Discrete optimization

Models and algorithms

Michel Bierlaire

Introduction to optimization and operations research

Modeling

Motivation

- Binary variables are convenient to model many situations.
- Action to be taken or not.
- A switch to set to “on”.
- We first investigate some techniques to translate logical rules into a mathematical formulation involving binary variables.

Logical identity

$$x = \begin{cases} 1 & \text{if } P \text{ is true,} \\ 0 & \text{if } P \text{ is false.} \end{cases}$$

P	x
True	1
False	0

Logical negation

$$\begin{aligned} P &: \quad x \\ \neg P &: \quad 1 - x \end{aligned}$$

P	$\neg P$	x	$1 - x$
True	False	1	0
False	True	0	1

Logical conjunction

$$
\begin{aligned}
P &: \quad x \\
Q &: \quad y \\
P \wedge Q &: \quad xy
\end{aligned}
$$

P	Q	$P \wedge Q$	x	y	xy
True	True	True	1	1	1
True	False	False	1	0	0
False	True	False	0	1	0
False	False	False	0	0	0

Note: if y and y are both variables, non linear formulation. Use a combination of two constraints instead.

Logical disjunction

$$\begin{aligned} P &: \quad x \\ Q &: \quad y \\ P \vee Q &: \quad x + y \geq 1 \end{aligned}$$

P	Q	$P \vee Q$	x	y	$x + y \geq 1$
True	True	True	1	1	Yes
True	False	True	1	0	Yes
False	True	True	0	1	Yes
False	False	False	0	0	No

Generalization: $P_1 \vee \ldots \vee P_r$. $\sum_{i=1}^{r} x_i \geq 1$.

Logical exclusive disjunction

$$
\begin{aligned}
P &: \quad x \\
Q &: \quad y \\
P \oplus Q &: \quad x + y = 1
\end{aligned}
$$

P	Q	$P \oplus Q$	x	y	$x + y = 1$
True	True	False	1	1	No
True	False	True	1	0	Yes
False	True	True	0	1	Yes
False	False	False	0	0	No

Logical implication

$$\begin{aligned} P &: \quad x \\ Q &: \quad y \\ P \Rightarrow Q &: \quad x \leq y \end{aligned}$$

P	Q	$P \Rightarrow Q$	x	y	$x \leq y$
True	True	True	1	1	Yes
True	False	False	1	0	No
False	True	True	0	1	Yes
False	False	True	0	0	Yes

Note: $P \Rightarrow Q$ is equivalent to $\neg P \vee Q$.

Logical equivalence

$$\begin{aligned} P &: \quad x \\ Q &: \quad y \\ P \Leftrightarrow Q &: \quad x = y \end{aligned}$$

P	Q	$P \Leftrightarrow Q$	x	y	$x = y$
True	True	True	1	1	Yes
True	False	False	1	0	No
False	True	False	0	1	No
False	False	True	0	0	Yes

Optional constraint $\geq$

- z is a binary variable.
- If $z = 1$, the constraint $f(x) \geq a$ must be verified.
- If $z = 0$, the constraint $f(x) \geq a$ must not be verified.

Assumption: f is bounded from below by L.

$$f(x) - L \geq 0 \text{ is always true.}$$

$$f(x) \geq L + (a - L)z$$

Optional constraint $\leq$

- z is a binary variable.
- If $z = 1$, the constraint $f(x) \leq a$ must be verified.
- If $z = 0$, the constraint $f(x) \leq a$ must not be verified.

Assumption: f is bounded from above by M.

$$f(x) \leq M \text{ is always true.}$$

$$f(x) \leq az + (1 - z)M$$

Disjunctive constraints

- Constraint 1: $f(x) \geq a$.
- Constraint 2: $g(x) \geq b$.
- One of them must be verified, but not necessarily both.

Assumption: f and g are bounded from below.

$$f(x) \geq L_f \text{ and } g(x) \geq L_g \text{ are always true.}$$

Introduce a binary variable z

$$f(x) \geq L_f + (a - L_f)z$$

$$g(x) \geq L_g + (b - L_g)(1 - z)$$

Linearization

Non linear specification

$$xy = z,\ \ x, y, z \in \{0, 1\}.$$

$$\begin{aligned} x + y &\le 1 + z \\ z &\le x \\ z &\le y. \end{aligned}$$

x	y	z	$x + y \le 1 + z$	$z \le x$	$z \le y$	$xy = z$
1	1	1	*Yes*	*Yes*	*Yes*	*Yes*
1	1	0	*No*	*Yes*	*Yes*	*No*
1	0	1	*Yes*	*Yes*	*No*	*No*
1	0	0	*Yes*	*Yes*	*Yes*	*Yes*
0	1	1	*Yes*	*No*	*Yes*	*No*
0	1	0	*Yes*	*Yes*	*Yes*	*Yes*
0	0	1	*Yes*	*No*	*No*	*No*
0	0	0	*Yes*	*Yes*	*Yes*	*Yes*

Definitions

Motivation

- Discrete optimization involves decision variables that must be integer.
- We define here some variants of discrete optimization problems.

Discrete optimization

Integer Linear Problem

$$\min_{x \in \mathbb{R}^n} c^T x$$

subject to

$$Ax = b$$
$$x \geq 0$$
$$x \in \mathbb{Z}^n$$

Mixed Integer Linear Problem

$$\min_{x \in \mathbb{R}^n, y \in \mathbb{R}^p} c_x^T x + c_y^T y$$

subject to

$$A_x x + A_y y = b$$
$$x \geq 0$$
$$y \geq 0$$
$$y \in \mathbb{Z}^p$$

Binary linear optimization problem

$$\min_{x \in \mathbb{N}^n} c^T x$$

subject to

$$Ax = b$$
$$x \in \{0, 1\}^n$$

Transformation

Consider $x \in \mathbb{N}$, $x \leq u$.

$$x = \sum_{i=0}^{K-1} 2^i z_i.$$

$$K = \lceil \log_2(u+1) \rceil.$$

$$Example : u = 5, K = 3, x = z_0 + 2z_1 + 4z_2.$$

$$\sum_{i=0}^{K-1} 2^i z_i \leq u.$$

Combinatorial optimization

$$\min f(x)$$

subject to

$$x \in \mathcal{F} \text{ a large finite set.}$$

Knapsack

Motivation

- We review some classical combinatorial optimization problems.
- We show how they can be modeled as a (mixed) integer linear optimization problem.
- We start by the knapsack problem.

The knapsack problem

- Patricia prepares a hike in the mountain.
- She has a knapsack with capacity Wkg.
- She considers carrying a list of n items.
- Each item has a utility u_i and a weight w_i.
- What items should she take to maximize the total utility, while fitting in the knapsack?

Modeling

Decision variables

$$x_i = \begin{cases} 1 & \text{if item } i \text{ goes into the knapsack,} \\ 0 & \text{otherwise} \end{cases}$$

Objective function

$$\max f(x) = \sum_{i=1}^{n} u_i x_i$$

Constraints

$$\sum_{i=1}^{n} w_i x_i \leq W, \; x_i \in \{0, 1\}, i = 1, \dots, n$$

The set covering problem

- After the FIFA World Cup, Camille wants to complete her collection of stickers.
- She can buy collections of stickers from her schoolmates.
- In each collection, there are stickers that she needs, but also stickers that she does not need.
- The schoolmates do not accept to sell stickers individually. The whole collection has to be purchased.
- Camille must decide which collections to purchase, in order to complete her own album, at a minimum price.

Definition

Data

- A set U of m elements.
- $S_i \subseteq U$, $i = 1, \ldots, n$.
- $a_{ij} = 1$ if element j belongs to subset S_i.
- Costs: c_i.

Objective

Choose J subsets S_{i_j}, $j = 1, \ldots, J$, of minimal total cost such that

$$\bigcup_{j=1}^{J} S_{i_j} = U.$$

Modeling

Decision variables

$$x_i = \begin{cases} 1 & \text{if subset } i \text{ is selected} \\ 0 & \text{otherwise} \end{cases},$$

Objective function

$$\min f(x) = \sum_{i=1}^{n} c_i x_i$$

Constraints

$$\sum_{i=1}^{n} a_{ij} x_i \geq 1, j = 1, \ldots, m \; x_i \in \{0, 1\}, i = 1, \ldots, n$$

The traveling salesman problem

- Consider a network $(\mathcal{N}, \mathcal{A})$ with n nodes representing cities.
- For any pair (i, j) of cities, the distance c_{ij} between them is known.
- Find the shortest possible itinerary that starts from the home town of the salesman, visit all other cities, and come back home.

Modeling

Decision variables

$$x_{ij} = \begin{cases} 1 & \text{if } j \text{ is visited just after } i \\ 0 & \text{otherwise} \end{cases},$$

Objective function

$$\min f(x) = \sum_{(i,j) \in \mathcal{A}}^{n} c_{ij} x_{ij}$$

Modeling

Constraints

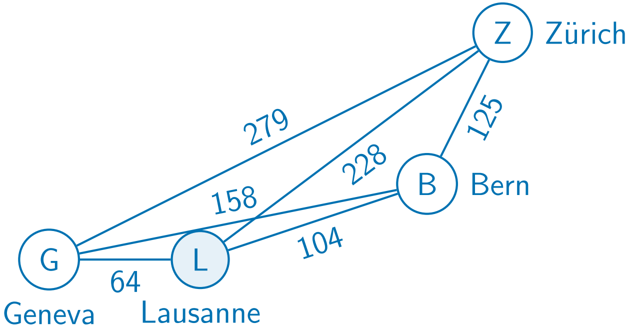
Exactly one successor in the tour

$$\sum_{j|(i,j)\in\mathcal{A}} x_{ij} = 1 \ \forall i \in \mathcal{N}$$

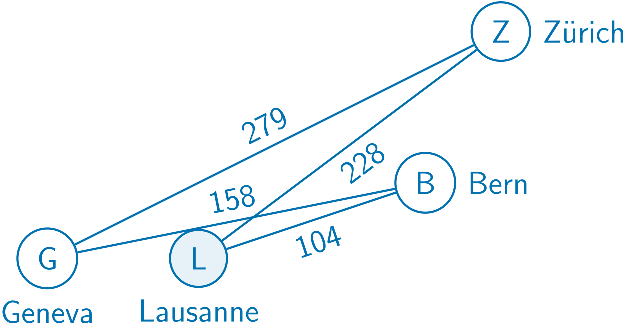
Exactly one predecessor in the tour

$$\sum_{i|(i,j)\in\mathcal{A}} x_{ij} = 1 \ \forall j \in \mathcal{N}$$

Network

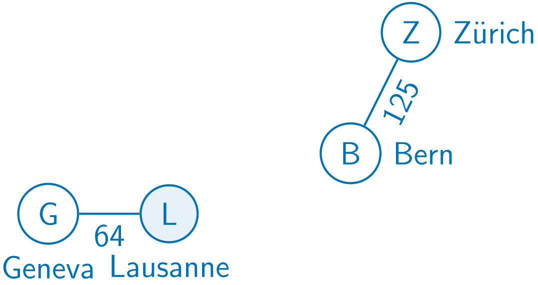


A tour



769km

Subtours



- $x_{LG} = x_{GL} = 1$, $x_{ZB} = x_{BZ} = 1$.
- 378km.
- There is exactly one predecessor for each city.
- There is exactly one successor for each city.
- There are several ways to eliminate subtours. We present one here.

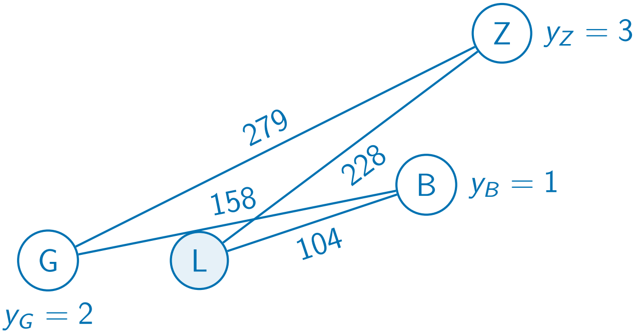
New variables

$$y_i : \text{ position of city } i \text{ in the tour }.$$

For each i and j different from home:

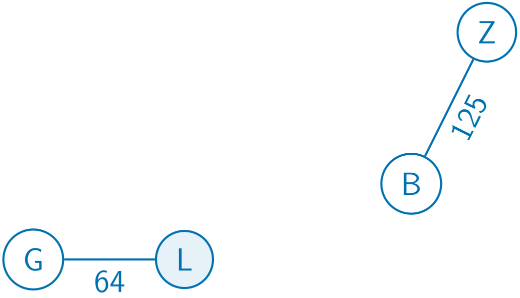
$$x_{ij} = 1 \Longrightarrow y_j \geq y_i + 1,$$

A tour



A subtour

The constraints cannot be verified in subtours not involving home.



$$y_B \geq y_Z + 1 \text{ and } y_Z \geq y_B + 1 : \textit{impossible}.$$

Additional constraints

$$x_{ij} = 1 \Longrightarrow y_j \geq y_i + 1.$$

Modeling exercise, using optional constraint see before.

$$x_{ij}(n-1) + y_i - y_j \leq n - 2.$$

If x_{ij}=1

$$(n-1) + y_i - y_j \leq n - 2, \quad y_j \geq y_i + 1$$

If x_{ij}=0

$$y_i - y_j \leq n - 2$$

Always verified because cities are numbered from 1 to $n-1$

Traveling salesman problem

$$\min_{x \in \mathbb{Z}^{n(n-1)}, y \in \mathbb{Z}^{(n-1)}} \sum_{i=1}^{n} \sum_{j \neq i} c_{ij} x_{ij}$$

subject to

$$\begin{aligned}
\sum_{j \neq i} x_{ij} &= 1 && \forall i = 1, \ldots, n, \\
\sum_{i \neq j} x_{ij} &= 1 && \forall j = 1, \ldots, n, \\
x_{ij}(n-1) + y_i - y_j &\leq n-2, && \forall i = 2, \ldots, n, j = 2, \ldots, n, i \neq j, \\
x_{ij} &\in \{0, 1\} && \forall i = 1, \ldots, n, j = 1, \ldots, n, i \neq j, \\
y_i &\geq 0 && \forall i = 2, \ldots, n.
\end{aligned}$$

The curse of dimensionality

Motivation

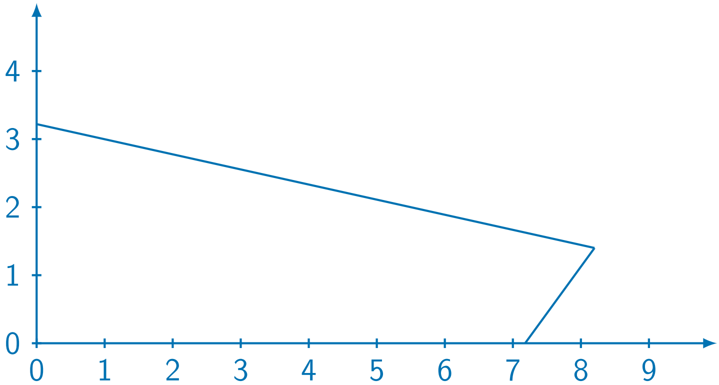
- When we have introduced the transhipment problem, we have seen that some problems can be solved by ignoring the integrality constraints, and the solution would be guaranteed to be integer.
- Unfortunately, this property occurs only exceptionally.
- There is no optimality condition for discrete optimization.

An example

$$\min_{x \in \mathbb{N}^2} -3x_1 - 13x_2$$

subject to

$$\begin{aligned} 2x_1 + 9x_2 &\leq 29 \\ 11x_1 - 8x_2 &\leq 79. \end{aligned}$$

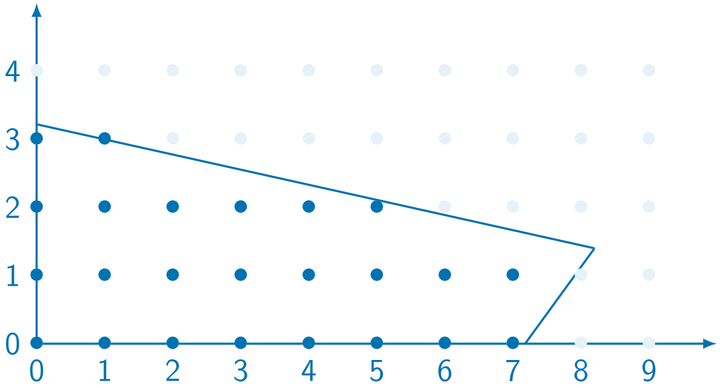


An example

$$\min_{x \in \mathbb{N}^2} -3x_1 - 13x_2$$

subject to

$$\begin{aligned} 2x_1 + 9x_2 &\leq 29 \\ 11x_1 - 8x_2 &\leq 79. \end{aligned}$$



Enumeration

x_1	x_2	$c^T x$	x_1	x_2	$c^T x$	x_1	x_2	$c^T x$
0	0	0	2	0	-6	4	2	-38
0	1	-13	2	1	-19	5	0	-15
0	2	-26	2	2	-32	5	1	-28
0	3	-39	3	0	-9	5	2	-41
1	0	-3	3	1	-22	6	0	-18
1	1	-16	3	2	-35	6	1	-31
1	2	-29	4	0	-12	7	0	-21
1	3	-42	4	1	-25	7	1	-34

Solution: (1,3) -42

Enumeration: the binary knapsack problem

- n items.
- Number of possibilities: 2^n.
- For each of them,
  1. check feasibility,
  2. calculate the objective function.
- About $2n$ floating point operations.
- Processor: 1 Teraflops. 10^{12}

- $n = 34$: 1 second
- $n = 40$: 1 minute
- $n = 45$: 1 hour
- $n = 50$: 1 day
- $n = 58$: 1 year
- $n = 69$: 2583 years. Christian Era
- $n = 78$: 1 500 000 years. homo erectus.
- $n = 91$: 10^{10} years. Age of the universe.

Enumeration: the binary knapsack problem

1 Teraflops

- $n = 50$: 1 day.
- $n = 69$: 2,583 years.
- $n = 78$: 1,500,000 years.
- $n = 91$: 10^{10} years.

1000 Teraflops

- $n = 59$: 1 day.
- $n = 69$: 2.6 years.
- $n = 78$: 1,500 years.
- $n = 91$: 10 millions years.

Relaxation

Motivation

- We know how to solve linear optimization problems.
- We do not know how to solve discrete optimization problems.
- But if we forget about the integrality constraints, we obtain a linear optimization problem.
- It is called a relaxation, and happens to be very useful.

Relaxation

Original problem

$$\min_{x \in \mathbb{R}^{n_x}, y \in \mathbb{Z}^{n_y}, z \in \mathbb{N}^{n_z}} f(x, y, z)$$

subject to

$$\begin{aligned} g(x, y, z) &\leq 0 \\ h(x, y, z) &= 0 \\ y &\in \mathbb{Z}^{n_y} \\ z &\in \{0, 1\}^{n_z} \end{aligned}$$

where

- $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \times \mathbb{R}^{n_z} \to \mathbb{R}$,
- $g : \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \times \mathbb{R}^{n_z} \to \mathbb{R}^m$,
- $h : \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \times \mathbb{R}^{n_z} \to \mathbb{R}^p$.

Relaxation

$$\min_{x \in \mathbb{R}^{n_x}, y \in \mathbb{R}^{n_y}, z \in \mathbb{R}^{n_z}} f(x, y, z)$$

subject to

$$\begin{aligned} g(x, y, z) &\leq 0 \\ h(x, y, z) &= 0 \\ y &\in \mathbb{R}^{n_y} \\ z &\in [0, 1]^{n_z} \end{aligned}$$

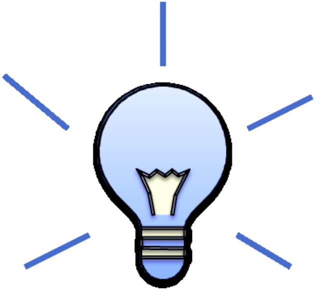
Lower bound

- Discrete optimization P: optimal solution: (x^*, y^*, z^*).
- Relaxation $R(P)$: optimal solution: (x_R^*, y_R^*, z_R^*).

$$f(x_R^*, y_R^*, z_R^*) \leq f(x^*, y^*, z^*).$$

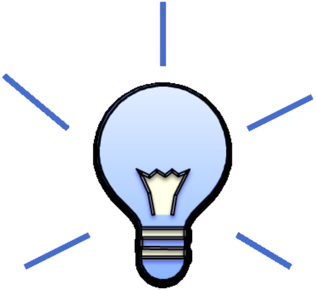
Proof: the integer solution (x^*, y^*, z^*) verifies the constraints of the relaxation.
Note: it is valid only for global minima.

Mixed Integer Linear Problems



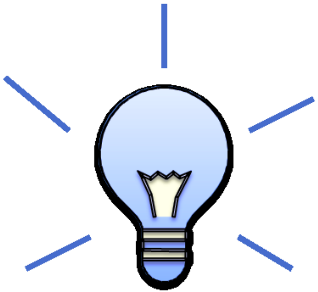
- Consider the relaxation $R(P)$.

Mixed Integer Linear Problems



- Consider the relaxation $R(P)$.
- Calculate (x_R^*, y_R^*, z_R^*) using the simplex algorithm.

Mixed Integer Linear Problems



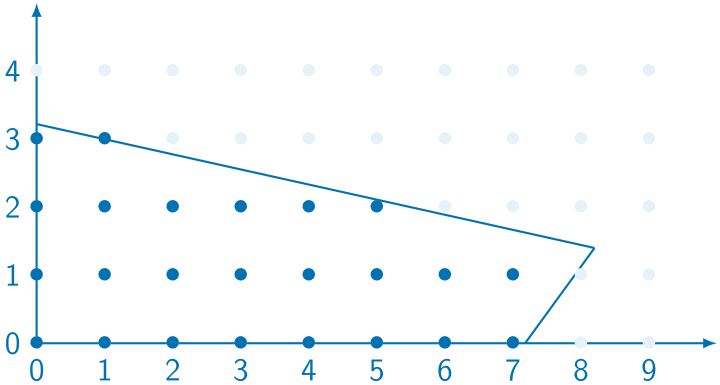
- Consider the relaxation $R(P)$.
- Calculate (x_R^*, y_R^*, z_R^*) using the simplex algorithm.
- Round the solution to the nearest integer values.

An example

$$\min_{x \in \mathbb{N}^2} -3x_1 - 13x_2$$

subject to

$$\begin{aligned} 2x_1 + 9x_2 &\leq 29 \\ 11x_1 - 8x_2 &\leq 79. \end{aligned}$$

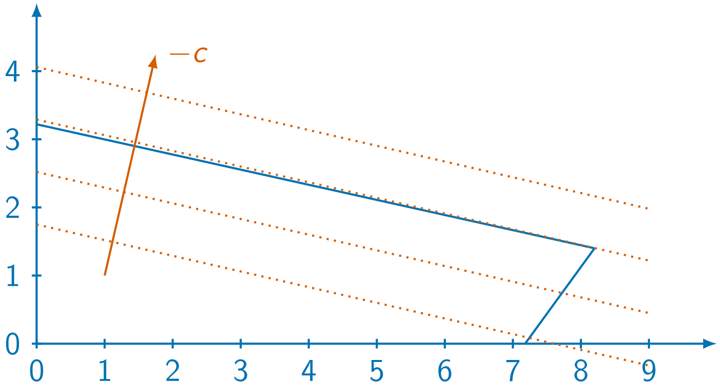


Solving the relaxation

$$\min_{x \in \mathbb{N}^2} -3x_1 - 13x_2$$

subject to

$$\begin{aligned} 2x_1 + 9x_2 &\leq 29 \\ 11x_1 - 8x_2 &\leq 79. \end{aligned}$$

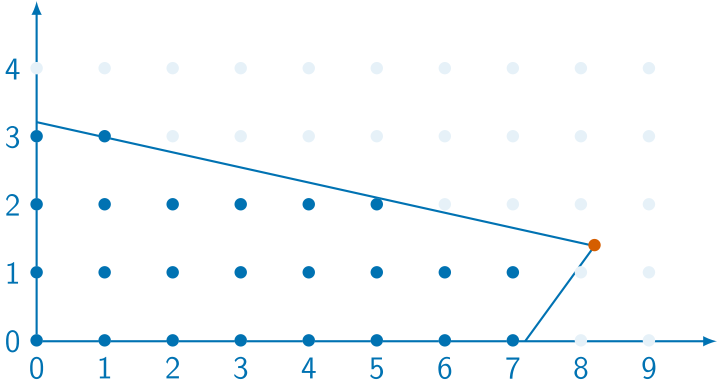


Rounding the solution

$$\min_{x \in \mathbb{N}^2} -3x_1 - 13x_2$$

subject to

$$\begin{aligned} 2x_1 + 9x_2 &\leq 29 \\ 11x_1 - 8x_2 &\leq 79. \end{aligned}$$



Rounding the solution

In this example

- Rounding always produce an infeasible point.
- The optimal solution $(1, 3)$ is far from the solution of the relaxation.

Branch & Bound

Motivation

- In the absence of optimality conditions, enumeration is the only way to find the optimal solution.
- However, it is most of the time impossible to perform explicitly due to the curse of dimensionality.
- The branch & bound method is some sort of implicit enumeration technique, that attacks the huge set of feasible solutions using a "divide and conquer" strategy.

Combinatorial optimization

$$\min_x f(x)$$

subject to

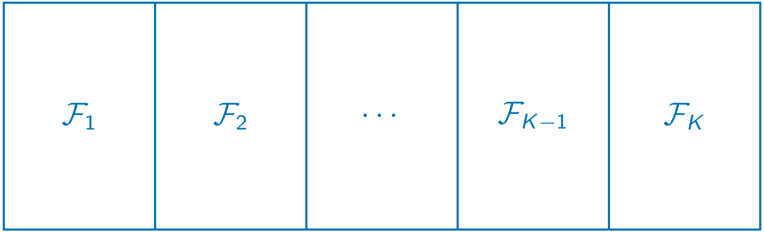
$$x \in \mathcal{F},$$

where $\mathcal{F}$ is a large set containing a finite number of elements.

Divide

$\mathcal{F}$

Divide



Divide

Problem P

$$\min_x f(x)$$

subject to

$$x \in \mathcal{F},$$

Solution: x^*.

Problem P_k

$$\min_x f(x)$$

subject to

$$x \in \mathcal{F}_k,$$

Solution: x_k^*.

Conquer

Theorem 26.1

Consider i such that

$$f(x_i^*) \leq f(x_k^*), \; k = 1, \ldots, K.$$

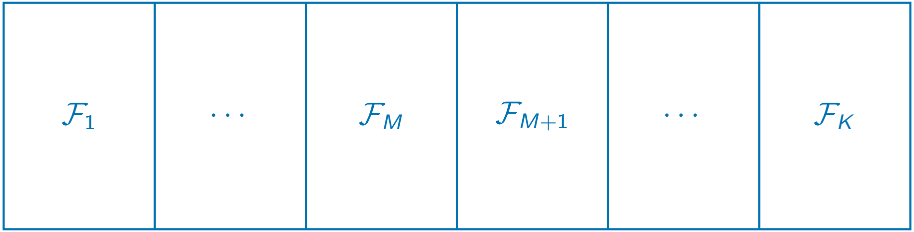
Then,

$$f(x^*) = f(x_i^*),$$

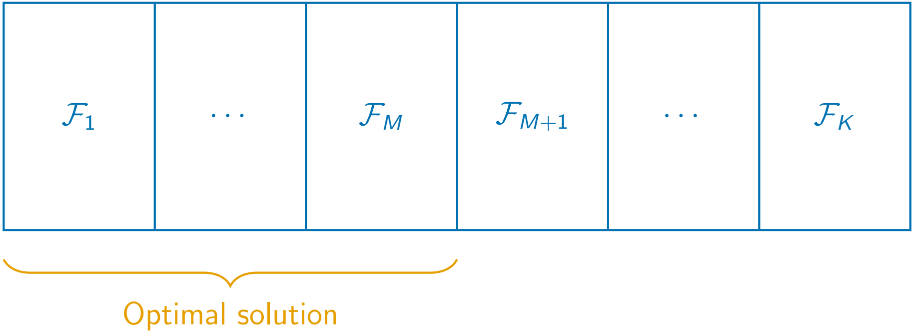
and x_i^* is solution of P.

$f(x^*) \leq f(x_i^*)$
$\exists j$ such that $x^* \in \mathcal{F}_j$
Optimality of x_j^*: $f(x_j^*) \leq f(x) \forall x \in \mathcal{F}_j$
$x^* \in \mathcal{F}_j \Rightarrow f(x_j^*) \leq f(x^*)$
$f(x^*) \leq f(x_i^*) \leq f(x_j^*) \leq f(x^*)$

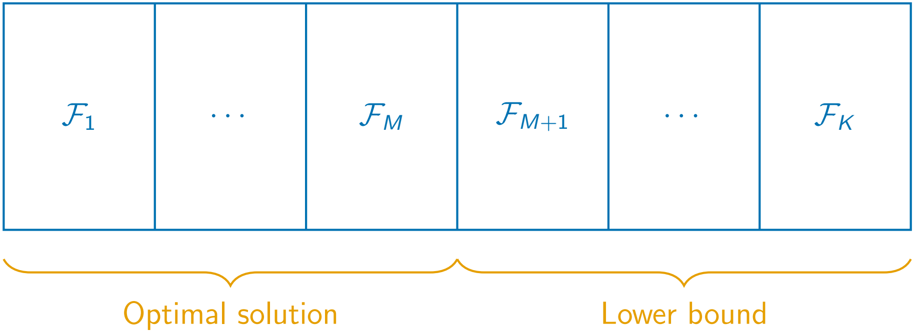
Divide



Divide


$\mathcal{F}_1$
$\cdots$
$\mathcal{F}_M$
$\mathcal{F}_{M+1}$
$\cdots$
$\mathcal{F}_K$
Optimal solution

Divide


$\mathcal{F}_1$
$\cdots$
$\mathcal{F}_M$
$\mathcal{F}_{M+1}$
$\cdots$
$\mathcal{F}_K$
Optimal solution
Lower bound

Divide

Problem P

$$\min_x f(x)$$

subject to

$$x \in \mathcal{F}.$$

Solution: x^*.

Problem P_k, $k \leq M$

$$\min_x f(x)$$

subject to

$$x \in \mathcal{F}_k.$$

Solution: x_k^*.

Problem P_k, $k > M$

$$\min_x f(x)$$

subject to

$$x \in \mathcal{F}_k.$$

Lower bound: $\ell(P_m) \leq f(x_k^*)$.

Conquer

Corollary 26.2

Consider i such that

$$f(x_i^*) \leq f(x_k^*),\ k = 1, \ldots, M,$$

and

$$f(x_i^*) \leq \ell(P_k),\ k = M{+}1, \ldots, K.$$

Then,

$$f(x^*) = f(x_i^*),$$

and x_i^* is solution of P.

$\ell(P_k) \leq f(x_k^*)$ for each $k > M$
Therefore, $f(x_i^*) \leq f(x_k^*)$ for all k
The previous theorem applies.

Example

Problem P

$$\min x_1 - 2x_2$$

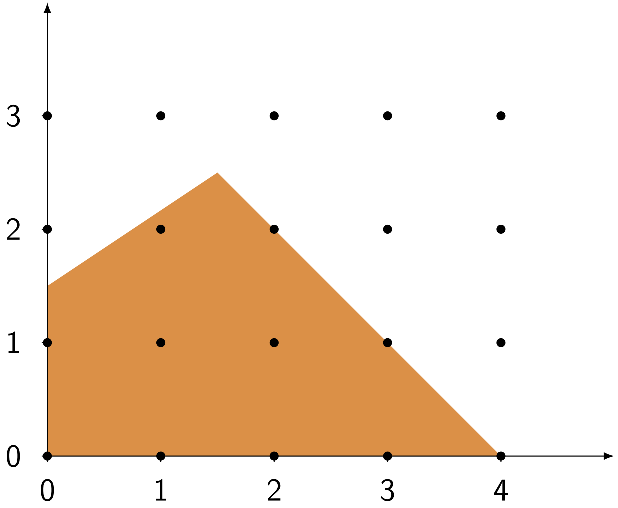
subject to

$$\begin{aligned} -4x_1 + 6x_2 &\leq 9 \\ x_1 + x_2 &\leq 4 \\ x_1, x_2 &\geq 0 \\ x_1, x_2 &\in \mathbb{N}. \end{aligned}$$

Upper bound

(0,0) is feasible: U=0

Example



Lower bound: relaxation

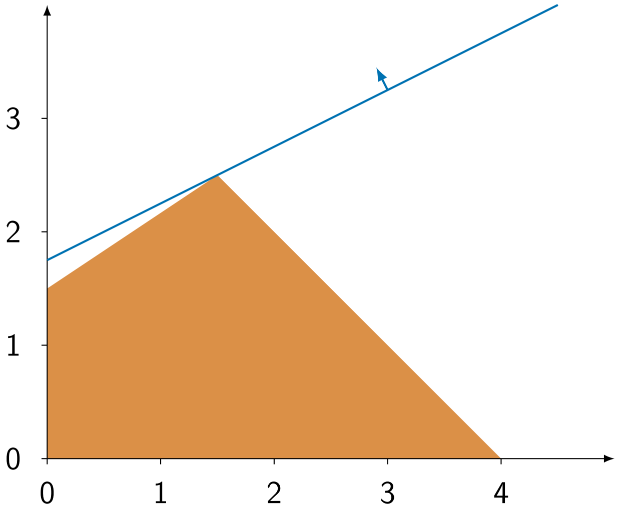
Problem $R(P)$:

$$\min x_1 - 2x_2$$

subject to

$$\begin{aligned} -4x_1 + 6x_2 &\leq 9 \\ x_1 + x_2 &\leq 4 \\ x_1, x_2 &\geq 0 \end{aligned}$$

Lower bound

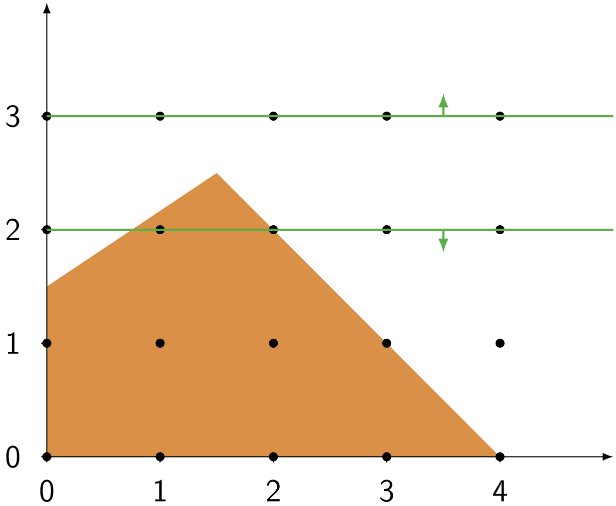


Divide

- Optimal solution of $R(P)$: $(1.5, 2.5)$
- $\ell(P)$: -3.5

P_1	P_2
$\min x_1 - 2x_2$	$\min x_1 - 2x_2$
s.c.	s.c.
$-4x_1 + 6x_2 \leq 9$	$-4x_1 + 6x_2 \leq 9$
$x_1 + x_2 \leq 4$	$x_1 + x_2 \leq 4$
$x_1, x_2 \geq 0$	$x_1, x_2 \geq 0$
$x_1, x_2 \in \mathbb{N}$	$x_1, x_2 \in \mathbb{N}$
$x_2 \leq 2$	$x_2 \geq 3$

Divide

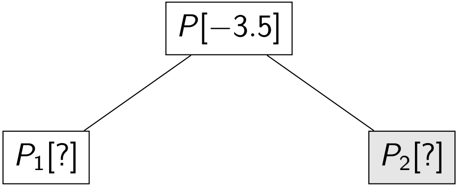


Tree representation

Upper bound

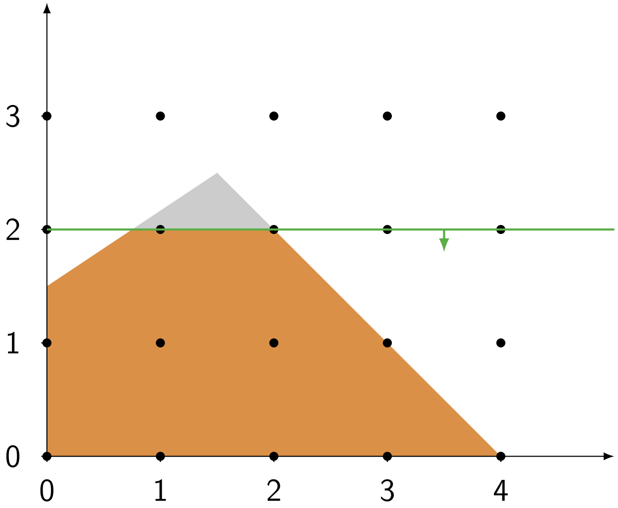
$U = 0$

Tree



P_2 is infeasible.

Problem P_1



Problem P_1

Note

The feasible set of P_1 is exactly the same as P.

P_1: lower bound

Problem P_1

$$\min x_1 - 2x_2$$

subject to

$$\begin{aligned} -4x_1 + 6x_2 &\leq 9 \\ x_1 + x_2 &\leq 4 \\ x_1, x_2 &\geq 0 \\ x_2 &\leq 2 \\ x_1, x_2 &\in \mathbb{N}. \end{aligned}$$

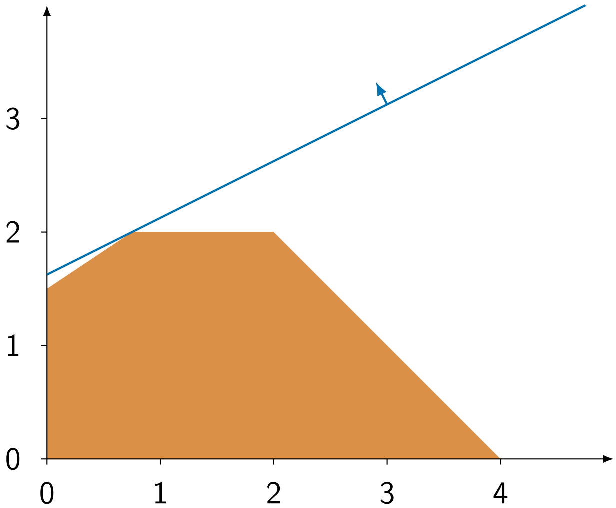
Relaxation $R(P_1)$

$$\min x_1 - 2x_2$$

subject to

$$\begin{aligned} -4x_1 + 6x_2 &\leq 9 \\ x_1 + x_2 &\leq 4 \\ x_1, x_2 &\geq 0 \\ x_2 &\leq 2 \end{aligned}$$

P_1: lower bound

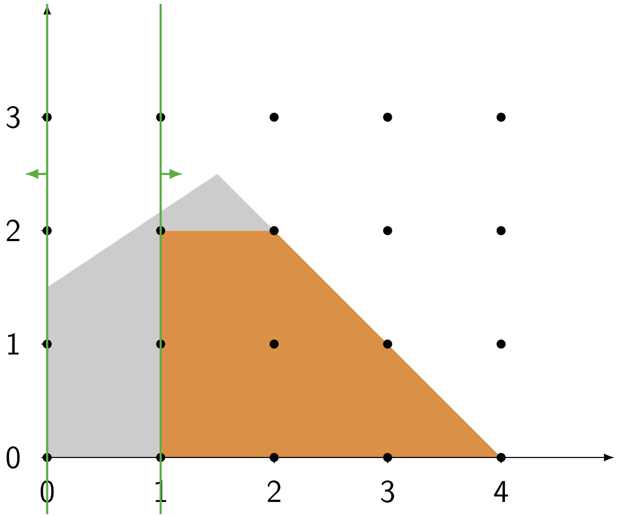


P_1: divide

- Optimal solution of $R(P_1)$: $(0.75, 2)$
- $\ell(P_1)$: -3.25

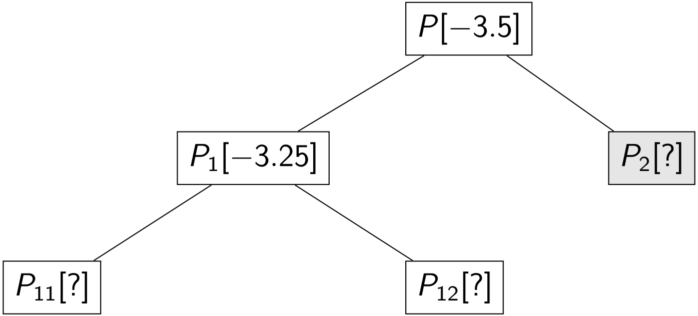
P_{11}	P_{12}
$\min x_1 - 2x_2$	$\min x_1 - 2x_2$
s.c.	s.c.
$-4x_1 + 6x_2 \leq 9$	$-4x_1 + 6x_2 \leq 9$
$x_1 + x_2 \leq 4$	$x_1 + x_2 \leq 4$
$x_1, x_2 \geq 0$	$x_1, x_2 \geq 0$
$x_1, x_2 \in \mathbb{N}$	$x_1, x_2 \in \mathbb{N}$
$x_2 \leq 2$	$x_2 \leq 2$
$x_1 \leq 0$	$x_1 \geq 1$

P_1: divide



Tree representation

$U = 0$



P_{11}: lower bound

Problem P_{11}

$$\min x_1 - 2x_2$$

subject to

$$\begin{aligned}
-4x_1 + 6x_2 &\leq 9, \\
x_1 + x_2 &\leq 4, \\
x_1, x_2 &\geq 0, \\
x_2 &\leq 2, \\
x_1 &\leq 0, \\
x_1, x_2 &\in \mathbb{N}.
\end{aligned}$$

Relaxation $R(P_{11})$

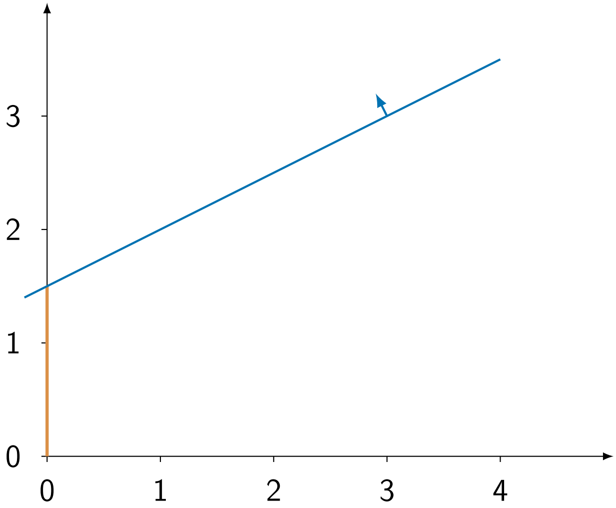
$$\min x_1 - 2x_2$$

subject to

$$\begin{aligned}
-4x_1 + 6x_2 &\leq 9, \\
x_1 + x_2 &\leq 4, \\
x_1, x_2 &\geq 0, \\
x_2 &\leq 2, \\
x_1 &\leq 0.
\end{aligned}$$

Note: $x_1 = 0$.

P_{11}: lower bound



P_{11}: lower bound

- Optimal solution of $R(P_{11})$: $(0, 1.5)$
- $\ell(P_{11})$: -3

P_{12}: lower bound

Problem P_{12}

$$\min x_1 - 2x_2$$

subject to

$$\begin{aligned} -4x_1 + 6x_2 &\leq 9 \\ x_1 + x_2 &\leq 4 \\ x_1, x_2 &\geq 0 \\ x_2 &\leq 2 \\ x_1 &\geq 1 \\ x_1, x_2 &\in \mathbb{N}. \end{aligned}$$

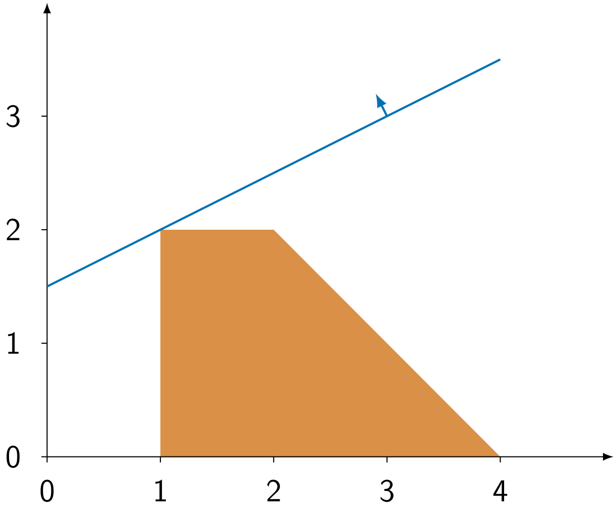
Relaxation $R(P_{12})$

$$\min x_1 - 2x_2$$

subject to

$$\begin{aligned} -4x_1 + 6x_2 &\leq 9 \\ x_1 + x_2 &\leq 4 \\ x_1, x_2 &\geq 0 \\ x_2 &\leq 2 \\ x_1 &\geq 1 \end{aligned}$$

P_{12}: lower bound



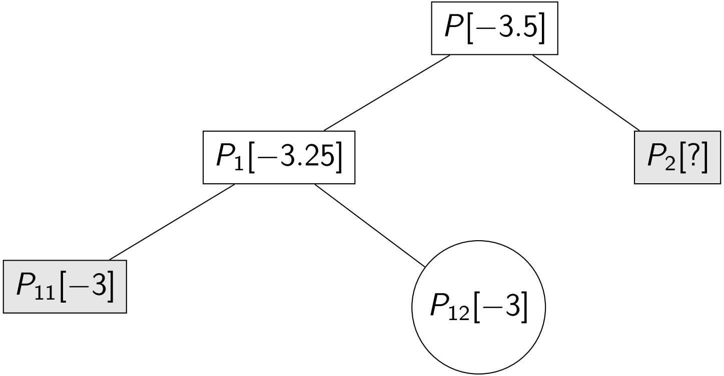
P_{12}: lower bound

- Optimal solution of $R(P_{12})$: $(1, 2)$
- $\ell(P_{12})$: -3

Integer solution
It is the optimal solution of P_{12}.
Feasible solution for P
Better upper bound: $U = -3$.

Tree representation

$U = -3$



Optimal solution: $(1, 2)$.

Branch & bound algorithm

At each iteration, we maintain

- a list of active subproblems $\mathcal{S} = \{P_1, P_2, \ldots\}$,
- an upper bound U, that is the value of the objective function at the best feasible solution encountered so far.
- Initialization:
  - Either $U = +\infty$,
  - or $U = f(x)$, where x is a known feasible solution.
  - $\mathcal{S} = \{P\}$.

Branch & bound algorithm

Iteration:

- Consider an active subproblem P_k.
- If P_k is infeasible, remove it from the list.
- Otherwise, calculate a lower bound $\ell(P_k)$.
- If $U \leq \ell(P_k)$, remove P_k from the list.
- Otherwise,
  - either solve P_k directly,
  - or partition its feasible set, and create new subproblems, that are added to the list.

Summary

- Modeling logical rules.
- Types of problems: integer, mixed, binary, and combinatorial.
- Examples: knapsack, set covering, traveling salesman.
- No optimality condition: the curse of dimensionality.
- Relaxation.
- Branch & bound.