Graph Theory and Optimization
Introduction on Linear Programming

Nicolas Nisse

Université Côte d'Azur, Inria, CNRS, I3S, France

October 2018

Thank you to F. Giroire for his slides

Outline

Motivation

Why linear programming is a very important tool?

- A lot of problems can be formulated as linear programmes, and
- There exist efficient methods to solve them
- or at least give good approximations.

- Solve difficult problems: e.g. original example given by Dantzig (1947). Best assignment of 70 people to 70 tasks.

$\rightarrow$ Magic algorithmic box.

Outline

What is a linear programme?

- Optimization problem consisting in
  - maximizing (or minimizing) a linear objective function
  - of n decision variables
  - subject to a set of constraints expressed by linear equations or inequalities.
- Originally, military context: "programme"="resource planning". Now "programme"="problem"
- Terminology due to George B. Dantzig, inventor of the Simplex Algorithm (1947)

Terminology

x_1, x_2 — Decision variables (generally: $\in \mathbb{R}$)

$$\begin{array}{rc} \max & 350x_1 + 300x_2 \\ \text{subject to} & \\ & x_1 + x_2 \leq 200 \\ & 9x_1 + 6x_2 \leq 1566 \\ & 12x_1 + 16x_2 \leq 2880 \\ & x_1, x_2 \geq 0 \end{array}$$

Objective function (linear!!)

Constraints (linear!!)

Terminology

x_1, x_2 Decision variables

$$\max \quad 350x_1 + 300x_2$$ Objective function

subject to

$$\begin{aligned} x_1 + x_2 &\leq 200 \\ 9x_1 + 6x_2 &\leq 1566 \\ 12x_1 + 16x_2 &\leq 2880 \\ x_1, x_2 &\geq 0 \end{aligned}$$ Constraints

In linear programme: objective function + constraints are all linear

Typically (not always): variables are non-negative

If variables are integer: system called Integer Programme (IP)

Terminology

Linear programmes can be written under the standard form:

$$\begin{array}{lrcll} \text{Maximize} & \sum_{j=1}^{n} c_j x_j & & & \\ \text{Subject to:} & \sum_{j=1}^{n} a_{ij} x_j & \leq & b_i & \text{for all } 1 \leq i \leq m \\ & x_j & \geq & 0 & \text{for all } 1 \leq j \leq n. \end{array}$$

- the problem is a maximization;
- all constraints are inequalities (and not equations);
- all variables are non-negative.

Outline

Example 1: a resource allocation problem

A company produces copper cable of 5 and 10 mm of diameter on a single production line with the following constraints:

- The available copper allows to produces 21 meters of cable of 5 mm diameter per week. Moreover, one meter of 10 mm diameter copper consumes 4 times more copper than a meter of 5 mm diameter copper.
- Due to demand, the weekly production of 5 mm cable is limited to 15 meters and the production of 10 mm cable should not exceed 40% of the total production.
- Cable are respectively sold 50 and 200 euros the meter.

What should the company produce in order to maximize its weekly revenue?

Example 1: a resource allocation problem

A company produces copper cable of 5 and 10 mm of diameter on a single production line with the following constraints:

- The available copper allows to produces 21 meters of cable of 5 mm diameter per week. Moreover, one meter of 10 mm diameter copper consumes 4 times more copper than a meter of 5 mm diameter copper.
- Due to demand, the weekly production of 5 mm cable is limited to 15 meters and the production of 10 mm cable should not exceed 40% of the total production.
- Cable are respectively sold 50 and 200 euros the meter.

What should the company produce in order to maximize its weekly revenue?

Example 1: a resource allocation problem

Define two decision variables:

- x_1: the number of meters of 5 mm cables produced every week
- x_2: the number of meters of 10 mm cables produced every week

The revenue associated to a production (x_1, x_2) is

$$z = 50x_1 + 200x_2.$$

The capacity of production cannot be exceeded

$$x_1 + 4x_2 \leq 21.$$

Example 1: a resource allocation problem

Define two decision variables:

- x_1: the number of meters of 5 mm cables produced every week
- x_2: the number of meters of 10 mm cables produced every week

The revenue associated to a production (x_1, x_2) is

$$z = 50x_1 + 200x_2.$$

The capacity of production cannot be exceeded

$$x_1 + 4x_2 \leq 21.$$

Example 1: a resource allocation problem

Define two decision variables:

- x_1: the number of meters of 5 mm cables produced every week
- x_2: the number of meters of 10 mm cables produced every week

The revenue associated to a production (x_1, x_2) is

$$z = 50x_1 + 200x_2.$$

The capacity of production cannot be exceeded

$$x_1 + 4x_2 \leq 21.$$

Example 1: a resource allocation problem

The demand constraints have to be satisfied

$$x_2 \leq \frac{4}{10}(x_1 + x_2)$$

$$x_1 \leq 15$$

Negative quantities cannot be produced

$$x1 \geq 0, x2 \geq 0.$$

Exercise: Write the above programme in standard form

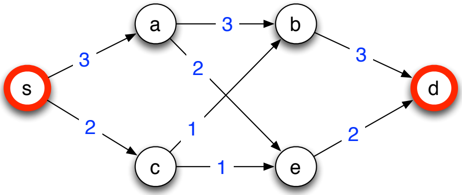
Example 1: a resource allocation problem

The model: To maximize the sell revenue, determine the solutions of the following linear programme x_1 and x_2:

$$
\begin{array}{rc}
\max & z = 50x_1 + 200x_2 \\
\text{subject to} & \\
 & x_1 + 4x_2 \leq 21 \\
 & -4x_1 + 6x_2 \leq 0 \\
 & x_1 \leq 15 \\
 & x_1, x_2 \geq 0
\end{array}
$$

Example 2: Maximum flow (Reminder on the Problem)

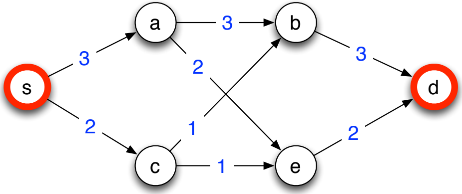
Directed graph: $D=(V,A)$, **source** $s \in V$, **destination** $d \in V$, capacity $c: A \rightarrow \mathbb{R}^+$.
$N^-(s)=\emptyset$ and $N^+(d)=\emptyset$



flow $f: A \rightarrow \mathbb{R}^+$ such that :

- capacity constraint: $\forall a \in A,\ f(a) \leq c(a)$
- conservation constraint: $\forall v \in V \setminus \{s,d\},\ \sum_{w \in N^-(v)} f(wv) = \sum_{w \in N^+(v)} f(vw)$
- value of flow: $v(f) = \sum_{w \in N^+(s)} f(sw)$.

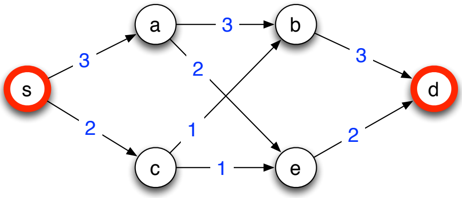
Example 2: Maximum flow (on an example)



Exercise: Give a LP computing a maximum flow in the above graph

hint: variables correspond to the expected solution

Example 2: Maximum flow (on an example)



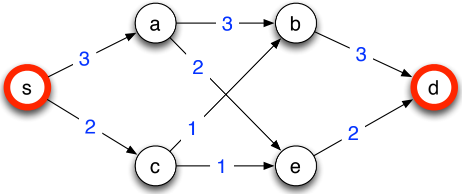
Exercise: Give a LP computing a maximum flow in the above graph

hint: variables correspond to the expected solution

Solution: flow $f : A \rightarrow \mathbb{R}^+$

Variables: $f_x \in \mathbb{R}^+$ for each $x \in A$

Example 2: Maximum flow (on an example)

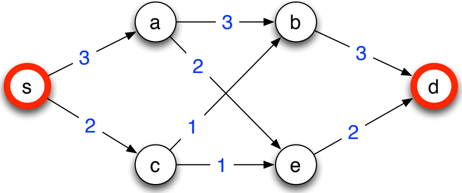


Exercise: Give a LP computing a maximum flow in the above graph
hint: variables correspond to the expected solution

Solution: flow $f : A \rightarrow \mathbb{R}^+$
Objective: maximize the flow leaving s
subject to:

Variables: $f_x \in \mathbb{R}^+$ for each $x \in A$
Max. $f_{sa} + f_{sc}$

Example 2: Maximum flow (on an example)



Exercise: Give a LP computing a maximum flow in the above graph

hint: variables correspond to the expected solution

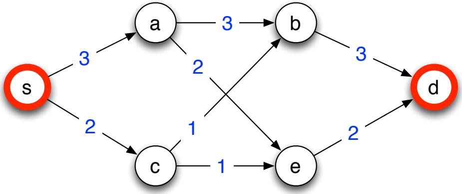
Solution: flow $f : A \rightarrow \mathbb{R}^+$ — Variables: $f_x \in \mathbb{R}^+$ for each $x \in A$

Objective: maximize the flow leaving s — Max. $f_{sa} + f_{sc}$

subject to:

Capacity constraints: $f_{sa} \leq 3$; $f_{sc} \leq 2$; $f_{ab} \leq 3$; $f_{ae} \leq 2$; $f_{cb} \leq 1$; $f_{ce} \leq 1$; $f_{bd} \leq 3$; $f_{ed} \leq 2$.

Example 2: Maximum flow (on an example)



Exercise: Give a LP computing a maximum flow in the above graph

hint: variables correspond to the expected solution

Solution: flow $f : A \rightarrow \mathbb{R}^+$ Variables: $f_x \in \mathbb{R}^+$ for each $x \in A$

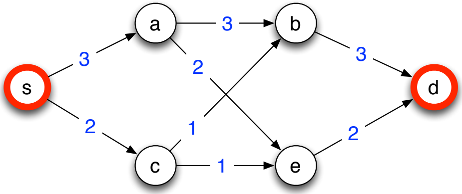
Objective: maximize the flow leaving s Max. $f_{sa} + f_{sc}$

subject to:

Capacity constraints: $f_{sa} \leq 3$; $f_{sc} \leq 2$; $f_{ab} \leq 3$; $f_{ae} \leq 2$; $f_{cb} \leq 1$; $f_{ce} \leq 1$; $f_{bd} \leq 3$; $f_{ed} \leq 2$.

Conservation constraints: $f_{sa} = f_{ab} + f_{ae}$; $f_{sc} = f_{cb} + f_{ce}$; $f_{ae} + f_{ce} = f_{ed}$ and $f_{ab} + f_{cb} = f_{bd}$.

Example 2: Maximum flow (on an example)



Exercise: Give a LP computing a maximum flow in the above graph
hint: variables correspond to the expected solution

Solution: flow $f : A \rightarrow \mathbb{R}^+$ — Variables: $f_x \in \mathbb{R}^+$ for each $x \in A$

Objective: maximize the flow leaving s — Max. $f_{sa} + f_{sc}$

subject to:

Capacity constraints: $f_{sa} \leq 3$; $f_{sc} \leq 2$; $f_{ab} \leq 3$; $f_{ae} \leq 2$; $f_{cb} \leq 1$; $f_{ce} \leq 1$; $f_{bd} \leq 3$; $f_{ed} \leq 2$.

Conservation constraints: $f_{sa} = f_{ab} + f_{ae}$; $f_{sc} = f_{cb} + f_{ce}$; $f_{ae} + f_{ce} = f_{ed}$ and $f_{ab} + f_{cb} = f_{bd}$.

Variables domain: $f_x \geq 0$ for any $x \in A$

Example 2: Maximum flow

$D = (V, A)$ be a graph with capacity $c : A \to \mathbb{R}^+$, and $s, t \in V$.
Problem: Compute a maximum flow from s to t.

Solution: $f : A \to \mathbb{R}^+$ $\Rightarrow$ variables f_a, for each $a \in A$

Objective function: maximize value of the flow $\sum_{u \in N^+(s)} f(su)$

Constraints:

- capacity constraints: $f(a) \leq c(a)$ for each $a \in A$
- flow conservation: $\sum_{u \in N^+(v)} f(vu) = \sum_{u \in N^-(v)} f(uv), \forall v \in V \setminus \{s, t\}$

Example 2: Maximum flow

$D = (V, A)$ be a graph with capacity $c : A \to \mathbb{R}^+$, and $s, t \in V$.
Problem: Compute a maximum flow from s to t.

$$
\begin{array}{llcll}
\text{Maximize} & \displaystyle\sum_{u \in N^+(s)} f(su) & & & \\
\text{Subject to:} & f(a) & \leq & c(a) & \text{for all } a \in A \\
& \displaystyle\sum_{u \in N^+(v)} f(vu) & = & \displaystyle\sum_{u \in N^-(v)} f(uv) & \text{for all } v \in V \setminus \{s, t\} \\
& f(a) & \geq & 0 & \text{for all } a \in A
\end{array}
$$

Outline

Solving Difficult Problems

- Difficulty: Large number of solutions.
  - Choose the best solution among 2^n or $n!$ possibilities: all solutions cannot be enumerated.
  - Complexity of studied problems: often NP-complete.
    but Polynomial-time solvable when variables are real !!
- Solving methods:
  - Optimal solutions:
    - Graphical method (2 variables only).
    - Simplex method. exponential-time, work well in practice
    - interior point method polynomial-time
    - Ellipsoid polynomial-time
  - Approximations:
    - Theory of duality (assert the quality of a solution).
    - Approximation algorithms.

Graphical Method

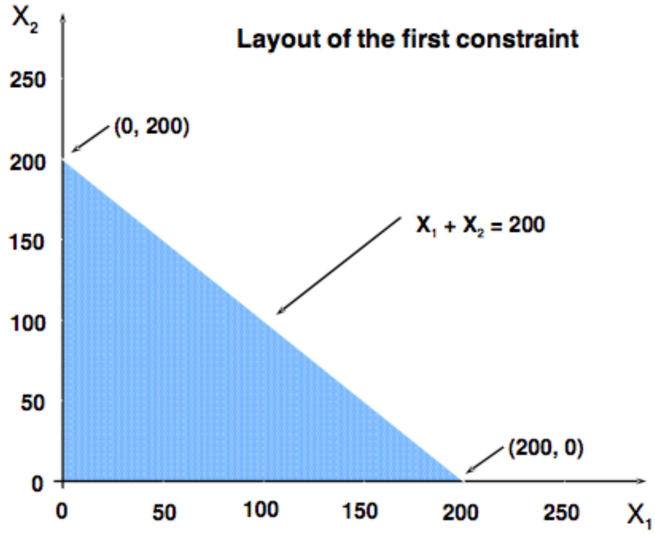
- The constraints of a linear programme define a zone of solutions.
- The best point of the zone corresponds to the optimal solution.
- For problem with 2 variables, easy to draw the zone of solutions and to find the optimal solution graphically.

Graphical Method

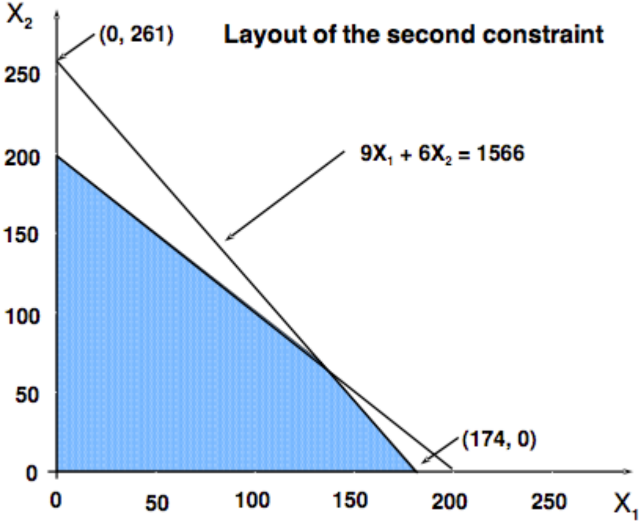
Example:

$$\begin{array}{rl} \max & 350x_1 + 300x_2 \\ \text{subject to} & \\ & x_1 + x_2 \leq 200 \\ & 9x_1 + 6x_2 \leq 1566 \\ & 12x_1 + 16x_2 \leq 2880 \\ & x_1, x_2 \geq 0 \end{array}$$

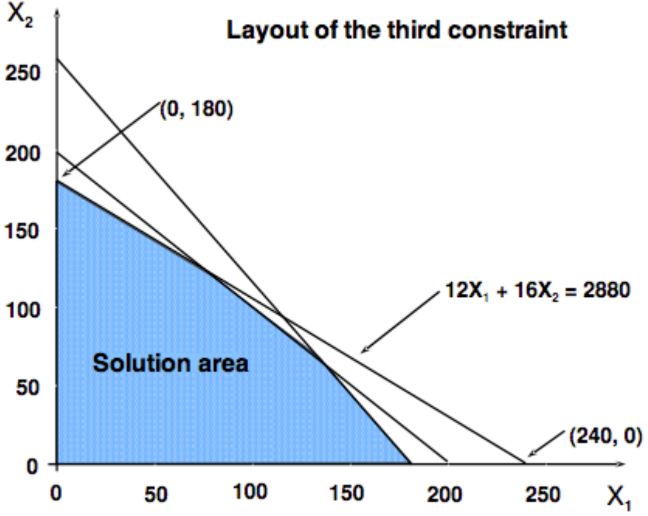
Graphical Method



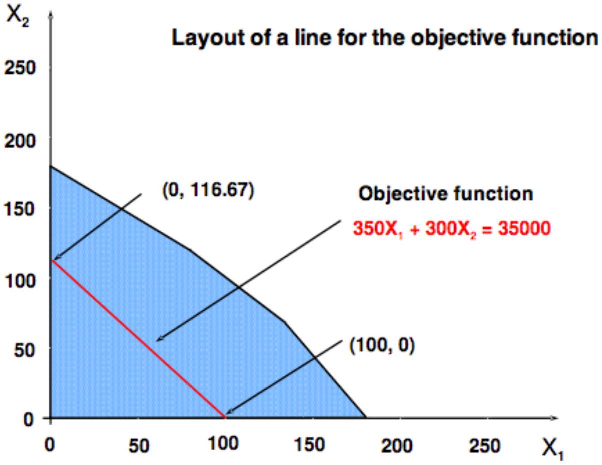
Graphical Method



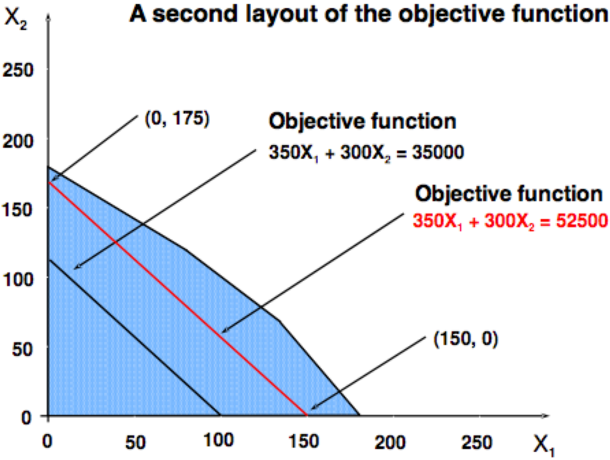
Graphical Method



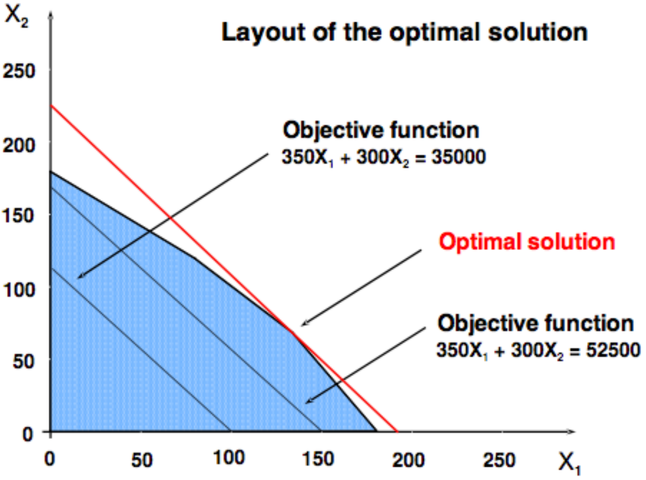
Graphical Method



Graphical Method



Graphical Method



Computation of the optimal solution

The optimal solution is at the intersection of the constraints:

$$x_1 + x_2 = 200$$

$$9x_1 + 6x_2 = 1566$$

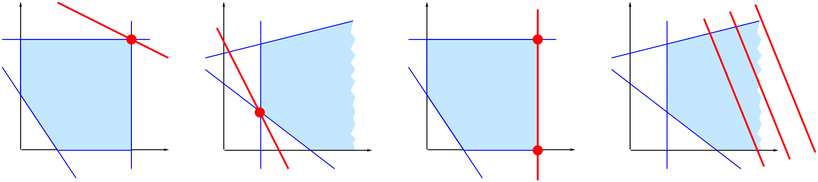
We get:

$$x_1 = 122$$

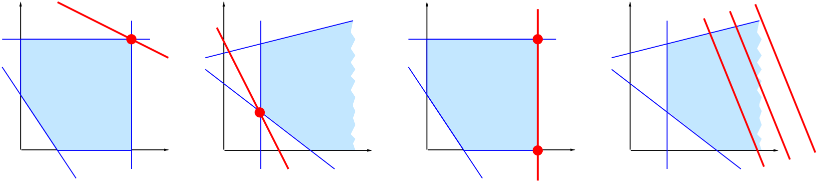
$$x_2 = 78$$

$$\text{Objective} = 66100.$$

Optimal Solutions: Different Cases



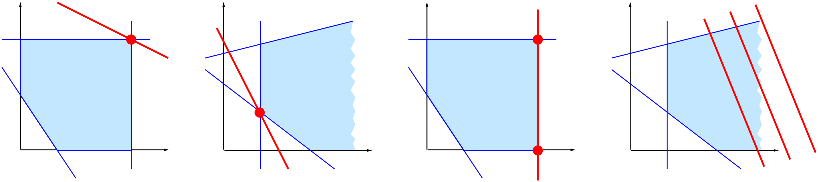
Optimal Solutions: Different Cases



Three different possible cases:

- a single optimal solution,
- an infinite number of optimal solutions, or
- no optimal solutions.

Optimal Solutions: Different Cases

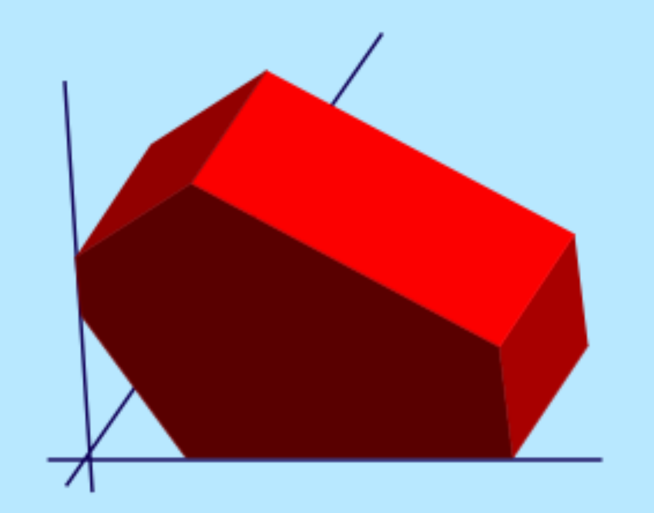


Three different possible cases:

- a single optimal solution,
- an infinite number of optimal solutions, or
- no optimal solutions.

If an optimal solution exists, there is always a corner point optimal solution!

Solving Linear Programmes



Solving Linear Programmes

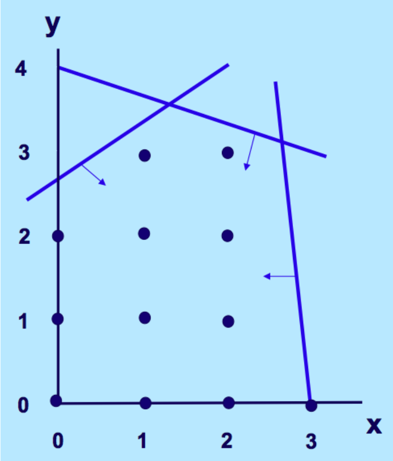
- The constraints of an LP give rise to a geometrical shape: a convex polyhedron.
- If we can determine all the corner points of the polyhedron, then we calculate the objective function at these points and take the best one as our optimal solution.
- The Simplex Method intelligently moves from corner to corner until it can prove that it has found the optimal solution.

Solving Linear Programmes

- Geometric method impossible in higher dimensions
- Algebraical methods:
  - Simplex method (George B. Dantzig 1949): skim through the feasible solution polytope.
    Similar to a "Gaussian elimination".
    Very good in practice, but can take an exponential time.
  - Polynomial methods exist:
    - Leonid Khachiyan 1979: ellipsoid method. But more theoretical than practical.
    - Narendra Karmarkar 1984: a new interior method. Can be used in practice.

But Integer Programming (IP) is different!

- Feasible region: a set of discrete points.
- Corner point solution not assured.
- No "efficient" way to solve an IP.
- Solving it as an LP provides a relaxation and a bound on the solution.



Summary: To be remembered

- What is a linear programme.
- The graphical method of resolution.
- Linear programs can be solved efficiently (polynomial).
- Integer programs are a lot harder (in general no known polynomial algorithms).
  In this case, we look for approximate solutions.