Optimization

based on the lectures of
Stephan THOMASSÉ

notes written by
Hugo SALOU

ENS Lyon - M1

Part 1. Linear optimization

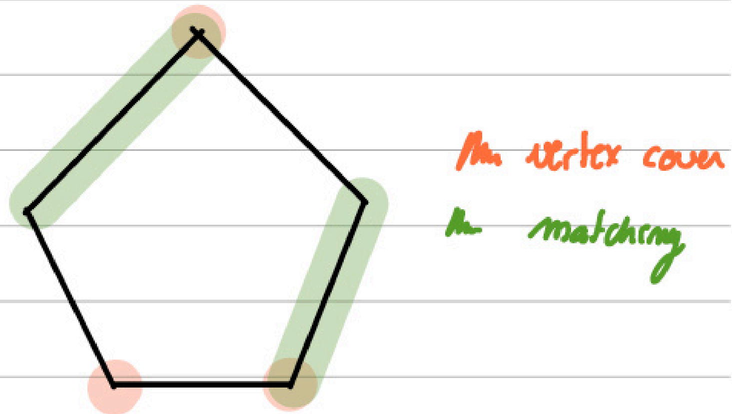
I Modelisation of problems

<u>Example</u>

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 min size of a vertex cover.

For the pentagonal graph:

$$\tau(G) = 3$$
$$\nu(G) = 2$$



A matching is a set of disjoint edges. We will write $\nu(G)$ for the max size of a matching.

Computing a min size vertex cover is NP-hard; computing a max-size matching is very tricky but poly-time (Edmond's thm).

It is obvious that:

$$\nu \leq \tau$$

We will define fractional relaxations of these problems.

Let x_v be a variable for every vertex $v \in V$. We ask that

$$\forall uv \in E, \quad x_u + x_v \geq 1 \quad \text{and} \quad \forall u \in V, \ x_u \geq 0$$

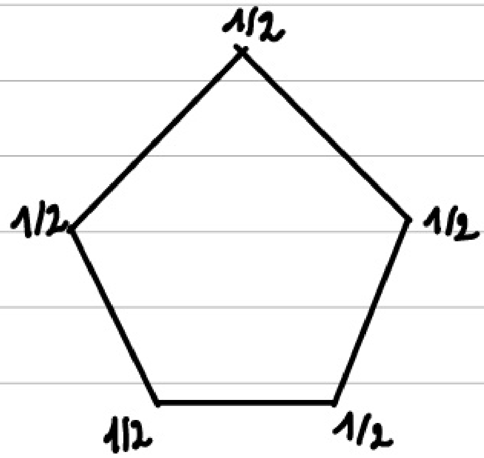
such that $\sum_{u \in V} x_u$ is minimal. We will write τ^* for the min.

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

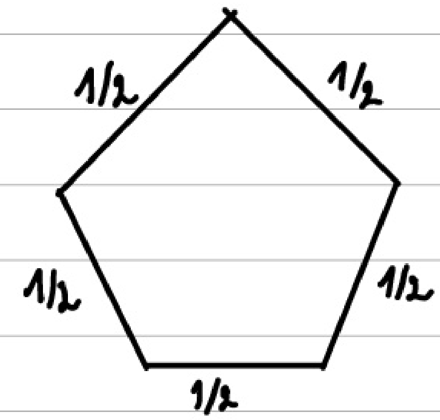
$$\forall e \in E, \quad y_e \geq 0 \qquad \text{and } \forall u \in V, \sum_{e \ni u} y_e \leq 1.$$

We will write ν^* for the max of $\sum_{e \in E} y_e$.

For the pentagon graph, we have :



$\tau^*(G) = \frac{5}{2}$



$\nu^*(G) = \frac{5}{2}$

In general, we have that

$$\nu \leq \underbrace{\nu^* = \tau^*}_{\text{"primal / dual" parameters}} \leq \tau .$$

<u>Remark</u> LP (linear programming) is in P. Linear solver programs can be done in poly-time, thus computing relaxed solutions is possible and useful.

Duality: LP come by pairs and parameters of primal & duals are equal.

Why is LP tractable?

1) The <u>simplex algorithm</u> is efficient but not in P.
2) There is a poly-time algo (using ellipsoids) but not useful in practice.
3) There is an algo that is both efficient and in P using interaction-point methods.

II The Simplex Algorithm.

Consider the following LP:

$$(P): \quad \max 5x_1 + 4x_2 + 3x_3$$
$$\text{s.t.} \left| \begin{array}{l} x_1, x_2, x_3 \geq 0 \\ 2x_1 + 3x_2 + x_3 \leq 5 \\ 4x_1 + 3x_2 + 2x_3 \leq 11 \\ 3x_1 + 4x_2 + 2x_3 \leq 8 \end{array} \right.$$

We can try to increase $x_1, \dots, x_3$ but what's the next step?
Introduce <u>slack variables</u> $x_4, \dots, x_6$ one for each constraint.

We then transform (P) in

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

(P) is equivalent to $\max z$ st. $\underbrace{(\mathcal{D}_0)}_{\text{(initial) dictionnary}}$ & $x_1, \dots, x_6 \geq 0$.

To a dictionnary we associate a solution by setting non-basic variables to 0 and getting solutions for the basic variables.

For $(\mathcal{D}_0)$, its solution is $\underbrace{(5, 11, 8)}$ $= (x_4, x_5, x_6)$ for an objective of 0.

! if one of these would be negative, there would be a problem (eg. empty domain)

We can try by hand to increase z by increasing one and only variable: highest limitation is $x_1 \leq 5/2$ with constraint x_4.

Now... what's next? It's <u>PIVOT</u> time (Dantzig's idea). We call x_1 the leaving var and x_4 the entering var.
We can exchange the role of x_1 and x_4 and get

$$(D_1) : \begin{cases} x_1 = 5/2 - x_4/2 - 3x_2/2 - x_3/2 \\ x_5 = 1 + 2x_4 + 5x_2 \\ x_6 = 1/2 + 3x_4/2 - x_2/2 \\ z = 25/4 - 5x_4/2 - 7x_2/2 + x_3/2 \end{cases}$$

We have that (D_1) is equiv. to (D_0) and thus to (P).

We will now iterate the process, choosing a new entering var.

To increase z, we can only increase x_3, but we are constrained to $x_1 \leq 1$ by x_6's constraint. By pivot, we obtain

$$(D_2) \begin{cases} x_1 = 2 - x_4 - 2x_2 + x_6 \\ x_2 = 1 + 2x_4 + 5x_2 \\ x_3 = 1 + 3x_4 + x_2 - 2x_6 \\ z = \underline{\underline{13}} - x_4 - 3x_2 - x_6 \end{cases}$$

No entering variable! We are sitting on an optimal solution and the simplex algorithm.

This means the optimal for (P) is 13 as $x_1, \dots, x_6 \geq 0$.

! We also have a certificate of optimality.

(P) : $\max\ 5x_1 + 4x_2 + 3x_3$

s.t.
$$\begin{cases} x_1, x_2, x_3 \geq 0 \\ 2x_1 + 3x_2 + x_3 \leq 5 & (1) \\ 4x_1 + 3x_2 + 2x_3 \leq 11 & (2) \\ 3x_1 + 4x_2 + 2x_3 \leq 8 & (3) \end{cases}$$

In (D_2) we have

$$y = 13 - \underline{x_4} - 3x_2 - \underline{x_6}$$

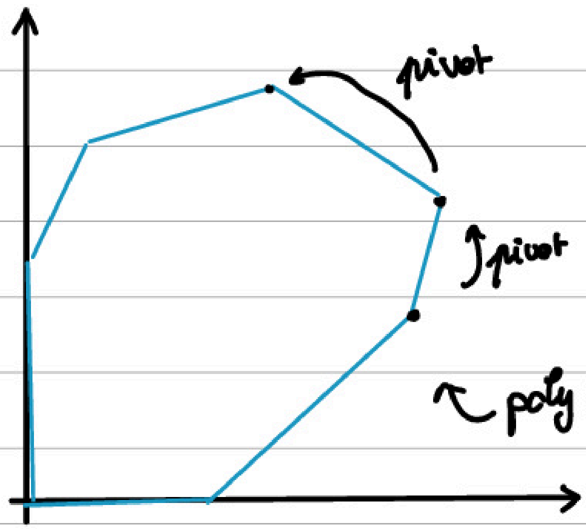
slack

and their coef is $-(1)$.

We sum $1 \times (1)$ and $1 \times (3)$ and get

$$\underbrace{5x_1 + 7x_2 + 3x_3}_{\text{Objective function}} \leq 13$$

Intuition: what are pivots?



← polyhedra of constraints

We move between adjacent vertices of the constraint polyhedra.

How many steps? Consider P with n vertices and m facets.
The <u>skeleton</u> of P is the graph obtained from vertices of P and facets of P.
An upper bound on the number of pivots is diam(skeleton).

For the cube, we have a dim° of 3, with diameter of 3 and 6 facets.
For K_4 the complete 4-vertices-graph, we have a dim° of 3 with 4 facets and a diameter of 1.

It is conjectured that

$$\text{diam} \leq \# \text{facets} - \text{dim}^\circ \qquad \text{(Hirsch's conjecture)}$$

In general, this is false!

III Applications of linear programming

Consider the two-player game of Morra:

1) Alice hides one or two coins;
2) Bob hides one or two coins
3) Alice & 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 announced number.

This is a zero-sum game: the goal is for each player to guess the other's hidden num. of coins. If one of the player guesses correctly, but not the others, the winner wins all the hidden coins.

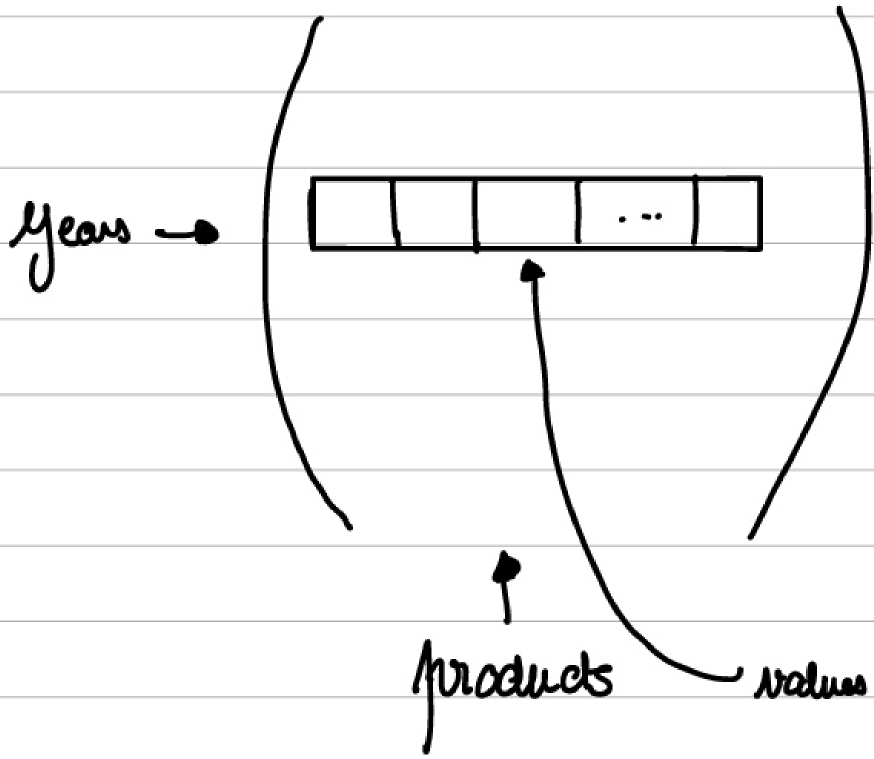
Alice has a pay off matrix:

$$\begin{array}{cc} & \text{Alice} \\ & \begin{array}{cccc} (11) & (12) & (21) & (22) \end{array} \\ \text{Bob} \begin{array}{c} (11) \\ (12) \\ (21) \\ (22) \end{array} & \begin{pmatrix} 0 & -2 & 3 & 0 \\ 2 & 0 & 0 & -3 \\ -3 & 0 & 0 & 4 \\ 0 & 3 & -4 & 0 \end{pmatrix} \end{array}$$

In the general setting, $M = (m_{ij})_{i,j}$ with two players (the row and the column player). Each will choose one col $\hookrightarrow i$ resp. row $\hookrightarrow j$ and recieves m_{ij}.

__Example__ Rock Paper Scisors

__Remark__ We can represent how much you can bet on stock on year $n+1$.



Distribute money on products by assuming that year $n+1$ is a linear combination of the previous ones $\rightsquigarrow$ play safe.

Alice wants a probability vector which maximizes gain for every possible move of Bob. In Morra's game, it means maximizing

$$\min \begin{pmatrix} -2x_2 + 3x_3 \\ 2x_1 + 3x_4 \\ -3x_1 + 4x_4 \\ 3x_2 - 4x_3 \end{pmatrix}$$

$$\text{s.t.} \quad x_1 + x_2 + x_3 + x_4 = 1$$
$$\text{and } x_1, \dots, x_4 \geq 0$$

This is not exactly a LP, but we can easily translate it into one:

$$\max\ y \quad \text{st} \quad \left| \begin{array}{ll} -2x_2 + 3x_3 \geq y & (1) \\ 2x_1 + 3x_4 \geq y & (2) \\ -3x_1 + 4x_4 \geq y & (3) \\ 3x_2 - 4x_3 \geq y & (4) \\ x_1 + x_2 + x_3 + x_4 = 1 & (5) \\ x_1, \dots, x_4 \geq 0 & \end{array} \right.$$

We can find an unconventional solution (alternatively, we can use the Simplex algorithm to get a solution).
The game is symmetric thus $y = 0$. By $3 \times (2) + 2 \times (3)$, we get that $x_4 = 0$ and $x_1 = 0$. Finally, by $x_2 = 1 - x_3$, we can conclude that:

$$x_3 \geq \frac{2}{5} = 0{,}4 \qquad \text{and} \qquad x_2 \leq \frac{3}{7} = 0{,}\overline{428571}.$$

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

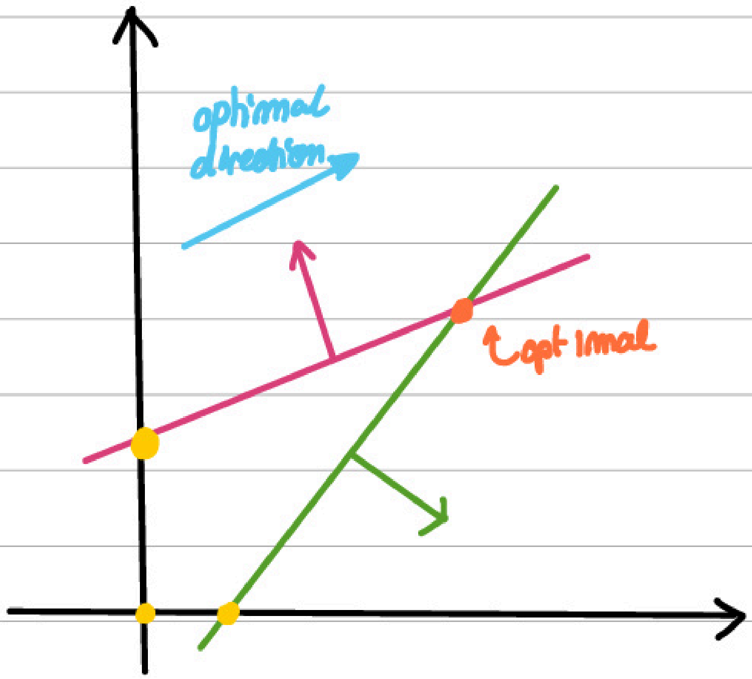
III Visualizing the pivots

Consider (P) : maximize $x_1 + x_2$

such that $x_1 - x_2 \leq 1$ $\quad C_1$

$-x_1 + 2x_2 \leq 2$ $\quad C_2$

$x_1, x_2 \geq 0$



$$(\mathcal{D}_0) \begin{cases} x_3 = 1 - x_1 + x_2 \\ x_4 = 2 + x_1 - x_2 \\ \hline z = x_1 + x_2 \end{cases}$$

Solution associated with $(\mathcal{D}_0)$ is obtained by $x_1 = 0$ and $x_2 = 0$

↳ the solution S_0 is in the intersection of n hyperplanes

x_1 enters

x_3 leaves

$$(\mathcal{D}_1) \begin{cases} x_1 = 1 - x_3 + x_2 \\ x_4 = 3 + x_3 - x_2 \\ \hline z = 1 - x_3 - 2x_2 \end{cases}$$

S_1 is at the intersection of $x_2 = 0$ and $x_3 = 0$, that is, $x_1 + x_4 = 1$.

x_2 enters
x_4 leaves

$$(\mathcal{D}_2) \begin{cases} x_1 = 4 - 2x_3 - x_4 \\ x_2 = 3 - x_3 - x_4 \\ \hline z = 7 - \underbrace{3}_{\geq 0} x_3 - \underbrace{2}_{\geq 0} x_4 \end{cases}$$

S_2 is at the intersection of $x_1 - x_2 = 1$ and $-x_1 + 2x_2 = 2$.

Remark $3 \times C_1 + 2\, C_2$ gives us $x_1 + x_2 \leq 7$.

This is a certificate of optimality

Each pivot can be seen on the polyhedral domain as a move from one vertex v to another vertex v' such that:

- vv' is an edge of the domain

or

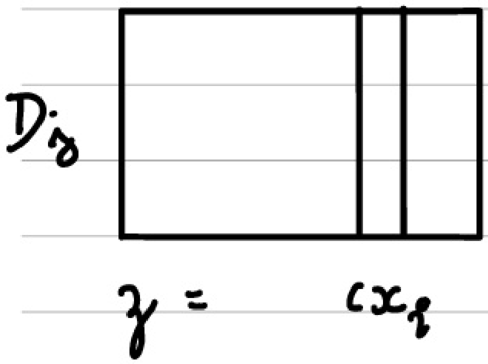
- $v = v'$ degenerate pivot which happens if v is represented by more than one facet of the domain.

V Overview of the Simplex

- Start with an initial dictionnary $\mathcal{D}_0$
  ↳ can be that some constants are < 0
  (c.f next part)

If not, then 0 is a solution and we can thus iterate the pivots.

When in dictionnary D_j there exists $c>0$ with x_j entering.



If all coeficients x_i in D_j are ≥ 0

then the LP is <u>unbounded</u>!

For example,

$$\begin{cases} x_3 = 4 + 2x_1 + x_4 \\ x_1 = 2 \qquad\quad - x_4 \\ \hline z = 6 + x_2 - x_4 \end{cases}$$

x_2 enters but there are no leaving variables

Putting $x_1 := t$, we get a half-line of solution given by

$$\vec{x} = (t, 0, 4+2t, 0)$$

with $z = 6+t$.

The solution is UNBOUNDED!

When there are no entering variables, all coefficients in z are ≤ 0. This means we have found the optimum!

The only question is TERMINATION.

<u>Remark:</u> A dictionnary is defined by the choice of the n possible non basic variables among $n+m$ variables.

(P) in dim° n with m variables has at most $\binom{n+m}{n}$ possible dictionnaries.

If the simplex does not terminate (and choices are deterministic) then it cycles into a sequence of dictionnaries

$$D_1 \to D_2 \to \cdots \to D_t \to D_2 \to \cdots$$

<u>Remark:</u> In such a case, the objective does not increase, thus it stays on the same vertex.

To see that pivot $D_1 \to D_2$ with z that doesn't increase.

$$\text{In } (D_1), \quad \begin{array}{l} \vdots \\ x_i = b_i \quad - \ldots a_{ij} \; x_j \ldots \\ \vdots \end{array} \qquad \text{with } a_{ij} > 0$$

$$z = \qquad \ldots c_j x_j \cdots \qquad \text{with } c_j > 0$$

with x_j is entering and x_i is leaving.

We must have $b_i = 0$ otherwise z augments by

$$\frac{c_j - b_i}{a_{ij}}$$

Thus, in the solution associated to (D_1) we have $x_j = 0$.

In S_2 solution associated to (D_2) we have $x_j = 0$ and also

$$(D_2) \begin{cases} \vdots \\ x_j = 0 + \cdots \\ \vdots \end{cases}$$

Moreover, (D_2) has all non-basic variables of (D_1) (save x_j) thus the solution is the same

$$S_1 = S_2.$$

Avoiding Cycling

We add a very small perturbation to all constraint. Every vertex of the domain is now derived by a unique set of n hyperplanes. To recover the original solution, you use rounding.

→ Do this formally with a sequence of infinitesimal

$$\varepsilon_1 < \varepsilon_2 < \dots < \varepsilon_n.$$

Bland's rule

- Choose every entering variable with lowest index (among the possible candidates).
- Same for leaving

Theorem Simplex does not cycle with Bland's rule.

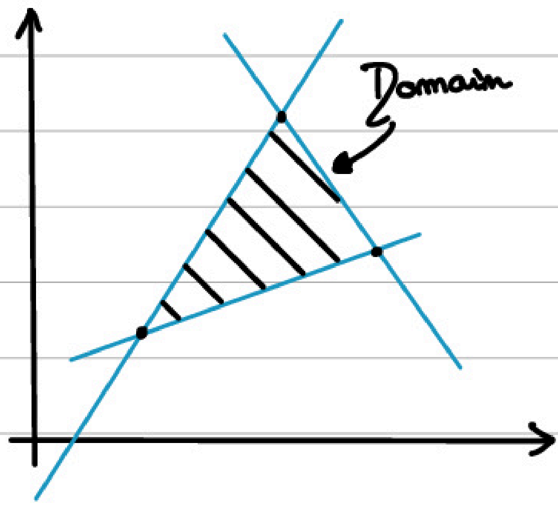
How many steps? We do not even know is a poly(n+m) path exists from one vertex to another.

VI The first phase (Initialization)

How do we start when 0 is not a solution?

$$(P)\ \max\ 2x_1 + x_2$$
$$\text{st} \left| \begin{array}{l} -2x_1 + x_2 \leq -2 \\ x_1 - 2x_2 \leq -2 \\ x_1 + x_2 \leq 7 \\ x_1, x_2 \geq 0 \end{array} \right.$$

☹ 0 is not a solution.
((D_0) contains $x_3 = \boxed{-2} + \dots$)



We have to "jump" on one vertex of the domain.

To do that we just have to solve

$$\text{minimize } x_0 \quad (\text{i.e. maximize } -x_0)$$

such that
$$\begin{cases} -2x_1 + x_2 \leq -2 + x_0 \\ x_1 - 2x_2 \leq -2 + x_0 \\ x_1 + x_2 \leq 7 + x_0 \\ x_1, x_2 \geqslant 0 \end{cases}$$

The initial dictionnary of this other LP is

still < 0

$$(D_0') \left\{ \begin{array}{l} x_3 = \boxed{-2} + 2x_1 - x_2 + x_0 \\ x_4 = \boxed{-2} - x_1 + 2x_2 + x_0 \\ x_5 = \boxed{7} - x_1 - x_2 + x_0 \\ \hline w = x_0 \end{array} \right.$$

Now we do an <u>illegal pivot!</u>

x_0 will enter and leaving is the one with min value, for instance x_3.

All values are ≥ 0 :)

$$(D_1') \left\{ \begin{array}{l} x_0 = \boxed{2} - 2x_1 + x_2 + x_3 \\ x_4 = \boxed{0} - 3x_1 + 3x_2 + x_3 \\ x_5 = \boxed{9} - 3x_1 + x_3 \\ \hline w = -2 + 2x_1 - x_2 - x_3 \end{array} \right.$$

$\wr$

Iterates pivots

$$(D_3')\begin{cases} x_2 = 2 + 2x_4/3 - x_0 + x_3/3 \\ x_1 = 2 + x_4/3 - x_0 + 2x_3/3 \\ x_5 = 3 - x_4 + 3x_0 - x_3 \\ \hline w = -x_0 \end{cases}$$

$\boxed{\text{If the optimum } w \text{ is } < 0 \text{ then the domain is empty.}}$

To solve (P) we use the initial dictionnary:

$$(D_0) : \left|\begin{array}{l} x_2 = 2 + 2x_4/3 \cancel{-x_0} + x_3/3 \\ x_1 = 2 + x_4/3 \cancel{-x_0} + 2x_3/3 \\ x_5 = 3 - x_4 + \cancel{3x_0} - x_3 \\ \hline \eta = \underbrace{6 + 4x_4/3 + 5x_3/3} \end{array}\right.$$

from (D_3') without the x_0's

$$\begin{aligned} \eta &= 2x_1 + x_2 \\ &= 4 + 2x_4/3 + 4x_3/3 \\ &\quad + 2 + 2x_4/3 + x_3/3 \\ &= 6 + 4x_4/3 + 5x_3/3 \end{aligned}$$

<u>Homework</u>: Code the Simplex algorithm

VII Duality

The goal is to certify the optimality of a solution.

Example: maximize $z := 4x_1 + x_2 + 5x_3 + 3x_4$
under the constraints

(P) primal

$$
\begin{aligned}
x_1 - x_2 - x_3 + 3x_4 &\leq 1 \quad \Big) \times y_1 \geq 0 \\
5x_1 + x_2 + 3x_3 + 8x_4 &\leq 55 \quad \Big) \times y_2 \geq 0 \\
-x_1 + 2x_2 + 3x_3 - 5x_4 &\leq 3 \quad \Big) \times y_3 \geq 0 \\
x_1, \dots, x_4 &\geq 0
\end{aligned}
$$

OPT: $(0, 14, 0, 5)$

How to find an upper bound on (P)?

↳ Make linear combinations of the constraints (with non-negative coefficients, $y_1, \dots, y_n$) in such a way that the left hand terms "majorates" the objective function. Then,

$$y_1 + 55y_2 + 3y_3$$

will be an upper bound!

The best bound one can derive is a solution of the following DUAL linear problem:

(D) "dual" minimize $y_1 + 55y_2 + 3y_3$

with the constraints

$$\begin{cases} y_1 + 5y_2 - y_3 \geq 4 \\ -y_1 + y_2 + 2y_3 \geq 1 \\ -y_1 + 3y_2 + 3y_3 \geq 5 \\ 3y_1 + 8y_2 - 5y_3 \geq 3 \\ y_1, y_2, y_3 \geq 0 \end{cases}$$

<u>OPT</u>: $(11, 0, 6)$

<u>Lemma</u> (Weak Duality Theorem)

The value of a solution of (D) is always at least the value of any solution of (P).

<u>Proof</u> A LP (P) reads

(P) maximize $c^T x$ such that $\begin{cases} Ax \leq b \\ x \geq 0 \end{cases}$

and the dual is

(D) minimize $b^T y$ such that $\begin{cases} A^T y \geq c \\ y \geq 0 \end{cases}$

Assuming that x is a solution of (P) and y

a solution of (D). we have :

$$\text{value of } (P) = c^T x \leq (y^T A)\, x = y^T (A x) \leq y^T b = \underset{(D)}{\text{val of}}$$

$\square$

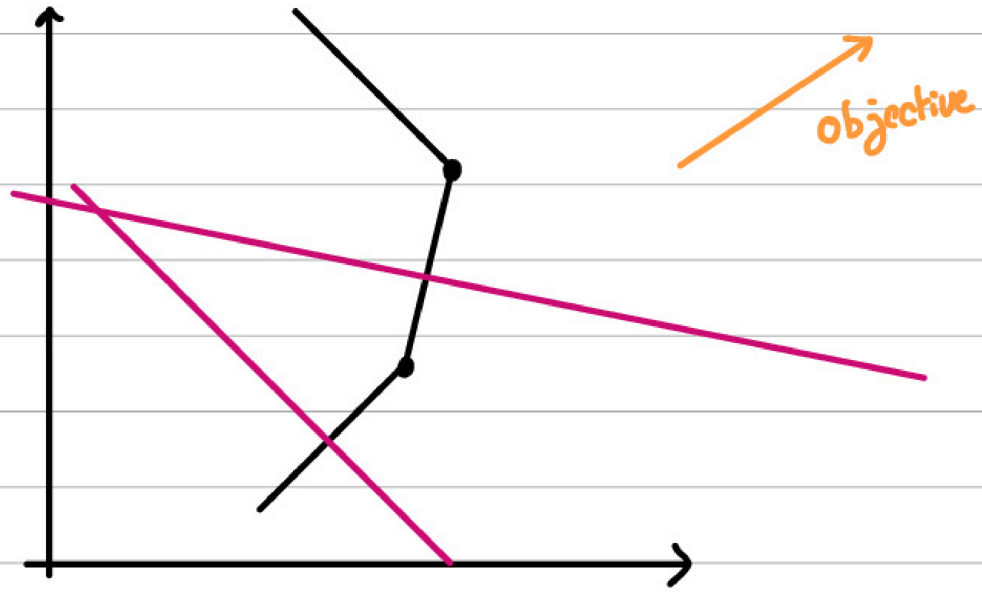
<u>Theorem</u> (Gale, Khan, Tucker in '51).

If (P) has an optimal solution then (D) has an optimal solution with the same value.

Many proofs ; one can be made on the simplex and the fact that, if (P) has a solution, then the final dictionnary provides the linear combinations.

A vague idea is the following. When the simplex steps, the objective function in the positive cone of the (normal) vectors corresponding to the hyperplanes defined by the non-basic variables.

The coefs (i.e. the dual solution) can be read in the last dictionnary.



Consequences & Remarks

① Can always easily certify optimality

② "Decision is optimization"

Decision LP

Input : Given a set of inequalities $Ax \le b$

Output : TRUE if there exists a solution x (return x)
FALSE otherwise

Remark that if <u>DECISION LP</u> is in $\mathbb{P}$ then so is <u>Solving LP</u>

We want to solve $\max c^T x$ st $Ax \le b$ and $x \ge 0$

Form a decision version on variables (x, y) where y are the variables of the dual, and ask for a point in the polyhedron:

$$\text{Domain: } \left| \begin{array}{l} Ax \leq b \\ A^T y \leq c \\ c^T x = b^T y \end{array} \right.$$

Any solution (x, y) of this domain is an optimal solution of (P) and an optimal solution of (D).

All research to find an algorithm to solve LP points in the direction of solving Decision LP.

③ Certificate for Decision

A system of equalities $Ax = b$ has no solution IFF A linear combination of these equalities give $0 = 1$.

$\uparrow$

Gauss

A system of inequalities $Ax \le b$ has no solution IFF (Strong duality) A non-negative combination of these inequalitites give $0 \le -1$.

A system of polynomial (multivariable) $P_1(x_1 \dots x_n)$ $\vdots$ $P_m(x_1 \dots x_n)$ has a common zero IFF (Hilbert's Nullstellensatz) there exists multipliers $Q_1, \dots, Q_m$ (polynomials) such that $\sum Q_i P_i = 1$

It is very to code 3SAT with polynomials but the size of the Q_i's are exponential.

Remark

- The dual of the dual is the primal.
- If a variable x_i in the primal is not constrained to be non-negative, it gives rise to an equality in the dual.

Conversely, if

$$(P) \quad \max \dots \quad st \quad \dots \quad a_i x = b \dots$$

then the y_i's are not constrained to be ≥ 0.

VIII Two examples of duality

1) Matching is the dual of vertex cover

Given $G = (V, E)$ and I the incidence matrix of G

$I := (I_{v,e})_{v \in V, e \in E}$ with $I_{v,e} = 1$ iff $v \in e$.

The (fractional) Minimum vertex cover is

$$\min \mathbb{1}^T x \quad \text{s.t.} \quad Ix \geq 1 \quad x \geq 0$$

The dual is maximize $\mathbb{1}^T y$ s.t. $I^T y \leq 1$, $y \geq 0$.

Maximizing of weights on edges s.t. no vertex recieves total weight more than 1 on its incident edges

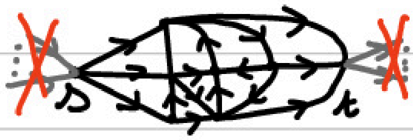
$\longrightarrow$ max fractionnal matching

2) Duality for max flow (bad case)

s is the source and t terminal

Flow is weight x_{uv} on each arc uv.

The relaxed max flow problem is



$$\text{maximize} \sum_{uv \text{ arc}} x_{uv}$$

$$\text{subject to } \forall v \notin \{s,t\},\ \sum_{uv} x_{uv} - \sum_{vw} x_{vw} = 0 \qquad (p_v)$$

$$\forall uv \text{ arc},\quad 0 \leq x_{uv} \leq c_{uv} \qquad (y_{uv})$$

Dualize two types of variables:

→ p_v's : unconstrained "potential"

→ $y_{uv} \geq 0$

The dual of flow is

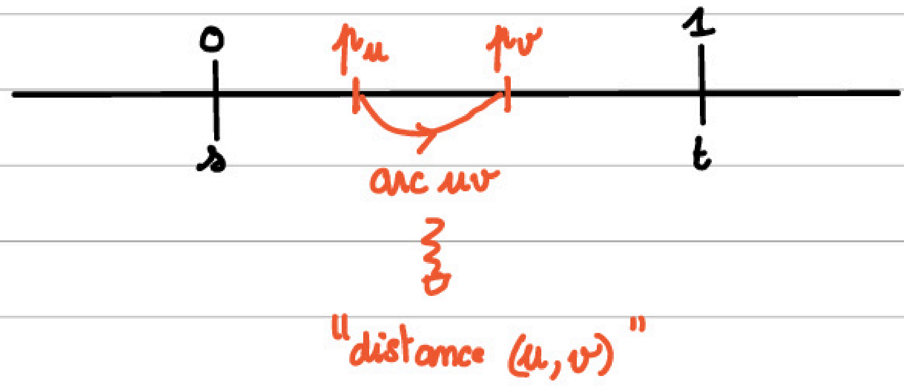
$$\text{minimize} \sum_{uv \text{ arc}} c_{uv} y_{uv} \quad \text{such that} \quad \left|\begin{array}{ll} y_{uv} - p_v + p_u \geq 0 & \forall uv \ \ {\scriptstyle u \neq s,\, v \neq t} \\ y_{ut} + p_u \geq 1 & \forall ut \\ y_{sv} - p_v \geq 0 & \forall sv \\ y_{uv} \geq 0 & \forall uv \\ p_v \text{ is not constrained} & \end{array}\right.$$

This corresponds to (by setting $p_t = 1$ and $p_s = 0$)

$$y_{uv} \geq p_v - p_u$$

The dual of flow involves finding a function p_v for every $v \neq s,t$ where $p_s = 0$ and $p_t = 1$

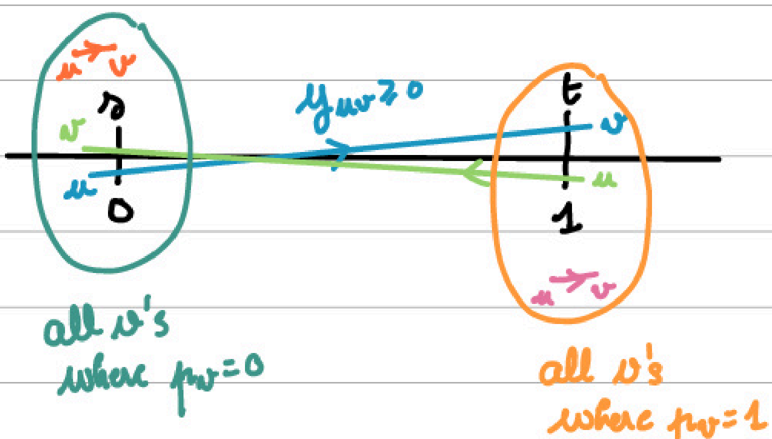
p is a potential



The cost of uv is $c_{uv}\, y_{uv}$ where $y_{uv} \geq \underbrace{p_u - p_v}_{\text{"distance"}}$

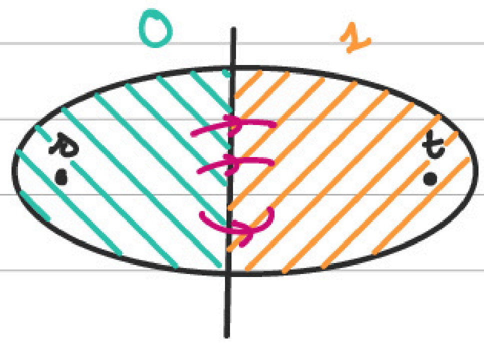
Unfortunately, backward arcs correspond to $y_{uv} = 0$.

<u>Integer solutions</u> involve setting p_v to 0 or 1



arcs in the same "group" have $y_{uv} = 0$

This corresponds exactly to the minimum st-cut problem



The value of the function is $\sum\limits_{\substack{p_u=0 \\ p_v=1}} c_{uv}$

IX Concrete interpretation of dual variables

Given raw materials A, B, C needed to produce products x_1, x_2, x_3. The respective compositions are

$$\begin{array}{c|ccc|} x_1 & 1 & 2 & 3 \\ x_2 & 2 & 3 & 1 \\ x_3 & 3 & 2 & 1 \\ \hline & A & B & C \end{array}$$

In our stock we have the following amount:

A is 5
B is 4
C is 6

We can sell x_1 for 1
x_2 for 2
x_3 for 2

(and we admit rational solutions)

(P) maximize $x_1 + 2x_2 + 2x_3$
such that

$$\left|\begin{array}{l} x_1 + 2x_2 + 3x_3 \leq 5 \\ 2x_1 + 3x_2 + 2x_3 \leq 4 \\ 3x_1 + x_2 + x_3 \leq 6 \\ x_1, x_2, x_3 \geq 0 \end{array}\right.$$

OPT: $\left(0, \frac{5}{2}, \frac{7}{2}\right)$ for a value of $\frac{18}{5}$

(D) minimize $5y_1 + 4y_2 + 6y_3$
such that

$$\left|\begin{array}{l} y_1 + 2y_2 + 3y_3 \geq 1 \\ 2y_1 + 3y_2 + y_3 \geq 2 \\ 3y_1 + 2y_2 + y_3 \geq 2 \\ y_1, y_2, y_3 \geq 0 \end{array}\right.$$

OPT: $\left(\frac{2}{5}, \frac{2}{5}, 0\right)$ for a value of $\frac{18}{5}$

__Interpretation__ The value of the dual correspond to the cost of raw materials from your point of view (at optimum of (P)).

Suppose the value of A in the market is 0.5. Then, maybe it is better to sell $\varepsilon > 0$ amount of A. ⟿ (P_ε).

In (P_ε), the first constraint is now

$$x_1 + 2x_2 + 3x_3 \leq 5 - \varepsilon$$

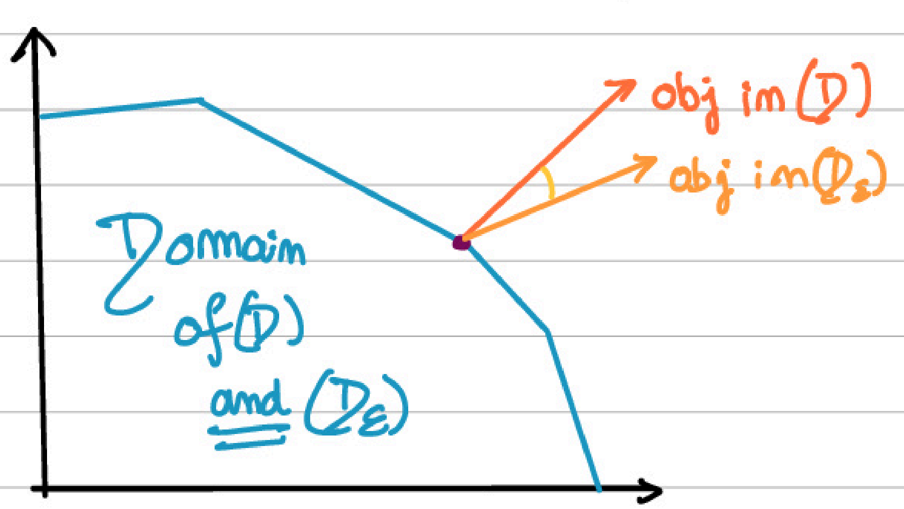
☹ It is very hard to see how (P) evolves since the domain is changing.

__From the point of view of (D):__

(D_ε) minimize $(5-\varepsilon)y_1 + 4y_2 + 3y_3$
such that

EXACTLY THE SAME CONSTRAINTS AS (D)

The domain is fixed but only the objective function is tilted by ε.



Under the (natural) hypothesis that the optimal solution of (D_ε) does not change, the value of (D) is

$$0.4 \times (5-\varepsilon) + 0.4 \times 4 +$$

Thus $obj(D_\varepsilon) - obj(D)$ is -0.4ε

Since the ε part is sold for 0.5ε we gain 0.1ε.

How large do we choose ε without the optimal solution of (D) changing? We will use Complementary Slackness.