MA427 Lecture 5
Integer Programming
Branch-and-bound and good formulations

Giacomo Zambelli

Department of Mathematics

11 February, 2018

Today's lecture

- Branch-and-bound
- Good formulations
- Knapsack cover inequalities
- Ideal formulations

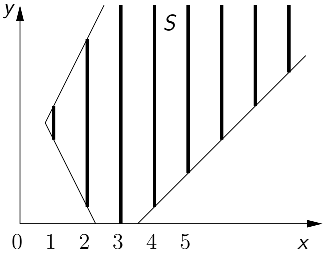
Mixed Integer Linear Programming Problems (MILP)

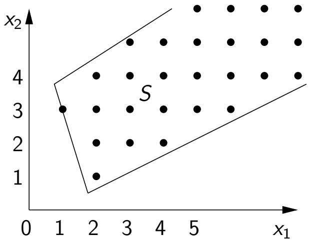
$$\begin{aligned} z_I = \max c^\top x \\ Ax \leq b \\ x \geq 0 \\ x_i \in \mathbb{Z}, \qquad i \in I. \end{aligned}$$

where $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$ and $I \subseteq \{1, \dots, n\}$.

- integer variables:x_i, $i \in I$.
- continuous variables: x_i, $i \notin I$.
- Pure Integer Linear Programming Problem (ILP): if all variables are integer.
- Binary linear programming: if all variables are binary (i.e. $x_i \in \{0, 1\}$.
- Feasible region: $X = \{x : Ax \leq b, x \geq 0, x_i \in \mathbb{Z} \text{ for } i \in I\}$.

Mixed Integer Linear Programming Problems





Mixed Integer Linear Programming Problems

$$
\begin{aligned}
z_I &= \max c^\top x \\
Ax &\le b \\
x &\ge 0 \\
x_i &\in \mathbb{Z}, \qquad i \in I.
\end{aligned}
$$

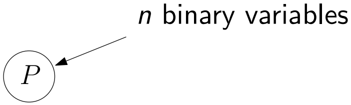
Linear relaxation of the problem

$$
\begin{aligned}
z_L &= \max c^\top x \\
Ax &\le b \\
x &\ge 0
\end{aligned}
\tag{1}
$$

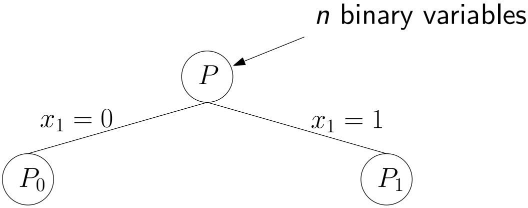
Note: $z_I \le z_L$.

Observation: if the optimal solution of the linear relaxation x^L satisfies the integrality requirements, then x_L is optimal also for the (MILP) problem and $z_I = z_L$

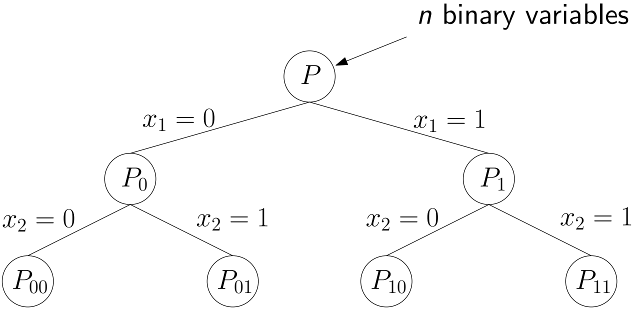
Enumeration by branching



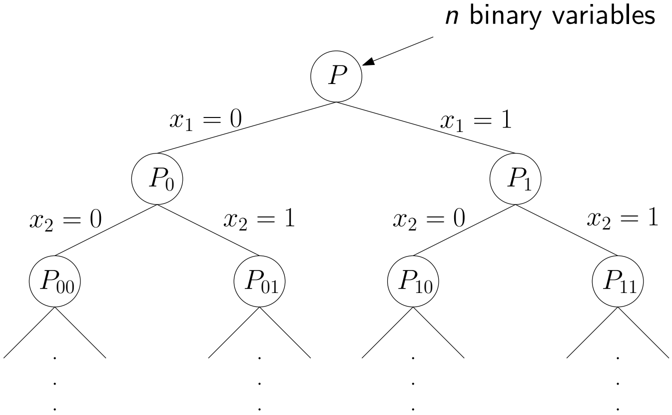
Enumeration by branching



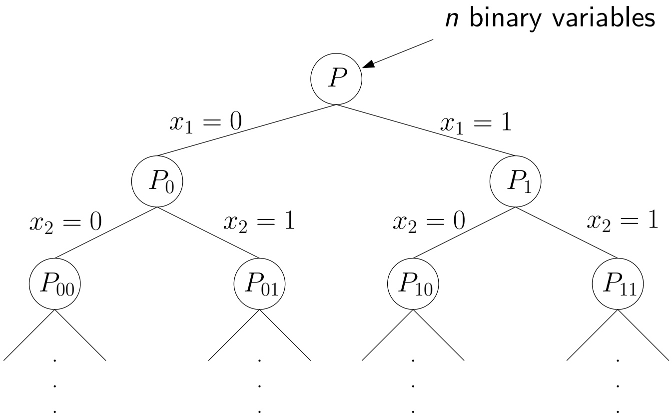
Enumeration by branching



Enumeration by branching



Enumeration by branching



2^n **subproblems at the end**

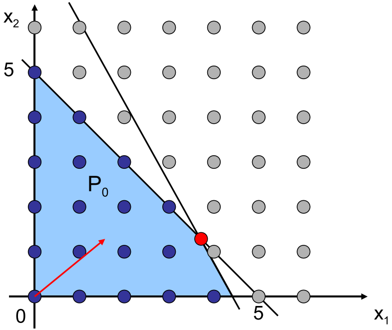
Drawbacks of complete enumeration

- Suppose that, in a IP with n binary variables, we enumerate all 2^n possible solutions.
- If we check one solution per microsecond:

# variables	Time
$n = 15$	$\sim .03$ sec
$n = 20$	~ 1 sec
$n = 30$	~ 20 min
$n = 40$	~ 12 days
$n = 50$	~ 35 years
$n = 60$	$\sim 37,500$ years
$n = 100$	$\sim 40,197$ trillion years

The branch-and-bound method

$$\begin{array}{rcl} z_I^0 = \max \quad 5x_1 + \frac{17}{4}x_2 & & \\ x_1 + x_2 & \leq & 5 \\ 10x_1 + 6x_2 & \leq & 45 \\ x_1, x_2 & \geq & 0 \\ x_1, x_2 & \in & \mathbb{Z} \end{array}$$



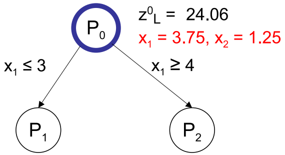
Optimal LP solution is $x_1 = 3.75$, $x_2 = 1.75$, with value $z_L^0 = 24.06$.

The branch-and-bound method

Branch on variable x_1:

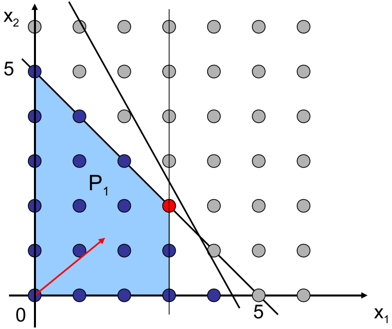
$$
\begin{array}{rrcl}
z_I^1 = \max & 5x_1 + \frac{17}{4}x_2 & & \\
 & x_1 + x_2 & \leq & 5 \\
 & 10x_1 + 6x_2 & \leq & 45 \\
 & x_1 & \leq & 3 \\
 & x_1, x_2 & \geq & 0 \\
 & x_1, x_2 & \in & \mathbb{Z}
\end{array} \qquad (P_1)
$$

$$
\begin{array}{rrcl}
z_I^2 = \max & 5x_1 + \frac{17}{4}x_2 & & \\
 & x_1 + x_2 & \leq & 5 \\
 & 10x_1 + 6x_2 & \leq & 45 \\
 & x_1 & \geq & 4 \\
 & x_1, x_2 & \geq & 0 \\
 & x_1, x_2 & \in & \mathbb{Z}
\end{array} \qquad (P_2)
$$



The branch-and-bound method

$$
\begin{array}{rrcl}
z_I^1 = \max & 5x_1 + \frac{17}{4}x_2 & & \\
& x_1 + x_2 & \leq & 5 \\
& 10x_1 + 6x_2 & \leq & 45 \\
& x_1 & \leq & 3 \\
& x_1, x_2 & \geq & 0 \\
& x_1, x_2 & \in & \mathbb{Z}
\end{array}
$$



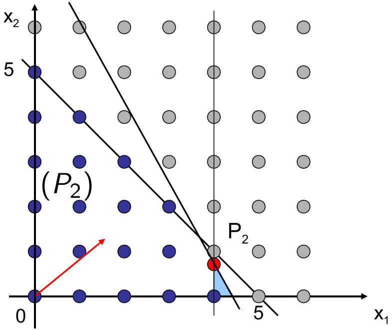
Optimal LP solution $x_1 = 3$, $x_2 = 2$, with value $z_L^1 = 23.5$

Integer solution! $(3, 2)$ is the incumbent solution. $LB := 23.5$.

Prune node by optimality

The branch-and-bound method

$$
\begin{array}{rcl}
z_I^1 = \max \quad 5x_1 + \frac{17}{4}x_2 & & \\
x_1 + x_2 & \leq & 5 \\
10x_1 + 6x_2 & \leq & 45 \\
x_1 & \geq & 4 \\
x_1, x_2 & \geq & 0 \\
x_1, x_2 & \in & \mathbb{Z}
\end{array}
$$



Optimal LP solution $x_1 = 4$, $x_2 = 0.83$, with value $z_L^2 = 23.54$.

The branch-and-bound method

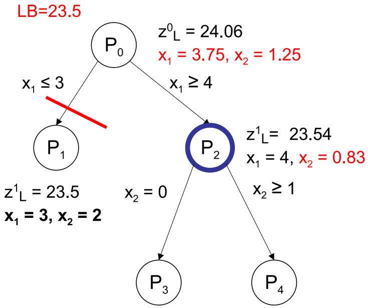
Branch on variable x_2:

$$
z_I^2 = \max \quad 5x_1 + \tfrac{17}{4}x_2 \qquad (P_3)
$$

$$
\begin{array}{rcl}
x_1 + x_2 & \leq & 5 \\
10x_1 + 6x_2 & \leq & 45 \\
x_1 & \geq & 4 \\
x_2 & \leq & 0 \\
x_1, x_2 & \geq & 0 \\
x_1, x_2 & \in & \mathbb{Z}
\end{array}
$$

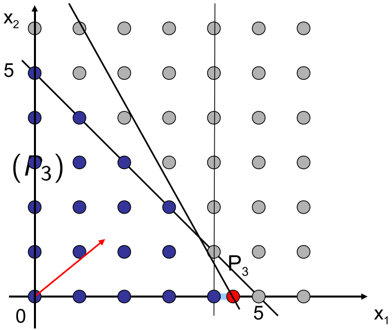
$$
z_I^2 = \max \quad 5x_1 + \tfrac{17}{4}x_2 \qquad (P_4)
$$

$$
\begin{array}{rcl}
x_1 + x_2 & \leq & 5 \\
10x_1 + 6x_2 & \leq & 45 \\
x_1 & \geq & 4 \\
x_2 & \geq & 1 \\
x_1, x_2 & \geq & 0 \\
x_1, x_2 & \in & \mathbb{Z}
\end{array}
$$



The branch-and-bound method

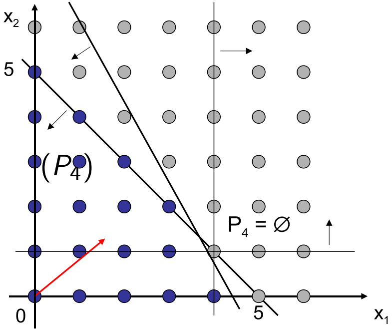
$$
\begin{aligned}
z_I^2 = \max \quad & 5x_1 + \tfrac{17}{4}x_2 & & \\
& x_1 + x_2 & \leq & \quad 5 \\
& 10x_1 + 6x_2 & \leq & \quad 45 \\
& x_1 & \geq & \quad 4 \\
& x_2 & \leq & \quad 0 \\
& x_1, x_2 & \geq & \quad 0 \\
& x_1, x_2 & \in & \quad \mathbb{Z}
\end{aligned}
$$



Optimal LP solution $x_1 = 4.5$, $x_2 = 0$, with value $z_L^3 = 22.5$.
$z_L^3 < LB(22.5 < 23.5) \implies$ prune by bound.

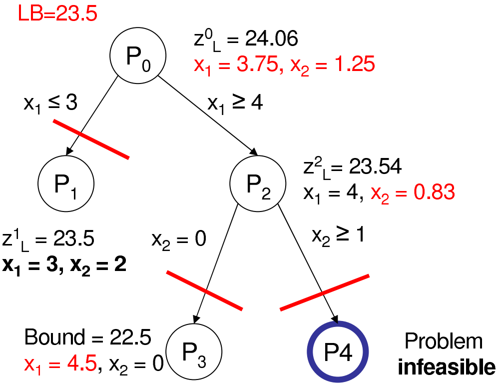
The branch-and-bound method

$$
\begin{array}{rrcl}
z_I^2 = \max & 5x_1 + \frac{17}{4}x_2 & & \\
& x_1 + x_2 & \leq & 5 \\
& 10x_1 + 6x_2 & \leq & 45 \\
& x_1 & \geq & 4 \\
& x_2 & \geq & 1 \\
& x_1, x_2 & \geq & 0 \\
& x_1, x_2 & \in & \mathbb{Z}
\end{array}
$$



LP relaxation infeasible: prune by infeasibility.

The branch-and-bound method



No more active nodes: the optimal solution is $(3, 2)$.

The branch-and-bound method (maximization)

Initialization: $\mathcal{T} := \{(P_0)\}$, $LB := -\infty$, x^* undefined;

1. If $\mathcal{T}$ has no active node, return the incumbent solution x^*, STOP;
2. Else, choose an active node (P) in $\mathcal{T}$.
3. Solve the linear relaxation of (P).
   3.1 **Pruning by infeasibility**: If LP relaxation is infeasible, prune (P);
   Else, let $\bar{x}$ be the optimum to the LP relaxation of (P).
   3.2 **Pruning by bound**: If $c^\top \bar{x} \leq LB$, then prune (P);
   Else
   3.3 **Pruning by optimality**: If $\bar{x}_i$ is integer for all $i \in I$, set $x^* := \bar{x}$ and $LB := c^\top \bar{x}$ and prune node (P).
   3.4 **Branching on variable**: If none of (a), (b), (c) occurs, choose an integer variable x_h such that $\bar{x}_h \notin \mathbb{Z}$ and branch on x_h:
   $$(P') := (P) \cap \{x_h \leq \lfloor \bar{x}_h \rfloor\} \quad , \quad (P'') := (P) \cap \{x_h \geq \lceil \bar{x}_h \rceil\}.$$
   Add (P') and (P'') to $\mathcal{T}$.
4. Return to 1.

Example: how to restart after branching

Suppose we solve the LP relaxation of a MIP in 5 variables with $x_1, x_2, x_3 \in \mathbb{Z}$, to obtain the optimal tableau

$$
\begin{array}{lccccccc}
\max z & & & & & & & \\
z & & +x_2 & & +4x_4 & +x_5 & = & -9 \\
 & & -\frac{1}{5}x_2 & +x_3 & -\frac{1}{5}x_4 & +\frac{1}{5}x_5 & = & \frac{1}{5} \\
 & x_1 & +\frac{1}{5}x_2 & & -\frac{4}{5}x_4 & -\frac{1}{5}x_5 & = & \frac{9}{5} \\
 & & x_1, \ldots, x_5 \geq 0 & & & & &
\end{array}
$$

We branch on x_1. The subproblems are $x_1 \leq 1$ and $x_1 \geq 2$.
How do we restart?

Example: how to restart after branching

$$
\begin{array}{lccccccc}
\max z & & & & & & & \\
z & & +x_2 & & +4x_4 & +x_5 & = & -9 \\
 & & -\frac{1}{5}x_2 & +x_3 & -\frac{1}{5}x_4 & +\frac{1}{5}x_5 & = & \frac{1}{5} \\
 & x_1 & +\frac{1}{5}x_2 & & -\frac{4}{5}x_4 & -\frac{1}{5}x_5 & = & \frac{9}{5} \\
 & & x_1, \ldots, x_5 \geq 0 & & & & &
\end{array}
$$

Subproblem: $x_1 \leq 1$

Add new slack variable $x_6 \geq 0$ and new inequality $x_1 + x_6 = 1$, expressed by non-basic variables as

$$
-\frac{1}{5}x_2 + \frac{4}{5}x_4 + \frac{1}{5}x_5 + x_6 = -\frac{4}{5}.
$$

Example: how to restart after branching

$$
\begin{array}{llllllll}
\max z & & & & & & & \\
z & & +x_2 & & +4x_4 & +x_5 & & = & -9 \\
 & & -\frac{1}{5}x_2 & +x_3 & -\frac{1}{5}x_4 & +\frac{1}{5}x_5 & & = & \frac{1}{5} \\
 & x_1 & +\frac{1}{5}x_2 & & -\frac{4}{5}x_4 & -\frac{1}{5}x_5 & & = & \frac{9}{5} \\
 & & -\frac{1}{5}x_2 & & +\frac{4}{5}x_4 & +\frac{1}{5}x_5 & +x_6 & = & -\frac{4}{5} \\
 & & x_1, \ldots, x_5, x_6 \geq 0 & & & & & &
\end{array}
$$

New dual feasible basis: $\{1, 3, 6\}$.

Good formulations

MILP Formulations

$$
\begin{aligned}
z_I = \max c^\top x & \\
Ax \le b & \\
x \ge 0 & \\
x_i \in \mathbb{Z}, & \qquad i \in I.
\end{aligned}
$$

$$X = \{x \in \mathbb{R}^n \,:\, Ax \le b,\, x \ge 0,\, x_i \in \mathbb{Z} \text{ for all } i \in I\}$$

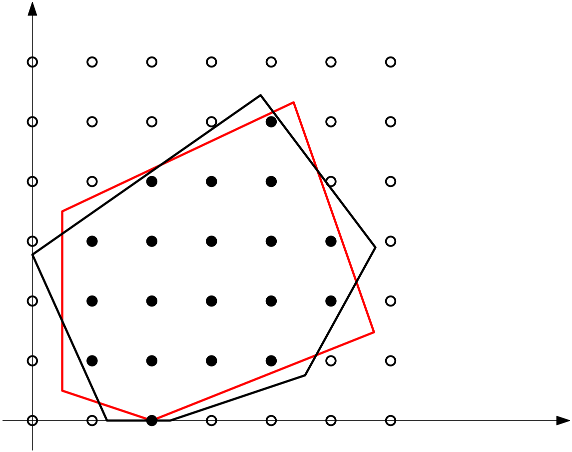
A formulation for the above (MILP) is a system

$$
\begin{aligned}
A'x \le b' \\
x \ge 0
\end{aligned}
$$

such that

$$X = \{x \in \mathbb{R}^n \,:\, A'x \le b',\, x \ge 0,\, x_i \in \mathbb{Z} \text{ for all } i \in I\}.$$

MILP Formulations



MILP Formulations

Two formulations for X:

$$\begin{aligned} Ax &\leq b \\ x &\geq 0 \end{aligned} \quad \text{and} \quad \begin{aligned} A'x &\leq b \\ x &\geq 0 \end{aligned}$$

Definition

The first formulation is better than the second if

$$\{x \,:\, Ax \leq b, \, x \geq 0\} \subseteq \{x \,:\, A'x \leq b', \, x \geq 0\}.$$

MILP Formulations

Two formulations for X:

$$\begin{array}{c} Ax \le b \\ x \ge 0 \end{array} \quad \text{and} \quad \begin{array}{c} A'x \le b \\ x \ge 0 \end{array}$$

Definition

The first formulation is better than the second if

$$\{x : Ax \le b, x \ge 0\} \subseteq \{x : A'x \le b', x \ge 0\}.$$

Note: if $Ax \le b$, $x \ge 0$ is a better formulation than $A'x \le b'$, $x \ge 0$, then

$$z_I \le z_L \le z'_L.$$

Better formulations give tighter lower-bounds.

Knapsack problems

We are given a knapsack of capacity b, and n types of objects of positive weight $a_1, \ldots, a_n$. Each object i has value c_i, $i = 1, \ldots, n$.

Knapsack problems

We are given a knapsack of capacity b, and n types of objects of positive weight $a_1, \ldots, a_n$. Each object i has value c_i, $i = 1, \ldots, n$.

Knapsack problem: *Load the knapsack without exceeding capacity in order to maximize the total value carried.*

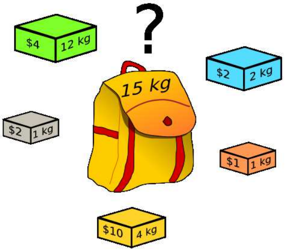
Binary knapsack problem: *Knapsack problem where we have exactly one object of each type.*

Knapsack problems

We are given a knapsack of capacity b, and n types of objects of positive weight $a_1, \ldots, a_n$. Each object i has value c_i, $i = 1, \ldots, n$.

Knapsack problem: *Load the knapsack without exceeding capacity in order to maximize the total value carried.*

Binary knapsack problem: *Knapsack problem where we have exactly one object of each type.*



Knapsack problems

We are given a knapsack of capacity b, and n types of objects of positive weight $a_1, \ldots, a_n$. Each object i has value c_i, $i = 1, \ldots, n$.

Knapsack problem: *Load the knapsack without exceeding capacity in order to maximize the total value carried.*

Binary knapsack problem: *Knapsack problem where we have exactly one object of each type.*

Formulation for the knapsack problem

$$
\begin{array}{ll}
\max & \sum_{i=1}^{n} c_i x_i \\
 & \sum_{i=1}^{n} a_i x_i \leq b \\
 & x \geq 0 \\
 & x \in \mathbb{Z}^n
\end{array}
$$

Knapsack problems

We are given a knapsack of capacity b, and n types of objects of positive weight $a_1, \ldots, a_n$. Each object i has value c_i, $i = 1, \ldots, n$.

Knapsack problem: *Load the knapsack without exceeding capacity in order to maximize the total value carried.*

Binary knapsack problem: *Knapsack problem where we have exactly one object of each type.*

Formulation for the binary knapsack problem

$$
\begin{array}{ll}
\max & \sum_{i=1}^{n} c_i x_i \\
 & \sum_{i=1}^{n} a_i x_i \leq b \\
 & 0 \leq x \leq 1 \\
 & x \in \mathbb{Z}^n
\end{array}
$$

Knapsack problems: minimal cover formulation

Binary knapsack set

$$K := \{x \in \{0,1\}^n : \sum_{i=1}^{n} a_i x_i \leq b\}$$

Covers: $C \subseteq \{1,\ldots,n\}$ is a

- **cover** if $\sum_{i \in C} a_i > b$.
- **minimal cover** if it is a cover and $\sum_{i \in C \setminus \{j\}} a_i \leq b$ for every $j \in C$.

Knapsack problems: minimal cover formulation

Binary knapsack set

$$K := \{x \in \{0,1\}^n : \sum_{i=1}^{n} a_i x_i \leq b\}$$

Covers: $C \subseteq \{1, \ldots, n\}$ is a

- **cover** if $\sum_{i \in C} a_i > b$.
- **minimal cover** if it is a cover and $\sum_{i \in C \setminus \{j\}} a_i \leq b$ for every $j \in C$.

Since objects in a cover exceed capacity, not all of them can be put in the knapsack.

Knapsack problems: minimal cover formulation

Binary knapsack set

$$K := \{x \in \{0,1\}^n : \sum_{i=1}^{n} a_i x_i \leq b\}$$

Covers: $C \subseteq \{1, \ldots, n\}$ is a

- **cover** if $\sum_{i \in C} a_i > b$.
- **minimal cover** if it is a cover and $\sum_{i \in C \setminus \{j\}} a_i \leq b$ for every $j \in C$.

Since objects in a cover exceed capacity, not all of them can be put in the knapsack. $\Rightarrow$ Every point in K satisfy the

cover inequality: $\displaystyle\sum_{i \in C} x_i \leq |C| - 1.$

Knapsack problems: minimal cover formulation

Binary knapsack set

$$K := \{x \in \{0,1\}^n : \sum_{i=1}^{n} a_i x_i \leq b\}$$

Covers: $C \subseteq \{1, \ldots, n\}$ is a

- **cover** if $\sum_{i \in C} a_i > b$.
- **minimal cover** if it is a cover and $\sum_{i \in C \setminus \{j\}} a_i \leq b$ for every $j \in C$.

Since objects in a cover exceed capacity, not all of them can be put in the knapsack.$\Rightarrow$ Every point in K satisfy the

cover inequality: $\sum_{i \in C} x_i \leq |C| - 1.$

Observe that the minimal cover inequalities (i.e. the cover inequalities corresponding to minimal covers) imply all covers inequalities.

Knapsack cover inequalities

$$K^C := \{x \in \{0,1\}^n : \sum_{i \in C} x_i \leq |C| - 1 \text{ for every minimal cover } C \text{ of } S\}.$$

Observation *The sets K and K^C coincide.*

Knapsack cover inequalities

$$K^C := \{x \in \{0,1\}^n : \sum_{i \in C} x_i \leq |C| - 1 \text{ for every minimal cover } C \text{ of } S\}.$$

Observation *The sets K and K^C coincide.*

$$K := \{x \in \{0,1\}^4 :\ 6x_1 + 6x_2 + 6x_3 + 5x_4 \leq 16\}.$$

Minimal cover formulation:

$$\begin{array}{lccccl} K^C := \{x \in \{0,1\}^4 : & x_1 & +x_2 & +x_3 & & \leq 2 \\ & x_1 & +x_2 & & +x_4 & \leq 2 \\ & x_1 & & +x_3 & +x_4 & \leq 2 \\ & & x_2 & +x_3 & +x_4 & \leq 2\}. \end{array}$$

In this instance, the second formulation is strictly better than the first.

Knapsack cover inequalities

$$K^C := \{x \in \{0,1\}^n : \sum_{i \in C} x_i \leq |C| - 1 \text{ for every minimal cover } C \text{ of } S\}.$$

Observation *The sets K and K^C coincide.*

$$K := \{x \in \{0,1\}^4 : \ 6x_1 + 6x_2 + 6x_3 + 5x_4 \leq 16\}.$$

Minimal cover formulation:

$$K^C := \{x \in \{0,1\}^4 : \begin{array}{ccccc} x_1 & +x_2 & +x_3 & & \leq 2 \\ x_1 & +x_2 & & +x_4 & \leq 2 \\ x_1 & & +x_3 & +x_4 & \leq 2 \\ & x_2 & +x_3 & +x_4 & \leq 2\}. \end{array}$$

In this instance, the second formulation is strictly better than the first.

$(\frac{5}{6}, 1, 0, 1)$

Ideal formulations

Given a set $X \subseteq \mathbb{R}^n$, the convex hull of X, denoted by $\mathrm{conv}(X)$, it the (unique) minimal convex set containing X.

Ideal formulations

Given a set $X \subseteq \mathbb{R}^n$, the convex hull of X, denoted by $\operatorname{conv}(X)$, it the (unique) minimal convex set containing X.

- $\operatorname{conv}(X)$ is the intersection of all convex sets containing X.
- In particular, if K is a convex set and $X \subseteq K$, then $\operatorname{conv}(X) \subseteq K$.

Ideal formulations

Given a set $X \subseteq \mathbb{R}^n$, the convex hull of X, denoted by $\mathrm{conv}(X)$, it the (unique) minimal convex set containing X.

NOTE: If X is the set of feasible solutions of a mixed-integer program, and $Ax \leq b$, $x \geq 0$ is a formulation for X, then the polyhedron $P = \{x : Ax \leq b, x \geq 0\}$ satisfies

$$\mathrm{conv}(X) \subseteq P.$$

Hence the best possible formulation of a MILP must contain $\mathrm{conv}(X)$.

Ideal formulations

Theorem (Fundamental theorem of Integer Programming)

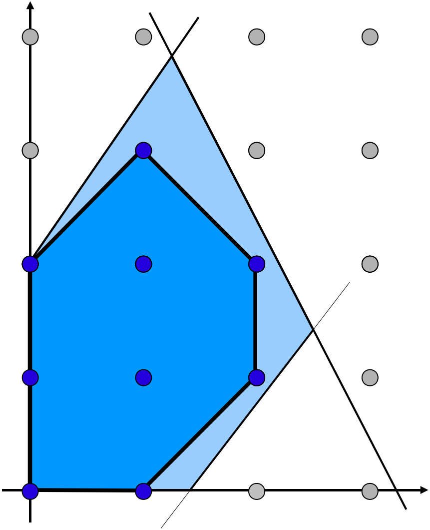
Given $A \in \mathbb{Q}^{m \times n}$, $b \in \mathbb{Q}^m$, let
$X = \{x \in \mathbb{R}^n : Ax \leq b,\, x \geq 0,\, x_i \in \mathbb{Z} \text{ for } i \in I\}$. *Then*
$\mathrm{conv}(X)$ *is a polyhedron.*

$\Longrightarrow$ there exists $\tilde{A} \in \mathbb{Q}^{\tilde{m} \times n}$ and $\tilde{b} \in \mathbb{Q}^{\tilde{m}}$ such that

$$\mathrm{conv}(X) = \{x \in \mathbb{R}^n : \tilde{A}x \leq \tilde{b},\, x \geq 0\}.$$

$\tilde{A}x \leq \tilde{b}$, $x \geq 0$ is the ideal formulation for X.

Ideal formulations



Ideal formulations

Theorem

$X = \{x : Ax \leq b,\, x \geq 0,\, x_i \in \mathbb{Z}$ for $i \in I\}$. For any $c \in \mathbb{R}^n$, let

$$z_I = \max_{x \in X} c^\top x \qquad \text{and} \qquad \tilde{z} = \max_{x \in \operatorname{conv}(X)} c^\top x \quad .$$

If all data is rational, then $z_I = \tilde{z}$, and all vertices of $\operatorname{conv}(X)$ are elements of X.

Ideal formulations

Theorem

$X = \{x : Ax \le b,\ x \ge 0,\ x_i \in \mathbb{Z}$ for $i \in I\}$. For any $c \in \mathbb{R}^n$, let

$$z_I = \max_{x \in X} c^\top x \quad \text{and} \quad \tilde{z} = \max_{x \in \operatorname{conv}(X)} c^\top x .$$

If all data is rational, then $z_I = \tilde{z}$, and all vertices of $\operatorname{conv}(X)$ are elements of X.

- If we know the ideal formulation, any optimal solution to the LP relaxation is also an optimal solution to the MILP.
- In principle solving integer programs can be reduced to Linear Programming.

Ideal formulations

Theorem

$X = \{x : Ax \leq b, x \geq 0, x_i \in \mathbb{Z} \text{ for } i \in I\}$. *For any* $c \in \mathbb{R}^n$, *let*

$$z_I = \max_{x \in X} c^\top x \quad \text{and} \quad \tilde{z} = \max_{x \in \mathrm{conv}(X)} c^\top x \; .$$

If all data is rational, then $z_I = \tilde{z}$, *and all vertices of* $\mathrm{conv}(X)$ *are elements of* X.

- If we know the ideal formulation, any optimal solution to the LP relaxation is also an optimal solution to the MILP.
- In principle solving integer programs can be reduced to Linear Programming. But there is one catch ...

Ideal formulations

Theorem

$X = \{x : Ax \le b,\ x \ge 0,\ x_i \in \mathbb{Z}$ for $i \in I\}$. For any $c \in \mathbb{R}^n$, let

$$z_I = \max_{x \in X} c^\top x \quad \text{and} \quad \tilde{z} = \max_{x \in \mathrm{conv}(X)} c^\top x \quad .$$

If all data is rational, then $z_I = \tilde{z}$, and all vertices of $\mathrm{conv}(X)$ are elements of X.

- If we know the ideal formulation, any optimal solution to the LP relaxation is also an optimal solution to the MILP.
- In principle solving integer programs can be reduced to Linear Programming. But there is one catch ...
  - How do you find the ideal formulation?
  - The ideal formulation can be extremely complicated, and can have exponentially many constraints.