2, i = 1, \ldots, m_2$$

A set of constraints stating that only one set of contraints must be satisfied can be written as

$$f_i^1(x) - \delta_1 M_i^1 \leq b_i^1, i = 1, \ldots, m_1$$
$$f_i^2(x) - \delta_2 M_i^2 \leq b_i^2, i = 1, \ldots, m_2$$
$$\delta_1 + \delta_2 = 1$$
$$\delta_1, \delta_2 \in \{0, 1\}$$

provided that the parameters M_i^j satisfy
$f_i^j(x) \leq b_i^j + M_i^j, i = 1, \ldots, m_j, j = 1, 2$

MILP: modeling tricks

Alternative sets of constraints

Consider two set of constraints

$$f_i^1(x) \leq b_i^1, i = 1, \ldots, m_1$$
$$f_i^2(x) \leq b_i^2, i = 1, \ldots, m_2$$

A set of constraints stating that only one set of contraints must be satisfied can be written as

$$f_i^1(x) - \delta_1 M_i^1 \leq b_i^1, i = 1, \ldots, m_1$$
$$f_i^2(x) - \delta_2 M_i^2 \leq b_i^2, i = 1, \ldots, m_2$$
$$\delta_1 + \delta_2 = 1$$
$$\delta_1, \delta_2 \in \{0, 1\}$$

provided that the parameters M_i^j satisfy
$f_i^j(x) \leq b_i^j + M_i^j, i = 1, \ldots, m_j, j = 1, 2$

This can be used to define nonconvex polygonal feasible sets.

MILP: modeling tricks

Conditional constraints 1

A conditional constraint of the form

$$f(x) > a \Longrightarrow g(x) \leq b$$

can be modeled with the alternative set of constraints

$$f(x) \leq a \quad \text{and/or} \quad g(x) \leq b$$

which in turn can be modeled as explained before (see more equivalences for conditional statements later on).

MILP: modeling tricks

K out of N constraints must hold

If we have a set of N constraints

$$f_1(x) \leq b_1, \ldots, f_N(x) \leq b_N$$

and only K out of the N constraints must hold, this can be modeled as follows:

$$f_1(x) \leq b_1 + M_1\delta_1$$
$$\ldots$$
$$f_N(x) \leq b_N + M_N\delta_N$$
$$\sum_{i=1}^{N} \delta_i = N - K$$
$$\delta_i \in \{0, 1\}, i = 1, \ldots, N$$

where M_i is an upper bound for $f_i(x) - b_i$.

MILP: modeling tricks

Modeling fixed costs

The discontinuous function to be minimized

$$\min f(x) = \begin{cases} 0 & \text{if } x = 0 \\ k + g(x) & \text{if } 0 < x \leq b \end{cases}$$

which sets a fixed cost k in case the variable x is used (in case $x > 0$) can be written as

$$\begin{aligned} \min \quad & k\delta + g(x) \\ s.t. \quad & x \leq b\delta \\ & x \geq 0 \\ & \delta \in \{0, 1\} \end{aligned}$$

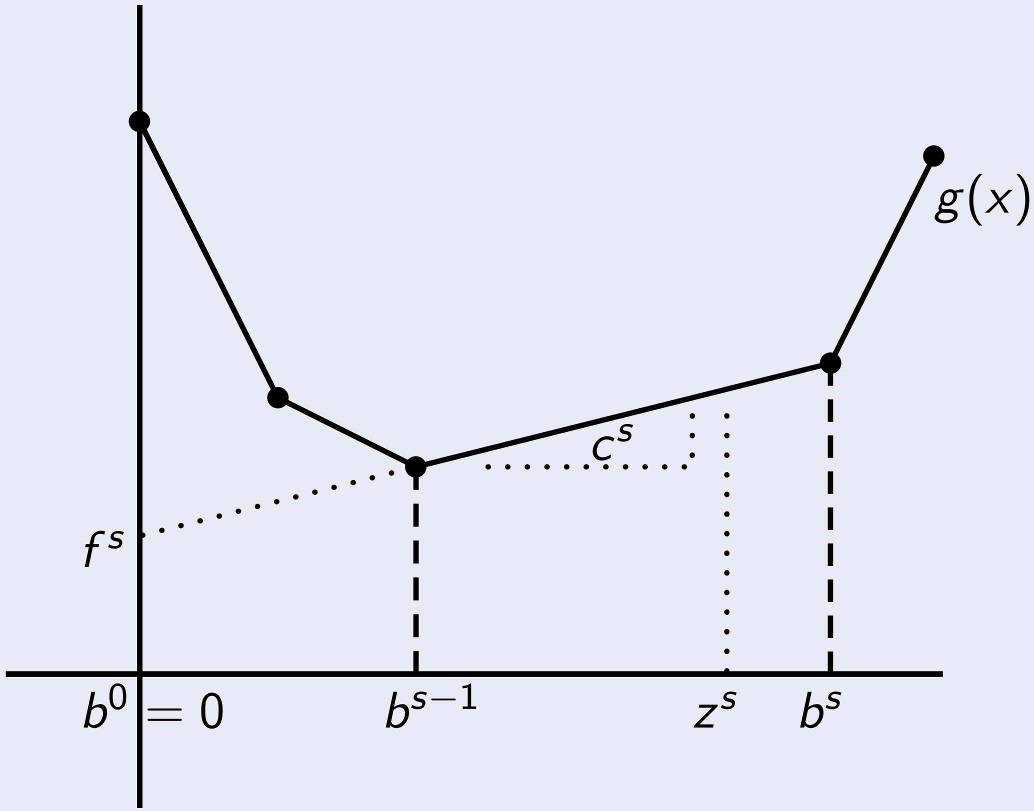
Notice that

$$\delta = \begin{cases} 0 & \text{if } x = 0 \\ 1 & \text{if } x > 0 \end{cases}$$

MILP: modeling tricks

Modeling a piecewise linear function

Consider the piecewise linear function $g(x)$ of the picture. Assume that there are $p+1$ breaking points, $b^0, \dots, b^p$. The slope of the s-th segment $[b^{s-1}, b^s]$ will be denoted by c^s, and the point where the line containing that segment cuts the $0Y$-axis by f^s. Then, the value of $g(x)$ at a point z^s on that segment is given by $g(z^s) = f^s + c^s z^s$.



MILP: modeling tricks

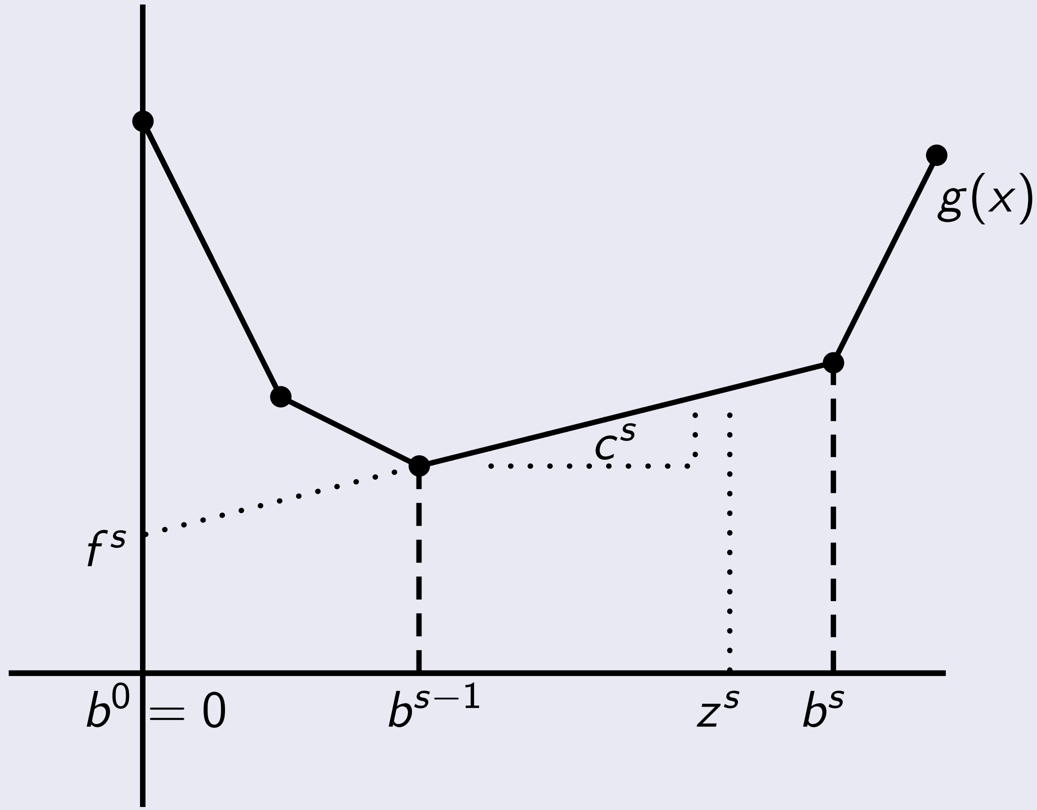
Modeling a piecewise linear function 1

Let us denote

$$z^s = \begin{cases} x & \text{if } x \in [b^{s-1}, b^s] \\ 0 & \text{otherwise} \end{cases} \text{ and } \delta_s = \begin{cases} 1 & \text{if } z^s > 0 \\ 0 & \text{otherwise} \end{cases}, s = 1 \ldots, p$$

Then function $g(x)$ can be rewritten as follows:

$$g(x) = \sum_{s=1}^{p} (c^s z^s + f^s \delta_s)$$
$$x = \sum_{s=1}^{p} z^s$$
$$b^{s-1}\delta_s \le z^s \le b^s \delta_s$$
$$\sum_{s=1}^{p} \delta_s = 1$$
$$\delta_s \in \{0, 1\}, s = 1 \ldots, p$$



MILP: modeling tricks

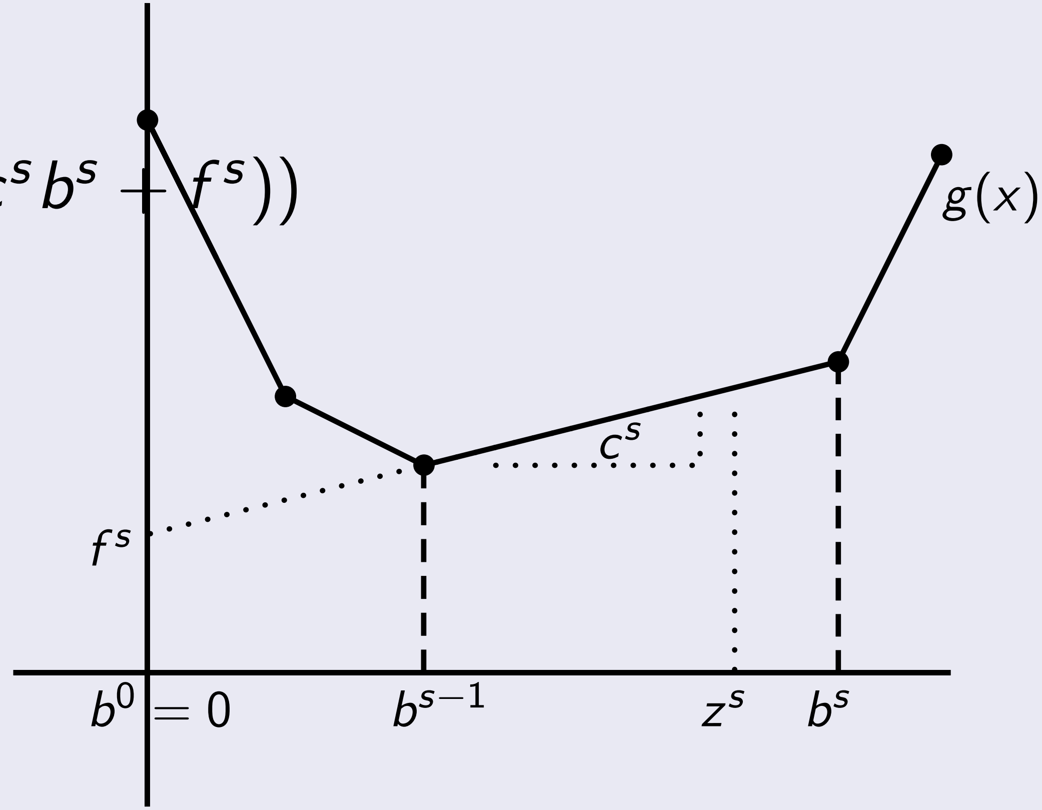
Modeling a piecewise linear function 2

Alternatively, since each point $z^s \in [b^{s-1}, b^s]$ may be written as a convex combination of its end points, $(b^{s-1}, c^s b^{s-1} + f^s)$ and $(b^s, c^s b^s + f^s)$,

$$(z^s, g(z^s)) = \lambda_s(b^{s-1}, c^s b^{s-1} + f^s) + \mu_s(b^s, c^s b^s + f^s), \ \lambda_s + \mu_s = 1$$

we can also rewrite the function $g(x)$ as follows:

$$g(x) = \sum_{s=1}^{p}(\lambda_s(c^s b^{s-1} + f^s) + \mu_s(c^s b^s + f^s))$$
$$x = \sum_{s=1}^{p}(\lambda_s b^{s-1} + \mu_s b^s)$$
$$\lambda_s + \mu_s = \delta_s$$
$$\sum_{s=1}^{p} \delta_s = 1$$
$$\delta_s \in \{0,1\}$$
$$\lambda_s, \mu_s \geq 0, i = 1, \ldots, p$$



MILP: modeling tricks

A function must take a value out of N possible values

$$f(x) = b_1 \vee b_2 \vee \ldots \vee b_N$$

can be modeled as

$$f(x) = \sum_{i=1}^{N} b_i \delta_i$$

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

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

MILP: modeling tricks

Transforming integer variables into binary variables

Assume that

$$0 \leq x \leq u, z \in \mathbb{Z}.$$

If $2^N \leq u \leq 2^{N+1}$ then we can represent x using binary variables as follows:

$$x = \sum_{i=0}^{N} 2^i \delta_i, \quad \delta_i \in \{0, 1\}, i = 1 \ldots, N$$

MILP: modeling tricks

Linearizing the product of two binary variables

Let $y_1, y_2 \in \{0, 1\}$ two binary variables, and assume that its product, $y_1 y_2$, which is a nonlinear expression, appears in a given formulation. We can linearize the product as follows:

$$\delta \leq y_1$$
$$\delta \leq y_2$$
$$\delta \geq y_1 + y_2 - 1$$
$$\delta \in \{0, 1\}$$

Notice that $\delta = y_1 y_2$.

MILP: modeling tricks

Linearizing the product of a binary and a continous variable

Let z be a continuous variable such that $L \leq z \leq U$, and $x \in \{0, 1\}$ be a binary variable. Assume that its product, zx, which is a nonlinear expression, appears in a given formulation. We can linearize the product as follows:

$$\begin{aligned} y &\leq Ux \\ y &\geq Lx \\ z - y &\leq U(1 - x) \\ z - y &\geq L(1 - x) \end{aligned}$$

Notice that $y = zx$

MILP: modeling tricks

Example: A discrete competitive location problem under the probabilistic choice rule.

- A chain wants to enter in a given area by opening p facilities.
- Those facilities are to be open in p of the s potential sites pre-selected by the chain.
- There already exists m competing facilities operating in the area.
- Customers follow a probabilistic choice rule (they patronize all the facilities, and the amount spent at each facility is proportional to its attraction).
- The objective is to maximize the market share captured by the locating chain.

MILP: modeling tricks

Example: A discrete competitive location problem under the probabilistic choice rule.

Indices

i index for demand points (or customers), $i = \{1, \ldots, n\}$.

j index for the facilities,

$j = 1, \ldots, s$, for the potential new facilities,

$j = s + 1, \ldots, s + m$, for the existing competing facilities.

Data

w_i demand (or buying power) of demand point i.

d_{ij} distance between demand point i and location j.

a_{ij} quality of facility j as perceived by deman point i.

β modulator of the distance

MILP: modeling tricks

Example: A discrete competitive location problem under the probabilistic choice rule.

Computed data

$$u_{ij} = \frac{a_{ij}}{(d_{ij}+1)^{\beta}}$$ attraction that demand point i feels towards facility j.

Variables

$$x_j = \begin{cases} 1 & \text{if a facility is open at } j \\ 0 & \text{otherwise} \end{cases}, j = 1\ldots, s$$

MILP: modeling tricks

Example: A discrete competitive location problem under the probabilistic choice rule.

$$\max \quad \sum_{i=1}^{n} w_i \frac{\sum_{j=1}^{s} u_{ij} x_j}{\sum_{j=1}^{s} u_{ij} x_j + \sum_{j=s+1}^{s+m} u_{ij}}$$

$$\text{s.t.} \quad \sum_{j=1}^{s} x_j = p$$

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

MILP: modeling tricks

Example: A discrete competitive location problem under the probabilistic choice rule.

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{n} w_i \frac{\sum_{j=1}^{s} u_{ij} x_j}{\sum_{j=1}^{s} u_{ij} x_j + \sum_{j=s+1}^{s+m} u_{ij}} \\
\text{s.t.} \quad & \sum_{j=1}^{s} x_j = p \\
& x_j \in \{0,1\}, j = 1, \ldots, s
\end{aligned}
$$

If we denote

$$
z_i = \frac{1}{\sum_{j=1}^{s} u_{ij} x_j + \sum_{j=s+1}^{s+m} u_{ij}}, i = 1, \ldots, n
$$

then the problem becomes

MILP: modeling tricks

Example: A discrete competitive location problem under the probabilistic choice rule.

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{n} w_i z_i \sum_{j=1}^{s} u_{ij} x_j \\
\text{s.t.} \quad & z_i = \frac{1}{\sum_{j=1}^{s} u_{ij} x_j + \sum_{j=s+1}^{s+m} u_{ij}}, i = 1, \ldots, n \\
& \sum_{j=1}^{s} x_j = p \\
& x_j \in \{0, 1\}, j = 1, \ldots, s \\
& z_i \geq 0, i = 1, \ldots, n
\end{aligned}
$$

MILP: modeling tricks

Example: A discrete competitive location problem under the probabilistic choice rule.

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{n} \sum_{j=1}^{s} w_i z_i u_{ij} x_j \\
\text{s.t.} \quad & z_i = \frac{1}{\sum_{j=1}^{s} u_{ij} x_j + \sum_{j=s+1}^{s+m} u_{ij}}, i = 1, \ldots, n \\
& \sum_{j=1}^{s} x_j = p \\
& x_j \in \{0, 1\}, j = 1, \ldots, s \\
& z_i \geq 0, i = 1, \ldots, n
\end{aligned}
$$

MILP: modeling tricks

Example: A discrete competitive location problem under the probabilistic choice rule.

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{n} \sum_{j=1}^{s} (w_i z_i u_{ij}) x_j \\
\text{s.t.} \quad & z_i = \frac{1}{\displaystyle\sum_{j=1}^{s} u_{ij} x_j + \sum_{j=s+1}^{s+m} u_{ij}}, i = 1, \ldots, n \\
& \sum_{j=1}^{s} x_j = p \\
& x_j \in \{0, 1\}, j = 1, \ldots, s \\
& z_i \geq 0, i = 1, \ldots, n
\end{aligned}
$$

MILP: modeling tricks

Example: A discrete competitive location problem under the probabilistic choice rule.

If we denote

$$y_{ij} = (w_i z_i u_{ij})x_j, i = 1, \ldots, n, j = 1, \ldots, s$$

and taking into account that the product $y = zx$, where $L \leq z \leq U$ is continuous and x binary can be linearized as

$$\begin{aligned} y &\leq Ux \\ y &\geq Lx \\ z - y &\leq U(1-x) \\ z - y &\geq L(1-x) \end{aligned}$$

we have that the product $y_{ij} = (w_i z_i u_{ij})x_j$ can be linearized as follows

$$\left.\begin{array}{l} y_{ij} \leq w_i x_j, \\ y_{ij} \geq 0 x_j \Leftrightarrow y_{ij} \geq 0, \\ w_i z_i u_{ij} - y_{ij} \leq w_i(1-x_j), \\ w_i z_i u_{ij} - y_{ij} \geq 0(1-x_j) \Leftrightarrow w_i z_i u_{ij} - y_{ij} \geq 0, \end{array}\right\} i = 1, \ldots, n, j = 1, \ldots, s$$

MILP: modeling tricks

Example: A discrete competitive location problem under the probabilistic choice rule.

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{n} \sum_{j=1}^{s} y_{ij} \\
\text{s.t.} \quad & z_i = \frac{1}{\sum_{j=1}^{s} u_{ij} x_j + \sum_{j=s+1}^{s+m} u_{ij}}, && i = 1, \ldots, n \\
& y_{ij} \leq w_i x_j, && i = 1, \ldots, n, j = 1, \ldots, s \\
& y_{ij} \geq 0, && i = 1, \ldots, n, j = 1, \ldots, s \\
& w_i z_i u_{ij} - y_{ij} \leq w_i (1 - x_j), && i = 1, \ldots, n, j = 1, \ldots, s \\
& w_i z_i u_{ij} - y_{ij} \geq 0, && i = 1, \ldots, n, j = 1, \ldots, s \\
& \sum_{j=1}^{s} x_j = p \\
& x_j \in \{0, 1\}, && j = 1, \ldots, s \\
& z_i \geq 0, && i = 1, \ldots, n \\
& y_{ij} \geq 0, && i = 1, \ldots, n, j = 1, \ldots, s
\end{aligned}
$$

MILP: modeling tricks

Example: A discrete competitive location problem under the probabilistic choice rule.

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{n}\sum_{j=1}^{s} y_{ij} & \\
\text{s.t.} \quad & z_i = \frac{1}{\sum_{j=1}^{s} u_{ij}x_j + \sum_{j=s+1}^{s+m} u_{ij}}, & i = 1,\ldots,n \\
& y_{ij} \le w_i x_j, & i = 1,\ldots,n, j = 1,\ldots,s \\
& y_{ij} \ge 0, & i = 1,\ldots,n, j = 1,\ldots,s \\
& w_i z_i u_{ij} - y_{ij} \le w_i(1 - x_j), & i = 1,\ldots,n, j = 1,\ldots,s \\
& w_i z_i u_{ij} - y_{ij} \ge 0, & i = 1,\ldots,n, j = 1,\ldots,s \\
& \sum_{j=1}^{s} x_j = p & \\
& x_j \in \{0,1\}, & j = 1,\ldots,s \\
& z_i \ge 0, & i = 1,\ldots,n \\
& y_{ij} \ge 0, & i = 1,\ldots,n, j = 1,\ldots,s
\end{aligned}
$$

MILP: modeling tricks

Example: A discrete competitive location problem under the probabilistic choice rule.

$$z_i = \frac{1}{\sum_{j=1}^{s} u_{ij} x_j + \sum_{j=s+1}^{s+m} u_{ij}}$$

MILP: modeling tricks

Example: A discrete competitive location problem under the probabilistic choice rule.

$$z_i = \frac{1}{\sum_{j=1}^{s} u_{ij} x_j + \sum_{j=s+1}^{s+m} u_{ij}} \Leftrightarrow$$

$$z_i \left(\sum_{j=1}^{s} u_{ij} x_j + \sum_{j=s+1}^{s+m} u_{ij}\right) = 1$$

MILP: modeling tricks

Example: A discrete competitive location problem under the probabilistic choice rule.

$$z_i = \frac{1}{\sum_{j=1}^{s} u_{ij}x_j + \sum_{j=s+1}^{s+m} u_{ij}} \Leftrightarrow$$

$$z_i(\sum_{j=1}^{s} u_{ij}x_j + \sum_{j=s+1}^{s+m} u_{ij}) = 1 \Leftrightarrow$$

$$z_i\sum_{j=1}^{s} u_{ij}x_j + z_i\sum_{j=s+1}^{s+m} u_{ij} = 1$$

MILP: modeling tricks

Example: A discrete competitive location problem under the probabilistic choice rule.

$$z_i = \frac{1}{\sum_{j=1}^{s} u_{ij} x_j + \sum_{j=s+1}^{s+m} u_{ij}} \Leftrightarrow$$

$$z_i\left(\sum_{j=1}^{s} u_{ij} x_j + \sum_{j=s+1}^{s+m} u_{ij}\right) = 1 \Leftrightarrow$$

$$z_i \sum_{j=1}^{s} u_{ij} x_j + z_i \sum_{j=s+1}^{s+m} u_{ij} = 1 \Leftrightarrow$$

$$w_i z_i \sum_{j=1}^{s} u_{ij} x_j + w_i z_i \sum_{j=s+1}^{s+m} u_{ij} = w_i$$

MILP: modeling tricks

Example: A discrete competitive location problem under the probabilistic choice rule.

$$z_i = \frac{1}{\sum_{j=1}^{s} u_{ij} x_j + \sum_{j=s+1}^{s+m} u_{ij}} \Leftrightarrow$$

$$z_i \left(\sum_{j=1}^{s} u_{ij} x_j + \sum_{j=s+1}^{s+m} u_{ij}\right) = 1 \Leftrightarrow$$

$$z_i \sum_{j=1}^{s} u_{ij} x_j + z_i \sum_{j=s+1}^{s+m} u_{ij} = 1 \Leftrightarrow$$

$$w_i z_i \sum_{j=1}^{s} u_{ij} x_j + w_i z_i \sum_{j=s+1}^{s+m} u_{ij} = w_i \Leftrightarrow$$

$$\sum_{j=1}^{s} w_i z_i u_{ij} x_j + w_i z_i \sum_{j=s+1}^{s+m} u_{ij} = w_i$$

MILP: modeling tricks

Example: A discrete competitive location problem under the probabilistic choice rule.

$$z_i = \frac{1}{\sum_{j=1}^{s} u_{ij} x_j + \sum_{j=s+1}^{s+m} u_{ij}} \Leftrightarrow$$

$$z_i \left(\sum_{j=1}^{s} u_{ij} x_j + \sum_{j=s+1}^{s+m} u_{ij}\right) = 1 \Leftrightarrow$$

$$z_i \sum_{j=1}^{s} u_{ij} x_j + z_i \sum_{j=s+1}^{s+m} u_{ij} = 1 \Leftrightarrow$$

$$w_i z_i \sum_{j=1}^{s} u_{ij} x_j + w_i z_i \sum_{j=s+1}^{s+m} u_{ij} = w_i \Leftrightarrow$$

$$\sum_{j=1}^{s} w_i z_i u_{ij} x_j + w_i z_i \sum_{j=s+1}^{s+m} u_{ij} = w_i \Leftrightarrow$$

$$\sum_{j=1}^{s} y_{ij} + w_i z_i \sum_{j=s+1}^{s+m} u_{ij} = w_i$$

$$
\begin{array}{lll}
\max & \displaystyle\sum_{i=1}^{n}\sum_{j=1}^{s} y_{ij} & \\
\text{s.t.} & \sum_{j=1}^{s} y_{ij} + w_i z_i \sum_{j=s+1}^{s+m} u_{ij} = w_i, & i = 1,\dots,n \\
 & y_{ij} \le w_i x_j, & i = 1,\dots,n, j = 1,\dots,s \\
 & y_{ij} \ge 0, & i = 1,\dots,n, j = 1,\dots,s \\
 & w_i z_i u_{ij} - y_{ij} \le w_i(1 - x_j), & i = 1,\dots,n, j = 1,\dots,s \\
 & w_i z_i u_{ij} - y_{ij} \ge 0 & i = 1,\dots,n, j = 1,\dots,s \\
 & \displaystyle\sum_{j=1}^{s} x_j = p & \\
 & x_j \in \{0,1\}, & j = 1,\dots,s \\
 & z_i \ge 0, & i = 1,\dots,n \\
 & y_{ij} \ge 0, & i = 1,\dots,n, j = 1,\dots,s
\end{array}
$$

MILP: modeling tricks

Example: A discrete competitive location problem under the probabilistic choice rule.

$$
\begin{array}{lll}
\max & \displaystyle\sum_{i=1}^{n}\sum_{j=1}^{s} y_{ij} & \\
\text{s.t.} & \sum_{j=1}^{s} y_{ij} + w_i z_i \sum_{j=s+1}^{s+m} u_{ij} \leq w_i, & i = 1,\ldots,n \\
& y_{ij} \leq w_i x_j, & i = 1,\ldots,n, j = 1,\ldots,s \\
& y_{ij} \geq 0, & i = 1,\ldots,n, j = 1,\ldots,s \\
& w_i z_i u_{ij} - y_{ij} \leq w_i(1 - x_j), & i = 1,\ldots,n, j = 1,\ldots,s \\
& w_i z_i u_{ij} - y_{ij} \geq 0 & i = 1,\ldots,n, j = 1,\ldots,s \\
& \displaystyle\sum_{j=1}^{s} x_j = p & \\
& x_j \in \{0,1\}, & j = 1,\ldots,s \\
& z_i \geq 0, & i = 1,\ldots,n \\
& y_{ij} \geq 0, & i = 1,\ldots,n, j = 1,\ldots,s
\end{array}
$$

MILP: modeling tricks

Let x be a continuous variable such that $L \leq x \leq U$. And let $\delta \in \{0, 1\}$ be a binary variable.

MILP: modeling tricks

Let x be a continuous variable such that $L \leq x \leq U$. And let $\delta \in \{0, 1\}$ be a binary variable.

Conditional constraints 2

$$\delta = 0 \Longrightarrow x \leq 0$$

can be modeled as

$$x \leq \delta U.$$

Since $P \Rightarrow Q$ is equivalent to $\neg Q \Rightarrow \neg P$ the previous expression also models

$$x > 0 \Longrightarrow \delta = 1$$

MILP: modeling tricks

Let x be a continuous variable such that $L \leq x \leq U$. And let $\delta \in \{0, 1\}$ be a binary variable.

Conditional constraints 3

$$\delta = 0 \Longrightarrow x \geq 0$$

can be modeled as

$$x \geq \delta L.$$

Since $P \Rightarrow Q$ is equivalent to $\neg Q \Rightarrow \neg P$ the previous expression also models

$$x < 0 \Longrightarrow \delta = 1$$

MILP: modeling tricks

Let $\epsilon > 0$ be a small number, and m and M two constants such that $m \leq f(x) - b \leq M$ for any feasible value of x. And let $\delta \in \{0, 1\}$ be a binary variable.

MILP: modeling tricks

Let $\epsilon > 0$ be a small number, and m and M two constants such that $m \leq f(x) - b \leq M$ for any feasible value of x. And let $\delta \in \{0, 1\}$ be a binary variable.

Conditional constraints 4 (type $\leq$)

$$\delta = 1 \Longrightarrow f(x) \leq b$$

can be modeled as

$$f(x) \leq b + M(1 - \delta).$$

Since $P \Rightarrow Q$ is equivalent to $\neg Q \Rightarrow \neg P$ the previous expression also models

$$f(x) > b \Longrightarrow \delta = 0$$

MILP: modeling tricks

Let $\epsilon > 0$ be a small number, and m and M two constants such that $m \leq f(x) - b \leq M$ for any feasible value of x. And let $\delta \in \{0, 1\}$ be a binary variable.

Conditional constraints 5 (type $\leq$)

$$f(x) \leq b \Longrightarrow \delta = 1$$

is equivalent to

$$\delta = 0 \Longrightarrow f(x) > b$$

which can be tranformed into

$$\delta = 0 \Longrightarrow f(x) \geq b + \epsilon.$$

The previous expressions can be both modeled as

$$f(x) \geq b + \epsilon + (m - \epsilon)\delta$$

MILP: modeling tricks

Let $\epsilon > 0$ be a small number, and m and M two constants such that $m \leq f(x) - b \leq M$ for any feasible value of x. And let $\delta \in \{0, 1\}$ be a binary variable.

MILP: modeling tricks

Let $\epsilon > 0$ be a small number, and m and M two constants such that $m \leq f(x) - b \leq M$ for any feasible value of x. And let $\delta \in \{0, 1\}$ be a binary variable.

Conditional constraints 6 (type $\geq$)

$$\delta = 1 \Longrightarrow f(x) \geq b$$

can be modeled as

$$f(x) \geq b + m(1 - \delta).$$

Since $P \Rightarrow Q$ is equivalent to $\neg Q \Rightarrow \neg P$ the previous expression also models

$$f(x) < b \Longrightarrow \delta = 0$$

MILP: modeling tricks

Let $\epsilon > 0$ be a small number, and m and M two constants such that $m \leq f(x) - b \leq M$ for any feasible value of x. And let $\delta \in \{0, 1\}$ be a binary variable.

Conditional constraints 7 (type $\geq$)

$$f(x) \geq b \Longrightarrow \delta = 1$$

is equivalent to

$$\delta = 0 \Longrightarrow f(x) < b$$

which can be transformed into

$$\delta = 0 \Longrightarrow f(x) \leq b - \epsilon.$$

The previous expressions can be both modeled as

$$f(x) \leq b - \epsilon + (M + \epsilon)\delta$$

MILP: modeling tricks

Let $\epsilon > 0$ be a small number, and m and M two constants such that $m \leq f(x) - b \leq M$ for any feasible value of x. And let $\delta \in \{0, 1\}$ be a binary variable.

MILP: modeling tricks

Let $\epsilon > 0$ be a small number, and m and M two constants such that $m \leq f(x) - b \leq M$ for any feasible value of x. And let $\delta \in \{0, 1\}$ be a binary variable.

Conditional constraints 8 (type $=$)

$$\delta = 1 \Longrightarrow f(x) = b \text{ is equivalent to } \delta = 1 \Longrightarrow \begin{cases} f(x) \leq b \\ f(x) \geq b \end{cases}$$

Hence, it can be modeled by the constraints

$$\begin{aligned} f(x) &\leq b + M(1 - \delta) \\ f(x) &\geq b + m(1 - \delta) \end{aligned}$$

Since $P \Rightarrow Q$ is equivalent to $\neg Q \Rightarrow \neg P$ the previous expression also models

$$f(x) \neq b \Longrightarrow \delta = 0$$

MILP: modeling tricks

Let $\epsilon > 0$ be a small number, and m and M two constants such that $m \leq f(x) - b \leq M$ for any feasible value of x. And let $\delta \in \{0, 1\}$ be a binary variable.

Conditional constraints 9 (type $=$)

$f(x) = b \Longrightarrow \delta = 1$ is equivalent to

$$\begin{gathered} f(x) \leq b \Longrightarrow \delta_1 = 1 \\ f(x) \geq b \Longrightarrow \delta_2 = 1 \\ \left.\begin{array}{l} \delta_1 = 1 \\ \delta_2 = 1 \end{array}\right\} \Longrightarrow \delta = 1 \\ \delta_1, \delta_2 \in \{0, 1\} \end{gathered}$$

which can be modeled as

$$\begin{gathered} f(x) \geq b + \epsilon + (m - \epsilon)\delta_1 \\ f(x) \leq b - \epsilon + (M + \epsilon)\delta_2 \\ \delta_1 + \delta_2 - \delta \leq 1 \\ \delta_1, \delta_2 \in \{0, 1\} \end{gathered}$$

MILP: modeling tricks

Let $\epsilon > 0$ be a small number, and m and M two constants such that $m \leq f(x) - b \leq M$ for any feasible value of x. And let $\delta \in \{0, 1\}$ be a binary variable.

Conditional constraints 9 (type $=$)

Since $f(x) = b \Longrightarrow \delta = 1$ is equivalent to $\delta = 0 \Longrightarrow f(x) \neq b$

this last conditional constraint can also be modeled as

$$f(x) \geq b + \epsilon + (m - \epsilon)\delta_1$$
$$f(x) \leq b - \epsilon + (M + \epsilon)\delta_2$$
$$\delta_1 + \delta_2 - \delta \leq 1$$
$$\delta_1, \delta_2 \in \{0, 1\}$$

MILP: modeling tricks

Let $\epsilon > 0$ be a small number, and m and M two constants such that $m \leq f(x) - b \leq M$ for any feasible value of x. And let $\delta \in \{0, 1\}$ be a binary variable.

Conditional constraints 10: double implications

Double implications can be transformed into two unidirectional implications. For instance

$$\delta = 1 \iff f(x) \leq b$$

is equivalent to

$$\begin{cases} \delta = 1 & \implies & f(x) \leq b \\ f(x) \leq b & \implies & \delta = 1 \end{cases}$$

Hence, it can be modeled as

$$f(x) \leq b + M(1 - \delta)$$
$$f(x) \geq b + \epsilon + (m - \epsilon)\delta$$

MILP: modeling tricks

Let $\epsilon > 0$ be a small number, and m and M two constants such that $m \leq f(x) - b \leq M$ for any feasible value of x. And let $\delta \in \{0,1\}$ be a binary variable.

Conditional constraints 10: double implications

$$\delta = 1 \Longleftrightarrow f(x) \geq b$$

can be modeled as

$$f(x) \geq b + m(1-\delta)$$
$$f(x) \leq b - \epsilon + (M+\epsilon)\delta$$

MILP: modeling tricks

Let $\epsilon > 0$ be a small number, and m and M two constants such that $m \leq f(x) - b \leq M$ for any feasible value of x. And let $\delta \in \{0, 1\}$ be a binary variable.

Conditional constraints 10: double implications

$$\delta = 1 \Longleftrightarrow f(x) = b$$

can be modeled as

$$\begin{gathered} f(x) \leq b + M(1 - \delta) \\ f(x) \geq b + m(1 - \delta) \\ f(x) \geq b + \epsilon + (m - \epsilon)\delta_1 \\ f(x) \leq b - \epsilon + (M + \epsilon)\delta_2 \\ \delta_1 + \delta_2 - \delta \leq 1 \\ \delta_1, \delta_2 \in \{0, 1\} \end{gathered}$$

MILP: modeling tricks

Equivalences for conditional propositions

The following equivalences can be used before converting them into constraints:

$P \Rightarrow Q$	$\neg P \vee Q$
$P \Rightarrow (Q \wedge R)$	$(P \Rightarrow Q) \wedge (P \Rightarrow R)$
$P \Rightarrow (Q \vee R)$	$(P \Rightarrow Q) \vee (P \Rightarrow R)$
$(P \wedge Q) \Rightarrow R$	$(P \Rightarrow R) \vee (Q \Rightarrow R)$
$(P \vee Q) \Rightarrow R$	$(P \Rightarrow R) \wedge (Q \Rightarrow R)$
$\neg(P \vee Q)$	$(\neg P) \wedge (\neg Q)$
$\neg(P \wedge Q)$	$(\neg P) \vee (\neg Q)$

MILP: modeling tricks

Assume that the indicator variable δ_i is equal to 1 when the constraint C_i holds:

$$\delta_i = \begin{cases} 1 & \text{if } C_i \text{ holds} \\ 0 & \text{otherwise} \end{cases}$$

Simple conditional or composed statements

$C_1 \vee C_2$	$\delta_1 + \delta_2 \geq 1$
$C_1 \wedge C_2$	$\delta_1 + \delta_2 = 2$
$\neg C_1$	$\delta_1 = 0$
$C_1 \implies C_2$	$\delta_1 \leq \delta_2$
$C_1 \iff C_2$	$\delta_1 = \delta_2$

MILP: modeling tricks

Complex conditional or composed statements

Complex conditional or composed statements are decomposed into two implications in order to model them easier.

Example

$$(C_1 \vee C_2) \Longrightarrow (C_3 \vee C_4 \vee C_5)$$

is modeled as

$$(\delta_1 + \delta_2 \geq 1) \Longrightarrow (\delta_3 + \delta_4 + \delta_5 \geq 1)$$

which, in turn, can be transformed into

$$(\delta_1 + \delta_2 \geq 1) \Rightarrow \delta = 1 \Rightarrow (\delta_3 + \delta_4 + \delta_5 \geq 1)$$

or more clearly,

$$\begin{cases} (\delta_1 + \delta_2 \geq 1) \Rightarrow \delta = 1 \\ \delta = 1 \Rightarrow (\delta_3 + \delta_4 + \delta_5 \geq 1) \end{cases}$$

which becomes

$$\begin{cases} \delta_1 + \delta_2 \leq 2\delta \\ \delta \leq \delta_3 + \delta_4 + \delta_5 \end{cases}$$

MILP: modeling tricks

Example

$$(x \leq b) \wedge (x \geq 1) \implies (y = z + 1)$$

is first transformed into

$$(x \leq b) \wedge (x \geq 1) \implies \delta = 1 \implies (y = z + 1)$$

and this in turn is written as

$$\begin{array}{lll}
(x \leq b) \Rightarrow \delta_1 = 1 & & x \geq b + \epsilon + (m_1 - \epsilon)\delta_1 \\
(x \geq 1) \Rightarrow \delta_2 = 1 & & x \leq 1 - \epsilon + (M_1 + \epsilon)\delta_2 \\
(\delta_1 = 1) \wedge (\delta_1 = 1) \Rightarrow \delta = 1 & \text{which becomes} & \delta_1 + \delta_2 - \delta \leq 1 \\
(\delta = 1) \Rightarrow (y \geq z + 1) & & y - z \geq 1 + m_2(1 - \delta) \\
(\delta = 1) \Rightarrow (y \leq z + 1) & & y - z \leq 1 + M_2(1 - \delta)
\end{array}$$

where $\epsilon > 0$ is a small number and $m_1 \leq x - b$, $M_1 \geq x - 1$, $m_2 \leq y - z - 1 \leq M_2$.

MILP: modeling tricks

More tricks have been designed to:

- Define nonconvex polygonal regions throught a set of constraints.
- Work with Special Ordered Sets of type 1 (SOS1), where in a set of variables only one of them can have a value different from 0, and SOS2, where in a set a variables at most two of them can be different from 0 and they must be consecutive variables.

MILP: modeling tricks

More tricks have been designed to:

- Define nonconvex polygonal regions throught a set of constraints.
- Work with Special Ordered Sets of type 1 (SOS1), where in a set of variables only one of them can have a value different from 0, and SOS2, where in a set a variables at most two of them can be different from 0 and they must be consecutive variables.

Sometimes the same problem can be modeled in different ways. A formulation ‘A’ is said to be better (stronger) than another formulation ‘B’ if the feasible set of ‘A’ is included in the feasible set of ‘B’.

MILP: modeling tricks

More tricks have been designed to:

- Define nonconvex polygonal regions throught a set of constraints.
- Work with Special Ordered Sets of type 1 (SOS1), where in a set of variables only one of them can have a value different from 0, and SOS2, where in a set a variables at most two of them can be different from 0 and they must be consecutive variables.

Sometimes the same problem can be modeled in different ways. A formulation ‘A’ is said to be better (stronger) than another formulation ‘B’ if the feasible set of ‘A’ is included in the feasible set of ‘B’. In this way, the solution of the LP relaxation of ‘A’ will have a better (or equal) objective value than the solution of the LP relaxation of ‘B’, so its gap (the difference between the solution of the linear relaxation and the integer solution) will be smaller.

MILP: modeling tricks

More tricks have been designed to:

- Define nonconvex polygonal regions throught a set of constraints.
- Work with Special Ordered Sets of type 1 (SOS1), where in a set of variables only one of them can have a value different from 0, and SOS2, where in a set a variables at most two of them can be different from 0 and they must be consecutive variables.

Sometimes the same problem can be modeled in different ways. A formulation 'A' is said to be better (stronger) than another formulation 'B' if the feasible set of 'A' is included in the feasible set of 'B'. In this way, the solution of the LP relaxation of 'A' will have a better (or equal) objective value than the solution of the LP relaxation of 'B', so its gap (the difference between the solution of the linear relaxation and the integer solution) will be smaller.

Interestingly, in MILP sometimes it is better a formulation with a bigger number of variables and constraints!