Optimization

Based on the lectures of
Stéphan THOMASSÉ *and Elisa* RICCIETTI
Notes written by Hugo SALOU

ENS DE LYON

October 4, 2025

Contents

1 Linear optimization.

Example 1.1.

- ▷ A *vertex cover* in a graph $G = (V, E)$ is a set of vertices $X \subseteq V$ such that, for every edge $xy \in E$, $x \in X$ or $y \in X$. We will write $\tau(G)$ for the minimum size of a vertex cover of G.
- ▷ A *matching* is a set of disjoint edges. We will denote $\nu(G)$ is the maximum size of a matching.

For the pentagonal graph $G_\triangle$, we have that that $\nu(G_\triangle) = 2$ and $\tau(G_\triangle) = 3$.

Computing a vertex cover of minimum size is **NP**-hard; and, finding a matching of maximal size is very tricky but polynomial (Edmond's theorem). It is obvious that we have $\nu \leq \tau$.

We will consider the *fractional relaxation* of these problems.

Consider a variable x_v for every vertex v and ask that

- ▷ $\forall uv \in E,\ x_u + x_v \geq 1$;
- ▷ $\forall u \in V,\ x_u \geq 0$;

such that $\sum_{v \in V} x_v$ is minimal, which we will write $\tau^*(G)$. This is called the *parameter fractional vertex cover*. We have that $\tau^*(G_\triangle) = \frac{5}{2}$.

For matching, we put a weight y_e for every edge e such that

- ▷ $\forall v \in V,\ \sum_{e \ni v} y_e \leq 1$;
- ▷ $y_e \geq 0$,

such that $\sum_{e\in E} y_e$ is maximized, which we will write ν^*. We have that $\nu^*(G_\triangle) = \frac{5}{2}$.

The fact that $\nu^*(G_\triangle) = \tau^*(G_\triangle)$ is a more general fact:

$$\nu \leq \nu^* = \tau^* \leq \tau.$$

Remark 1.1.

- ▷ The problem of linear programming is in **P**. Linear solver programs can be done in polynomial time. Thus, computing a released solution is possible and useful.
- ▷ *Duality*: Dual fractional parameters are equal, parameters come by pairs.

Remark 1.2 (Why is linear programming tractable?).

- ▷ There is an efficient algorithm called the *simplex algorithm*, but it is not in **P**.
- ▷ There is a polynomial algorithm (using ellipsoids) but it is not useful in practice.
- ▷ There is an algorithm which is both in **P** and efficient using interaction-point methods.

1.1 The simplex algorithm.

Let (P) be the following linear problem:

$$\text{(P)} : \text{maximize } 5x_1 + 4x_2 + 3x_3 \text{ with } \begin{cases} 2x_1 + 3x_2 + x_3 & \leq 5 \\ 4x_1 + 3x_2 + 2x_3 & \leq 11 \\ 3x_1 + 4x_2 + 2x_3 & \leq 8 \\ x_1, x_2, x_3 & \geq 0 \end{cases}.$$

We can try to increase x_1, or x_2, or x_3 but what's the next step? Introduce new variables called *slack variables* x_4, x_5, x_6 (one for each constraint).

We can transform (P) with:

$$(\mathrm{D}_0) : \begin{cases} x_4 = 5 - 2x_1 - 3x_2 - x_3 \\ x_5 = 11 - 4x_1 - x_2 - 2x_3 \\ x_6 = 8 - x_1 - 4x_2 - 2x_3 \\ z = 5x_1 + 4x_2 + 3x_3 \end{cases}.$$

The problem of maximizing the objectif function of (P) if equivalent to maximize z under the constraints of the inequalities transformed into equalities and with $x_1, \dots x_6 \geq 0$.

The problem (D_0) is the *(initial) dictionary.*

To a dictionary, we associate a solution by setting to 0 the non-basic variables x_1, x_2, x_3 and getting solutions for the basic variables x_4, x_5, x_6. In our case, we would have $x_4 = 5$, $x_5 = 11$ and $x_6 = 8$, for an objective of 0.

! If one of these values of the solution (here, $5, 11, 8$), would negative, there would be a problem. We could have an empty domain (this is the problem of solving the decision problem associated to (P)). We will see later how to solve this problem.

We can try by hand to increase z. If we try to increase x_1, we have that the highest limitation for x_1 is $\frac{5}{2}$ (we can see that using $x_4, x_5, x_6 \geq 0$ by solving for x_1 with $x_2 = x_3 = 0$, this constraint is from the one on x_4). We increase it, but... what next?

Now, this is the idea of Dantzig: **Pivot**. We will call x_1 the *entering variable* and x_4 the *leaving variable.* We will then exchange the role of x_1 and x_4 and substitute:

$$(\mathrm{D}_1) : \begin{cases} x_1 = \frac{5}{2} - \frac{x_4}{2} - \frac{3x_2}{2} - \frac{x_3}{2} \\ x_5 = 1 + 2x_4 + 5x_2 \\ x_6 = \frac{1}{2} + \frac{3}{2}x_4 - \frac{x_2}{2} \\ z = \frac{25}{4} - \frac{5x_4}{2} - \frac{7x_2}{2} + \frac{x_3}{2} \end{cases}.$$

We observe that (D_1) is equivalent to (D_0) and (P) when $x_1, \dots, x_6 \geq 0$. We can now iterate the process, choosing a new entering variable.

To increase z, we only have one choice: increasing x_3. To find the leaving variable, we have that $x_3 \leq 1$ with the constraint on x_6. We should then pivot x_3 and x_6:

$$(\mathrm{D}_2)\begin{cases} x_1 = 2 - x_4 - 2x_2 + x_6 \\ x_5 = 1 + 2x_4 + 5x_2 \\ x_3 = 1 + 3x_4 + x_2 - 2x_6 \\ z = 13 - x_4 - 3x_2 - x_6 \end{cases}.$$

Now, we have no choice for an entering variable. This means we are on an ***optimal solution*** and the simplex algorithm stops.

The solutions of (D_2) with $x_1, \dots, x_6 \geq 0$ are equivalent to (P). This means that $z \leq 13$. The solution associated to (D_2) is

$$s_2 = (\underset{x_1}{2}, \underset{x_2}{0}, \underset{x_3}{1}, \underset{x_4}{0}, \underset{x_5}{1}, \underset{x_6}{0}).$$

The optimal for (P) is $(2, 0, 1)$ for an objective of 13.

! The linear problem (D_2) contains the certificate that $\mathrm{OPT} \leq 13$. In $z = 13 - x_4 - 3x_2 - x_6$, the variables x_4 and x_6 are slack and thus correspond to the constraints

- ▷ $x_4 \to 2x_1 + 3x_2 + x_3 \leq 5$ $(\times 1)$;
- ▷ $x_6 \to 3x_1 + 4x_2 + 2x_3 \leq 8$ $(\times 1)$;

thus, the objective is

$$\mathrm{obj} \leq 5x_1 + 7x_2 + 3x_3 \leq 13,$$

which we obtain by summing the two rows. We get back the original objective functions. The certificate of optimality is: a non-negative combination of constraints is larger than the objective function.

Intuition: What are pivots? The simplex moves from every vertex to another (adjacent) vertex of the polyhedron.

How many steps? Consider a polyhedron P with n vertex and m facets. The *skeleton* is the graph obtain from vertices of P and edges of P. An upper bound of the number of pivots is the diameter (*i.e.* the distance between the furthest vertices) of G.

For the cube, we get that the dimension is 3 with a diameter 3 and 6 facets. For the K_4 the complete graph with 4 vertices, we have a dimension of 3, with 4 facets and we have a diameter of 1.

It is conjectured that the diameter is bounded by the number of facets minus the dimension (Hirsch conjecture).

If this conjecture is true, we have that there exists a sequence of pivots with length bounded by $n + m - n = m$.

This is false but polynomial versions are still open.

2 Application of linear programming.

Consider the two player game of Morra:

- ▷ Alice hides one or two coins;
- ▷ Bob hides one or two coins;
- ▷ Alice and Bob announce one or two coins.

Each player will have a pair $(i, j) \in [2] \times [2]$ where i is the hidden number of coins, and j is the guess. This is a zero-sum game:

. . .fill this . . .

Alice wants a probability vector which maximizes the gain for every possible move of Bob. In our example, it means maximizing

$$\min(-2x_2 + 3x_3, 2x_1 + 3x_4, -3x_1 + 4x_4, 3x_2 - 4x_3)$$

under the constraints $x_1 + x_2 + x_3 + x_4 = 1$ and $x_1, x_2, x_3, x_4 \geq 0$. This is not exactly a linear program, but we can easily translate it into one:

$$\text{maximize } y \text{ such that } \begin{cases} -2x_2 + 3x_3 \geq y \\ 2x_1 + 3x_4 \geq y \\ -3x_1 + 4x_4 \geq y \\ 3x_2 - 4x_3 \geq y \\ x_1 + x_2 + x_3 + x_4 = 1 \\ x_1, x_2, x_3, x_4 \geq 0 \end{cases} .$$

We can find an unconventional solution (alternatively, we can use the Simplex algorithm to get the solution). The game is symmetric thus $y = 0$. By multiplying the second by 3 and third inequality by 2, we get that $x_4 = 0$ and $x_1 = 0$. Finally, by $x_1 = 1 - x_3$, we can conclude that

$$x_3 \geq \frac{2}{5} = 0.4 \text{ and } x_1 \leq \frac{3}{7} = 0.\overline{428571}.$$

The optimal strategy for Alice is to chose $t \in [0.4, 0.\overline{428571}]$ and to play $(2, 1)$ with probability t and $(1, 2)$ with probability $1 - t$.

Let us go back to the simplex algorithm.