# ISE 426
# Optimization models and applications

Lecture 16 — October 27, 2015

- More IP
- $a^\top x \leq b \quad \Rightarrow \quad c^\top x \leq d$
- Extreme points and relaxations

# Fun with logic

Consider propositions $a, b, c \ldots$, all in $\{T, F\}$. They can all be modeled with **binary** variables $x_a, x_b, x_c \ldots \in \{0, 1\}$
The negation of $a$ is denoted as $\neg a$ or $\bar{a}$.

- $a \vee b$ (i.e., "$a \vee b$ is true") becomes $x_a + x_b \geq 1$
- $a \wedge b$ becomes $x_a = 1, x_b = 1$ or $x_a + x_b = 2$ (trivial!)
- $\neg a$ becomes $x_a = 0$

  Examples:
- $\neg(a \vee b)$ becomes $\neg(x_a + x_b \geq 1)$ or $x_a + x_b = 0$ or $x_a = x_b = 0$
- $a \vee \neg b$ becomes $x_a + (1 - x_b) \geq 1$ or $x_a \geq x_b$
- $\neg(a \wedge b)$ becomes $x_a + x_b \leq 1$
- $a \wedge \neg b$ becomes $x_a = 1, x_b = 0$ or $x_a + (1 - x_b) = 2$

# More fun with logic

- $a \Rightarrow b$ becomes $x_a \leq x_b$
- $a \Leftrightarrow b$ becomes $x_a = x_b \quad (x_a \leq x_b, \quad x_b \leq x_a)$

- $c \Rightarrow a \vee b$ becomes $x_c \leq x_a + x_b$
- $c \Leftarrow a \vee b$ becomes $x_c \geq x_a, x_c \geq x_b$
- $c \Leftrightarrow a \vee b$ becomes $x_c \geq x_a, x_c \geq x_b, x_c \leq x_a + x_b$

- $c \Rightarrow a \wedge b$ becomes $x_c \leq x_a, x_c \leq x_b$
- $c \Leftarrow a \wedge b$ becomes $x_c \geq x_a + x_b - 1$
- $c \Leftrightarrow a \wedge b$ becomes $x_c \leq x_a, x_c \leq x_b, x_c \geq x_a + x_b - 1$

# Examples

"$(a \wedge \neg b \wedge \neg c) \vee (b \wedge \neg c) \vee (\neg a \wedge c)$ is true" becomes

$$
\begin{aligned}
& x_1 + x_2 + x_3 \geq 1 \\
& x_1 \geq x_a + (1 - x_b) + (1 - x_c) - 2 \\
& x_1 \leq x_a \\
& x_1 \leq 1 - x_b \\
& x_1 \leq 1 - x_c \\
& x_2 \geq x_b + (1 - x_c) - 1 \\
& x_2 \leq x_b \\
& x_2 \leq 1 - x_c \\
& x_3 \geq (1 - x_a) + x_c - 1 \\
& x_3 \leq 1 - x_a \\
& x_3 \leq x_c
\end{aligned}
$$

# Examples

"$(a \wedge b \wedge \neg c) \vee (\neg a \wedge b \wedge \neg c) \Rightarrow (\neg a \wedge d \wedge e)$ is true" becomes

$$
\begin{aligned}
x_1 &\leq x_3 \\
x_2 &\leq x_3 \\
x_1 &\geq x_a + x_b + (1 - x_c) - 2 \\
x_1 &\leq x_a \\
x_1 &\leq x_b \\
x_1 &\leq 1 - x_c \\
x_2 &\geq (1 - x_a) + x_b + (1 - x_c) - 2 \\
x_2 &\leq 1 - x_a \\
x_2 &\leq x_b \\
x_2 &\leq 1 - x_c \\
x_3 &\geq (1 - x_a) + x_d + x_e - 2 \\
x_3 &\leq 1 - x_a \\
x_3 &\leq x_d \\
x_3 &\leq x_e
\end{aligned}
$$

# Switching constraints on/off

Suppose constraint $a_1x_1 + a_2x_2 + \ldots + a_nx_n \leq b$, or $a^\top x \leq b$ for short, depends on the value a binary variable $y$.

- for instance, it is implied by $y = 1$
- or it implies $y = 1$ when satisfied

How do we model this in MILP? The constraint can still be linear, but it will depend on $y$.

# Switching constraints on/off

Consider $y = 1 \Rightarrow a^\top x \leq b$.

- constraint $a^\top x \leq b$ is mandatory if $y = 1$.
- it is not if $y = 0$, that is, if $y = 0$ then $a^\top x \leq +\infty$

We need to **unify** these two constraints. Easy! We only need to change the right-hand side (rhs) of $a^\top x \leq b$:

$$a^\top x \leq b + M(1 - y) \quad \Leftrightarrow \quad \begin{cases} a^\top x \leq b + M \approx +\infty & \text{if } y = 0 \\ a^\top x \leq b & \text{if } y = 1 \end{cases}$$

$y = 1 \Rightarrow a^\top x \geq b$ translates to $a^\top x \geq b - M'(1 - y)$.

What about $y = 1 \Rightarrow a^\top x = b$? It's equivalent to $y = 1 \Rightarrow (a^\top x \geq b, a^\top x \leq b)$:

$$\begin{array}{l} a^\top x \leq b + M(1 - y) \\ a^\top x \geq b - M'(1 - y) \end{array}$$

$M$ and $M'$ are positive and, in general, different.

# Switching constraints on/off

Now the equivalence: $y = 1 \Leftrightarrow a^\top x \leq b$. It means

$$\begin{array}{ll} y = 1 \Rightarrow a^\top x \leq b & \quad (\textbf{and}) \\ a^\top x \leq b \Rightarrow y = 1 & \end{array}$$

or equivalently

$$\begin{array}{ll} y = 1 \Rightarrow a^\top x \leq b & \quad (\textbf{and}) \\ y = 0 \Rightarrow a^\top x > b & \end{array}$$

We can model the first as $a^\top x \leq b + M(1 - y)$; the second is:

$$a^\top x > b - M'y$$

In general, the "$>$" or "$<$" are not accepted, as it depends on the precision of solver/modeler/computer (e.g. $10^{-15}$). Same for "$\neq$". Use small numbers:

$$a^\top x \geq b + \epsilon - M'y$$

# What are good values of $M$ and $M'$?

$$a^\top x \leq b + M(1-y) \qquad a^\top x \geq b - M'(1-y)$$

- Suppose each variable $x_i$ has lower/upper bounds $l_i \leq x_i \leq u_i$. For short, let's use $l \leq x \leq u$ or $x \in [l, u]$
- Otherwise, use huge bounds $-10^{20} \leq x_i \leq 10^{20}$ (bad!)
- $a^\top x \leq +\infty$ means "$a^\top x$ is at most the maximum value it can take with $x \in [l, u]$" (redundant: $a^\top x$ can be anything)
- $a^\top x \geq -\infty$ means "$a^\top x$ is at least the minimum value it can take with $x \in [l, u]$" (likewise)
- If $a \geq 0$, i.e. if all $a_i \geq 0$, then $M = au$ and $M' = al$

$\Rightarrow$ $M$ is $\sum_{i:a_i>0} a_i u_i + \sum_{i:a_i<0} a_i l_i - b$

$\Rightarrow$ $M'$ is $b - (\sum_{i:a_i>0} a_i l_i + \sum_{i:a_i<0} a_i u_i)$

# Example 1

$$
\begin{array}{rcl}
y = 1 \Rightarrow 3x_1 + 5x_2 - 2x_3 \leq 6 \\
-7 \leq & x_1 & \leq 4 \\
0 \leq & x_2 & \leq 12 \\
-11 \leq & x_3 & \leq -1 \\
& y \in \{0, 1\} &
\end{array}
$$

becomes $3x_1 + 5x_2 - 2x_3 \leq 6 + M(1 - y)$, with

$$M = 3 \cdot 4 + 5 \cdot 12 - 2 \cdot (-11) - 6 = 94 - 6 = 88$$

which means

$$3x_1 + 5x_2 - 2x_3 \leq \begin{cases} 6 & \text{if } y = 1 \\ 6 + 88 = 94 & \text{if } y = 0 \end{cases}$$

**94** maximizes $3x_1 + 5x_2 - 2x_3$ for $x_1, x_2, x_3$ in their bounds.

It is a redundant rhs when $y = 0$ (exactly what we need).

# Example 2

$$\begin{aligned} y = 1 \Rightarrow 3x_1 + 5x_2 - 2x_3 &\geq 6 \\ -7 \leq \quad x_1 \quad &\leq 4 \\ 0 \leq \quad x_2 \quad &\leq 12 \\ -11 \leq \quad x_3 \quad &\leq -1 \\ y &\in \{0, 1\} \end{aligned}$$

becomes $3x_1 + 5x_2 - 2x_3 \geq 6 - M'(1 - y)$, with

$$M' = 6 - (3 \cdot -7 + 5 \cdot 0 - 2 \cdot (-1)) = 6 - (-19) = 25$$

which means

$$3x_1 + 5x_2 - 2x_3 \geq \begin{cases} 6 & \text{if } y = 1 \\ 6 - 25 = -19 & \text{if } y = 0 \end{cases}$$

**-19** minimizes $3x_1 + 5x_2 - 2x_3$ for $x_1, x_2, x_3$ in their bounds

It is a redundant rhs when $y = 0$ (exactly what we need).

# Example 3

$$
\begin{aligned}
y = 1 \Leftrightarrow 3x_1 + 5x_2 - 2x_3 &\leq 6 \\
-7 \leq \quad x_1 \quad &\leq 4 \\
0 \leq \quad x_2 \quad &\leq 12 \\
-11 \leq \quad x_3 \quad &\leq -1 \\
y &\in \{0, 1\}
\end{aligned}
$$

becomes

$$
\begin{aligned}
3x_1 + 5x_2 - 2x_3 &\leq 6 + M(1 - y) \\
3x_1 + 5x_2 - 2x_3 &\geq 6 + \epsilon - M'y
\end{aligned}
$$

$$
\begin{aligned}
&M = 3 \cdot 4 + 5 \cdot 12 - 2 \cdot (-11) - 6 = 94 - 6 = 88 \\
&M' = 6 + \epsilon - (3 \cdot -7 + 5 \cdot 0 - 2 \cdot (-1)) = 6 - (-19) = 25 + \epsilon
\end{aligned}
$$

which means

$$
3x_1 + 5x_2 - 2x_3 \begin{cases} \leq 6 & \text{if } y = 1 \\ \geq 6 + \epsilon & \text{if } y = 0 \end{cases}
$$

# Implications among constraints

$$a^\top x \leq b \Rightarrow c^\top x \leq d$$

is equivalent to

$$a^\top x \leq b \Rightarrow y = 1; \qquad y = 1 \Rightarrow c^\top x \leq d.$$

$a^\top x \leq b \Leftarrow c^\top x \leq d$ is equivalent to $c^\top x \leq d \Rightarrow a^\top x \leq b$.

$a^\top x \leq b \Leftrightarrow c^\top x \leq d$ is equivalent to $\begin{cases} c^\top x \leq d \Rightarrow a^\top x \leq b \\ a^\top x \leq b \Rightarrow c^\top x \leq d \end{cases}$

Now you can also model things such as

$$(a^\top x \leq b) \vee \neg(c^\top x \geq d) \Rightarrow (d^\top x \geq e) \wedge ((f^\top x \leq g) \vee \neg(h^\top x \geq p))$$

# Relaxations and efficiency

Integer programming problems:

$$\begin{array}{llllll}
(IP) & \min & c_1x_1 & +c_2x_2 & \dots & +c_nx_n & & \\
 & & a_{11}x_1 & +a_{12}x_2 & \dots & +a_{1n}x_n & \leq b_1 & \\
 & & a_{21}x_1 & +a_{22}x_2 & \dots & +a_{2n}x_n & \leq b_2 & \\
 & & \vdots & & & & & \\
 & & a_{m1}x_1 & +a_{m2}x_2 & \dots & +a_{mn}x_n & \leq b_m & \\
 & & & & & x_i \in \mathbb{Z} & \forall i \in J \subseteq \{1, 2 \dots, n\} &
\end{array}$$

or, for short,

$$\begin{array}{lll}
(IP) & \min & c^\top x \\
 & & Ax \leq b \\
 & & x_i \in \mathbb{Z} \quad \forall i \in J \subseteq N,
\end{array}$$

can be solved using their LP relaxation:

$$\begin{array}{lll}
(LP) & \min & c^\top x \\
 & & Ax \leq b.
\end{array}$$

A global optimum $z$ of $(LP)$ is a **lower bound** for $(IP)$.

# Relaxations and efficiency

If an optimal solution $x^\star$ of $(LP)$ is feasible for $(IP)$, i.e., for all $i \in J$ we have $x_i^\star \in \mathbb{Z}$, we're done!

This is **not** the case, usually…

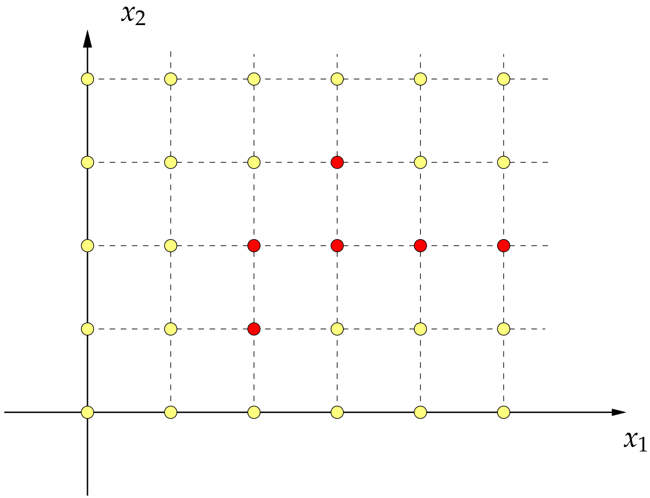
What do we know about the optimal solutions of $(LP)$? They are all *vertices* of the polyhedron

$$\{x \in \mathbb{R}^n : A^\top x \leq b\}$$

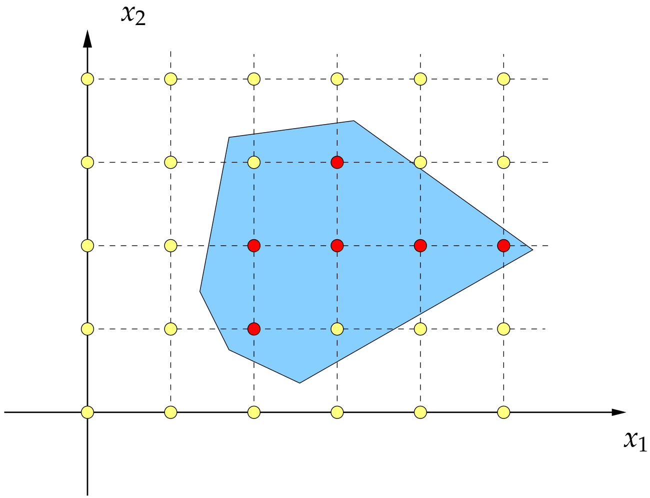
Therefore, it would be just great if all vertices of $(LP)$ were feasible for $(IP)$. Solving IPs would amount to solving LPs, which are a lot easier.

A good model may not achieve just that, but it can help a lot.

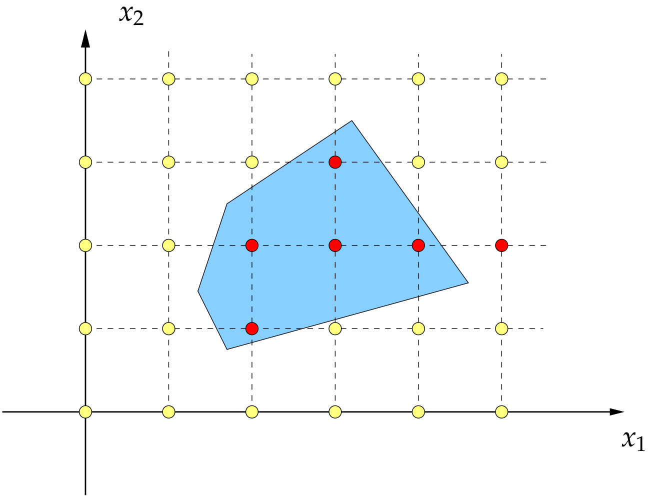
# Relaxations, the geometrical standpoint



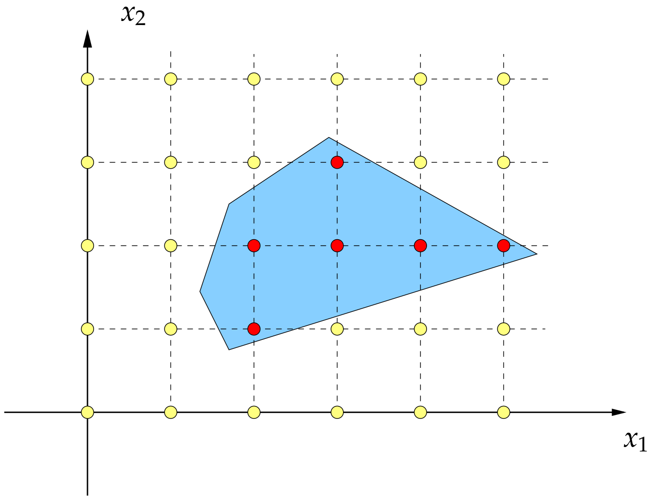
# Relaxations, the geometrical standpoint



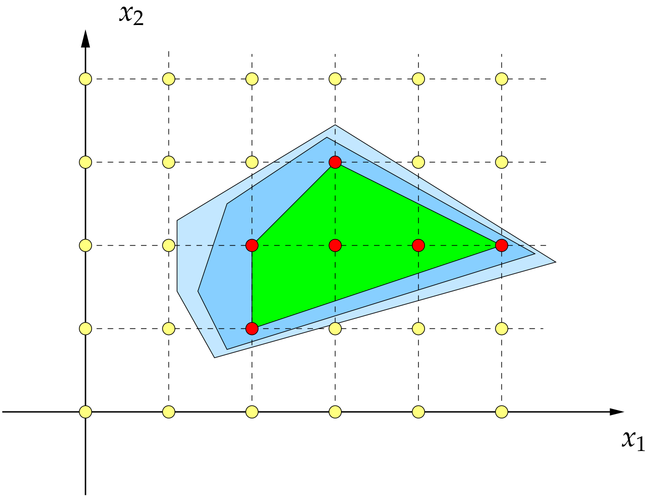
# Relaxations, the geometrical standpoint



# Relaxations, the geometrical standpoint



# Relaxations, the geometrical standpoint



# Relaxations: the clique inequality

Two models for one problem have the same feasible set and global optima, but may be solved differently:

$$\left.\begin{array}{rl} P_1 : \min & -7x_1 - 8x_2 - 9x_3 \\ \text{s.t.} & x_1 + x_2 \le 1 \\ & x_1 + x_3 \le 1 \\ & x_2 + x_3 \le 1 \\ & x_1, x_2, x_3 \in \{0,1\} \end{array}\right\} \equiv \left\{\begin{array}{rl} P_2 : \min & -7x_1 - 8x_2 - 9x_3 \\ \text{s.t.} & x_1 + x_2 + x_3 \le 1 \\ & x_1, x_2, x_3 \in \{0,1\} \end{array}\right.$$

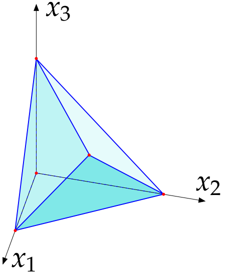
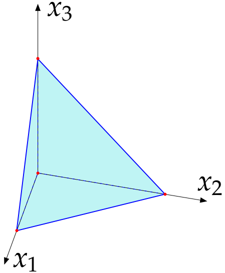
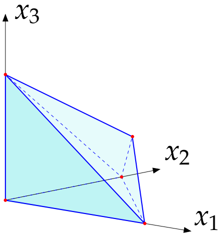
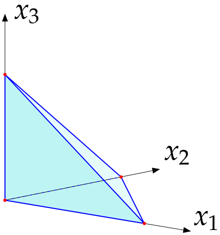
Consider relaxations $R_1, R_2$ of $P_1, P_2$ with $x_i \in [0,1]$.

$R_1$: optimal soln. $(\frac{1}{2}, \frac{1}{2}, \frac{1}{2})$, obj.f. $-12$: lower bound for $P_1, P_2$

$R_2$: optimal solution $(0,0,1)$, obj.f. $-9$: lower bound for $P_2$ and $P_1$, and **feasible** for $P_2$ and $P_1$!

$\Rightarrow$ optimum of $P_1, P_2$: $-9$, and $P_2$ is a better model than $P_1$

# Relaxations: the clique inequality

| $R_1$ | $R_2$ |
|---|---|
|  |  |
|  |  |

# Good vs. bad models: Uncapacitated Facility Location

A set $J$ of retailers has to be served by a set $S$ of plants, yet to be built. We don't know where the plants will be, but there is a set $I$ of potential sites, and there is

- a cost $f_i$ for building plant $i \in I$
- a (transportation) cost $c_{ij}$ from plant $i$ to retailer $j$

Each retailer will be served by exactly one plant (why?).
Choose a subset $S$ of $I$ such that the total cost is minimized.

Variables:

- $x_i, i \in I$: 1 if plant $i$ is built, 0 otherwise
- $y_{ij}$ assigns retailer $j$ to plant $i$: 1 if $i$ serves retailer $j$, 0 otherwise

# Good vs. bad models: Uncapacitated Facility Location

Objective function:

$$\sum_{i \in I} f_i x_i + \sum_{i \in I} \sum_{j \in J} c_{ij} y_{ij}$$

Constraints:

- for each customer, one facility: $\sum_{i \in I} y_{ij} = 1$
- customers go to mall $i$ if it's there:

$$\sum_{j \in J} y_{ij} \leq |J| x_i \quad \forall i \in I$$

**or**

$$y_{ij} \leq x_i \quad \forall i \in I, j \in J$$

# Good vs. bad models: Uncapacitated Facility Location

$(A)$

$$\begin{aligned}\min \quad & \sum_{i \in I} f_i x_i + \sum_{i \in I} \sum_{j \in J} c_{ij} y_{ij} \\ & \sum_{i \in I} y_{ij} = 1 \quad \forall i \in I \\ & \sum_{j \in J} y_{ij} \leq |J| x_i \quad \forall i \in I \\ & x_i, y_{ij} \in \{0, 1\}\end{aligned}$$

$(B)$

$$\begin{aligned}\min \quad & \sum_{i \in I} f_i x_i + \sum_{i \in I} \sum_{j \in J} c_{ij} y_{ij} \\ & \sum_{i \in I} y_{ij} = 1 \quad \forall i \in I \\ & y_{ij} \leq x_i \quad \forall i \in I, j \in J \\ & x_i, y_{ij} \in \{0, 1\}\end{aligned}$$

(A) and (B) are **equivalent**. However, for $|I| = |J| = 40$,

(A) takes 14 hours[1]

(B) takes 2 seconds

[1] On AMPL with an old version of CPLEX.