

### Université catholique de Louvain

# LINMA2471 Optimization models and methods II

### Course notes

#### Students:

Nicolas Boutet, Benjamin Chiem, Antoine de Comité, Alexandre de Touzalin, Céline Gérard, Andine Havelange, Florimond Houssiau, Sébastien Lagae & Vincent Schellekens

## **Contents**

I.	Notes	1
Α.	Linear and Convex modeling	2
	Linear and convex models	. 2
	a. Definitions and motivation	. 2
	b. Standard forms	. :
	i. Linear case: re-writing objective function, variables and constraints	
	ii. Standard form for linear models	
	iii. Standard form for convex models	
	iv. Transforming any problem into a convex problem	
	v. Approximate <i>any</i> convex problem by a linear problem	
	c. Modelling Tricks	
	i. Monotonicity	
	ii. Change of variables	
	iii. Misleading/Deceptive appearances	
	iv. Flexibility	
	v. Charnes and Cooper	
	d. Convex Optimization: Theorems and properties	
	i. Convex sets	
	ii. Convex functions	
	iii. Convexity and linear transformations	
в.	First-order methods	18
С.	Conic modeling and duality	19
D.	Interior-point methods	20
тт	Tala	0.1
II.	Labs	21
	. Introduction : AMPL	. 22
TTT	Exercices	2.

Part I.

Notes

### A. Linear and Convex modeling

#### 1. Linear and convex models

In this section we will see some important classes of optimization problems, and some techniques to re-write an optimization problem into another form.

#### a. Definitions and motivation

First of all, let's recall what the model of an optimization problem looks like.

**Definition A.1.** A general model has the following form:

$$\min_{x \in X \subseteq \mathbb{R}^n} f(x) \tag{A.1}$$

where x are the **variables**, X is the **feasible set** (also called domain or feasible region) and f is the **objective function**.

Note that the feasible set X is a subset of a *finite* dimensional space. Optimization within infinite-dimensional spaces are not covered in this course.

Let us next introduce two very important classes of models: the *linear* and *convex* models.

#### **Definition A.2.** A model is called a linear model if:

- 1. The objective function is linear/affine<sup>1</sup>, that is, of the form  $c^Tx/c^Tx + d$ .
- 2. The feasible set X is a **polyhedron**. A polyhedron is an intersection of a finite<sup>2</sup> number of closed **half-spaces**. As a reminder, a half-space is the set of points that lie on one side of a hyperplane; in an algebraic form :  $\{x \in \mathbb{R}^n | a^T x \ge b\}$  or  $\{x \in \mathbb{R}^n | a^T x \le b\}$ .

#### **Definition A.3.** A model is called a **convex model** if:

- 1. The objective function f is convex (see below).
- 2. The feasible set X is convex (see below).

We still need to define what are convex functions and sets:

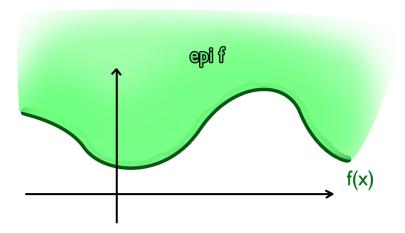
<sup>&</sup>lt;sup>1</sup>Note that the independent term of an affine function (d) can be easily dropped out, because it doesn't affect the optimal solution in any way. Therefore, every affine function can be replaced with a purely linear one

 $<sup>^2</sup>$ For example, a sphere is therefore not a polyhedron, because it is an intersection of an infinite number of closed half-planes.

**Definition A.4.** A set X is a **convex set** if it contains the segments between every pair of its points.

**Definition A.5.** A function f is a **convex function** if its **epigraph** is convex. The epigraph<sup>3</sup> of f is the set of points above (and including) the graph of f. Formally, we write this as: epi  $f := \{(x,t)|t \ge f(x)\}$ . This notion is illustrated on Figure A.1.

For the definition of convex functions, we didn't used the concept of derivative, because we want our definition to be as general as possible. In other words, a non-differentiable function can be convex<sup>4</sup>. Note also that every linear model is a particular case of a convex model.



**Figure A.1.:** Illustration of the epigraph of some one-dimensional (and non-convex) function. The epigraph of f is the region above the graph of f.

Now that we know the definitions of linear and convex models, what is the **motivation** to study such models? First, these models are useful to develop *efficient algorithms* with *guarantees* about the exactitude of the optimal solution and the speed of the algorithm.

Also, one could argue that studying linear/convex models is a restriction to the number of problems we will be able to solve, but it turns out that convex problems are not so rare in practice. Many problems are, or can be formulated as convex problems. Some problems can even be solved by using an equivalent convex problem: for example, the branch and bound algorithm transforms a discrete (and therefore non-convex) problem into a sequence of linear problems. One last -informal- reason to restrict ourselves to convex problem is that, for non-convex problems, there is nothing interesting we can really do or say.

#### b. Standard forms

As the general formulation of convex and linear problems can be very hard to use in order to develop a theory about them, due (mostly) to the variety of constraint types, it is important to define **standard forms**. Standard forms define a unique, specific, formulation of these problems, that is much simpler than the general form.

<sup>&</sup>lt;sup>3</sup>Note that if  $f: \mathbb{R}^n \to \mathbb{R}$  then  $epi \ f \subset \mathbb{R}^{n+1}$ 

<sup>&</sup>lt;sup>4</sup>For example, the norm function defined by  $f: \mathbb{R} \to \mathbb{R}: x \to |x|$  is convex

#### i. Linear case: re-writing objective function, variables and constraints

Before we define the standard form, let us observe a number of transformations that can be applied to a linear problem without changing its solution (that is, the two problems will be equivalent).

**Maximization and minimization** In general, a maximization problem can easily be formulated as a minimization problem. Indeed, maximizing a function f is equivalent to minimizing its opposite -f. If the solution x is the same, the value of the objective function -f(x) is simply the opposite of that of the original problem.

$$\underbrace{\mathbf{Max}}^{\mathsf{max}} f = - \min (-f) \underbrace{\mathbf{Min}}$$

**Non-negative variables** The variables used in the linear standard form are non-negative, that is, they are free with the implicit constraint  $x \ge 0$ . It is possible to transform free variables in non-negative variables by a process known as **splitting**.

$$oxed{\mathbf{X} \ \mathbf{non\text{-}negative}} oxed{\mathsf{splitting}} oxed{\mathbf{X} \ \mathbf{free}}$$

Splitting is done by, for every free variable  $x_i$ , adding two non-negative variables  $x_i^+$  and  $x_i^-$ , defined by the relationship  $x_i = x_i^+ - x_i^-$ . Then,  $x_i$  is substituted by  $x_i^+ - x_i^-$  everywhere in the problem.

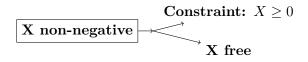
However, this can be severely inefficient, since it adds a variable for every free variable. In a problem with n free variables, n additional variables are created, thus doubling the size of the problem. As this can be a major performance issue, it is important to do the splitting without creating too many variables.

A simple observation about the current method that can be made is that, if we assume  $x_i$  has a unique value in the solution, the values of  $x_i^+$  and  $x_i^-$  are not uniquely defined (in some sense, there is one too many degree of freedom). By exploiting this notion, another formulation can be proposed: if a problem has n free variables  $x_i$ , we subtitute these by  $x_i^+ - x^-$ , where  $x_i^+$  and  $x^-$  are non-negative variable, and  $x^-$  is the same for all the variables. This method is better since it only creates one additional variable!

Why does this work? If and whenever a solution  $x^*$  is found, the value of  $x^{-,*}$  will be at most equal to the smaller (i.e. most negative)  $x_i^*$ . Then, by definition,  $x_i^{+,*} = x_i^* + x^{-,*}$  (with  $x^{-,*} \geq 0$ ), and since  $x_i^* \geq -x^{-,*}$ , we get that  $x_i^{+,*} \geq 0$ , and so  $x_i^{+,*}$  is indeed non-negative.

As an example, suppose a linear problem with 3 variables, all of them free, has as unique solution  $(x_1^*, x_2^*, x_3^*) = (-3, 7, -10)$ . A solution, in term of non-negative variables, is thus  $(x_1^{+,*}, x_2^{+,*}, x_3^{+,*}, x_3^{+,*}, x_3^{-,*}) = (7, 17, 0, -10)$ .

The reverse is also possible, although not really interesting.



**Equalities and inequalities** After treating the objective and the variables, it is important to treat the constraints (which offer the widest range of varieties). Firstly, turning equalities into inequalities is very straightforward:

Equality: 
$$f = g$$
 Two inequalities:  $f \ge g$  and  $f \le g$ 

The opposite is both more useful and more subtle:

Equality: 
$$f + s = g$$
 and slack variable  $s$  Slack Variable Inequality  $f \ge g$ 

To do this, we must introduce the concept of **slack variable** (variable d'écart in French). A slack variable is a non-negative variable added on the greater side of the inequality to make it an equality. Basically, its value is f - g, the slack between f and g, representing the margin before the constraint becomes an equality.

One last case to be treated is the **lower bound constraint**. This can of course be treated as a constraint, but is redundant with the nonnegativity of the variables in the standard form. A fairly simple solution is to subtitute the variable X with X - l, where l is the lower bound.

Lower Bound 
$$X \ge l$$
 Substitution non-negative  $X'$ :  $X = l + X'$ 

#### ii. Standard form for linear models

The standard form of a linear optimization problem is:

$$\begin{aligned} \min_{X} & c^{T}X \\ & AX = b \\ & X \geq 0 \quad (\Longleftrightarrow X \text{non-negative}) \end{aligned}$$

If the problem has n variables and m constraints, then  $c \in \mathbb{R}^{n \times 1}$ ,  $A \in \mathbb{R}^{m \times n}$  and  $b \in \mathbb{R}^{m \times 1}$ . These constitute the only data needed to uniquely define the problem.

The transformations exposed in the previous subsection illustrate the fact that it is possible to turn any linear optimization problem in the standard form.

#### iii. Standard form for convex models

There is **no known standard form** for convex optimization problems. This can be understood in the sense that the objective function can be very general, as well as the set X. Since, by definition, the set X is uncountable, it is very hard to represent it in a way that can be, for example, treated by a computer. However, it is possible, under certain assumptions, to represent the set X in a purely functional way, as a set of inequalities involving functions.

Most often, the set X is defined as a set of constraints. Let's suppose that there exists a

set of functions  $g_i$ ,  $i \in \{1 \ge \dots \ge m\}$ ,  $h_j$ ,  $j \in \{1 \ge \dots \ge l\}$ , such that:

$$X = \{x \in \mathbb{R}^n | g_1(x) \le 0, \dots, g_m(x) \le 0 \text{ and } h_1(x) = 0, \dots, h_l(x) = 0\}$$

This form is general, since the  $g_i(x) \geq 0$  constraint is equivalent to  $-g_i(x) \leq 0$ . The one exception lies in the fact that **strict inequalities** cannot be treated. But, as is explained later, this is not a real issue.

In general, this does not suffice to guarantee that X is convex. The following conditions are **sufficient** to ensure the convexity of the set X:

- the  $g_i$  functions are **convex** (this results from the choice that  $g_i$  should be smaller than zero: if the opposite were chosen, then the functions should be concave)
- the  $h_j$  functions are **linear**: it's tempting to say that convexity is enough, but it is not the case. For example, the function  $h: \mathbb{R}^2 \to \mathbb{R}: (x,y) \to x^2 + y^2 1$  defined as ensemble X the circle (and not the discus!) of radius 1, which is obviously not convex<sup>5</sup>.

In some cases, the constraints  $h_j(x) = 0$  can be relaxed to inequalities  $h_j(x) = 0$  (Shouldn't it be an inequality?), thus relaxing the linearity constraint on  $h_j$ . One of these cases, which will be developed in the next subsection, is when the objective function is linear.

A note on  $\neq$  The standard forms developed in this subsection do not allow for strict inequalities to be considered. This is because strict inequalities tend to make solutions disappear: in linear optimization, solutions are always located on the boundary of a closed polyhedron. By making this polyhedron open (with strict inequalities), the solutions disappear, as there is not admissible point with a minimal value (it is always possible to get *closer* to the boundary, thus reducing the objective function).

A solution to this is to treat strict inequalities as non-strict ones by introducing a tolerance  $\epsilon > 0$ , that describes how close to the open boundary the solutions can lie. The constraint f > g then becomes  $f \ge g + \epsilon$ . The choice of the tolerance depends on the context of the problem (and is to be discussed with the client, for example).

#### iv. Transforming any problem into a convex problem

We can turn any optimization problem into a convex problem by following two "easy" steps. First, the objective function can be made linear by adding a new variable. Then, the constraints can be made convex by an operation called taking the convex hull. Let us see these operations in detail.

**First step : making the objective function linear** Let us assume the following (general) model :

$$\min_{x \in X \subseteq \mathbb{R}^n} f(x)$$

<sup>&</sup>lt;sup>5</sup>In fact, each segment binding two points of the circle doesn't include any other point of the circle that its extremity...

The trick is to re-write this problem introducing a new variable t that is greater or equal<sup>6</sup> to f(x). We now have :

$$\min_{x\in\mathbb{R}^n,\;t\in\mathbb{R}}\;t\;\;\text{with}\;\;x\in X,\;(x,t)\in epi\;f$$

The new objective function is just t and is clearly linear, but the domain is now more complicated. In other words, we have traded simplicity in the objective function (wich is a good thing) by adding complexity in the domain (wich is usually already complex anyway, so it isn't that bad). Note that if the original problem was convex, the new problem is still convex (because it means  $epi\ f$  is convex).

**Second step:** making the constraints convex The feasible set can be made convex by taking the convex hull of the set.

**Definition A.6.** The **convex hull** of a set X (denoted by conv X) is the **smallest** convex set containing X. The smallest set means : the intersection of all possible convex sets containing X.

**Example A.1.** If we have a simple (non-convex) set containing two points in space, the convex hull of this set is the segment between those two points. This example is illustrated figure A.2. This gives us an (infeasible in practice, see "the catch" to make every problem convex, page 8) algorithm to take the convex hull of any set: just take every pair of points in the set and add the segment between those two points!

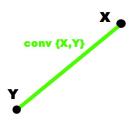


Figure A.2.: Illustration of the convex hull of a set X. Note that the convex hull also includes the original set, wich is not very well represented on the figure.

Taking the convex hull of the feasible set X obviously gives us a new optimization problem. What are the optimal solutions of this new problem?

**Theorem A.1.** Any optimal solution  $x^*$  to the original problem :

$$\min_{x \in X} c^T x$$

is also optimal for the new problem :

$$\min_{x \in conv} c^T x$$

<sup>&</sup>lt;sup>6</sup>It is intuitively more appealing to impose that t = f(x), so we still have "exactly the same" objective function. But since equality constraints are harder to handle, we prefer the inequality  $t \ge f(x)$ . So instead of minimizing f(x), we minimize a higher bound to f(x), wich is equivalent.

This also means that the *optimal values* of the two problems are the same. But because  $X \subseteq conv X$ , some optimal solutions of the new problem won't be in the original set X. So, in order to find the original optimal solutions, once we have solved the convex problem, we should always reject the optimal solutions of the new problems that aren't in X. Mathematically speaking:

$$\{x_{original\ problem}^*\} = \{x_{new\ convex\ problem}^*\} \cap X$$

So, to summarize, we can make every problem in the world convex, and we can (not yet, but after finishing this course) solve convex problems with good algorithms! This implies that basically any optimization problem can be solved easily! It seems too good to be true, and it is, since there is a **catch**. Although the definition of complex hull is rather simple, taking the convex hull of a general (that is, a little sophisticated) set is a very difficult operation.

So, in general, this approach is useless. There are specific cases, however, where this can be very useful!

The first, is a special case of linear optimization with **or** constraints. In such a problem, the admissible set is the union of (possibly disjoint) polyhedra. Computing the convex hull of the union of polyhedra can be done very efficiently, if the vertices are known, since the only task is to compute the vertices that will stay extreme in the union. The figure A.3 illustrates this example.

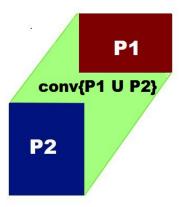


Figure A.3.: Two (simple) disjoint polyhedron and the convex hull (in green) of the union.

An interesting point to be raised here, is that such a problem could also be solved by computing the solution on every polyhedron, then choosing the best one. For small problems, this is of course valid, but for high number of dimensions, the cost of solving the problem on each polyhedron becomes prohibitive, while computing the convex hull remains a relatively cheap operation.

The second case where taking the convex hull is useful is for (some) discrete models. Indeed:

$$\min_{x_i \in \{-1,1\}} c^T x \iff \min_{x} c^T x \\ -1 \le x_i \le 1$$

The convex hull of  $\{-1,1\}^n$  is  $[-1,1]^n$ , and so both problems have the same solutions. The flow problem with integers (as seen in LINMA1702), which retains integer only solutions

with relaxation, is an example of taking the convex hull of discrete problems without changing the nature of solutions.

#### UNTIL THERE

#### v. Approximate any convex problem by a linear problem

It is possible to approximate convex problems by linear ones. Since the objective can always be converted to a linear function by adding a variable, the only work to be done concerns the constraints. Basically, the idea is to approximate the (closed<sup>7</sup>) set X by a finite intersection of half-spaces (thus, linear constraints).

The way to do this is to use **projections** of points on the convex space. The projection of a point u on a set X is defined as the point  $u_p \in X$  that minimizes the distance between itself and u.

**Theorem A.2.** Uniqueness and existence of projection Let X be a **closed**, **non-empty**, **convex** set in  $\mathbb{R}^n$ , the projection of any exterior point on X exists and is unique.

It is easy to see that the closeness and non-emptiness guarantee the existence of such projection. The unicity, however, is ensured by the convexity of the set. As an example, it is easy to see that a non-convex set such as the unit circle as an infinite number of projections of the origin.

With this concept of projection, we can introduce the **separation property**: for every exterior point u of a convex closed set X, there exists a plane that  $separates\ u$  from X, that is, such that every element of X is on  $one\ side$  of the plane, while u is on the  $other\ side$ . This results directly from the uniqueness and existence of the projection of u on X, although this was not demonstrated in class. Intuitively, this separation plane can be built perpendicular to the segment joining u and its projection, without intersecting the convex plane.

So, to approximate a convex set by a linear one, the following method should be applied: for every point that should not be in X, create a separation plane. Such plane defines a half-space containing all of X. The intersection of all the half-spaces obtained this way forms a polyhedron containing X.

Moreover, it is interesting to note that an infinite number of points will create the convex set itself! This yields a new definition for a convex set: a convex set can always be written as the infinite intersection if half-spaces. This definition also proves immediately that every polyhedra is convex.

<sup>&</sup>lt;sup>7</sup>Why not open? Because an open set corresponds to strict inequalities, which cannot and will not be treated with the common optimization tools (see the note on  $\neq$ )

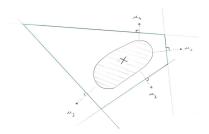


Figure A.4.: Approximation of a non-linear problem by a linear one.

#### c. Modelling Tricks

After studying the standard form of some optimization problems, we will see some modelling tricks which permit to transform a non-linear or non-convex problem into a linear or convex optimization problem.

#### i. Monotonicity

Definition 1. A monotonic function over an interval is a function that is either increasing or decreasing over this interval.

The transformation that is described in the following lines uses monotonic functions to turn an optimization problem into a simpler one. These operations do not change the problem but can change the solution and the optimal value of the objective of the problem. Let's see some examples.

**Example 1**  $min ||x||_2$  with  $x \in X$  is equivalent to  $min ||x||_2^2$  with  $x \in X$ . In fact,  $z \to z^2$  is a monotonic function (increasing in this case) over  $\mathbb{R}$ . By solving this problem, we will find the same optimal solution than the first model but the value of the objective function will be different.

**Example 2** These functions are monotonic and can be used like in the previous example to simplify the problem :

- $z \rightarrow e^z$
- $z \rightarrow log(z), (z > 0)$
- .  $z \to -\frac{1}{z}, \ (z>0)$  for example, we can change  $\min \, \frac{1}{\|x\|_2}$  to  $\min \, \|x\|_2$

We not only use this trick to modify the objective function, but also some constraints. For example :

$$f(x) \le b \Leftrightarrow e^{f(x)} \le e^b$$

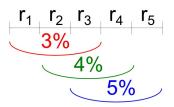


Figure A.5.: Rates

**Example : Advertisement for bank account :** This example allows us to look at the effects of monotonicity in real life optimization problems. We put money into an account where we can't take our money back sooner than 5 years after. The bank guarantees that we have a high percentage when we take back our money after 5 years and assures the three rate over 3 years given on Figure A.5.

What is the worst global rate compatible with this? Therefore, we are looking for the cumulative effect of the 5 rates (given over a year).

To solve this problem, we introduce the variables  $r_i$  corresponding to the rate per year where i = 1, ..., 5. We don't want a rate which is negative or null so we have the following optimization problem:

 $\min_{r_i>0} r_1 r_2 r_3 r_4 r_5$ 

under the constraints:

 $r_1r_2r_3 \ge 1.03$ 

 $r_2r_3r_4 > 1.04$ 

 $r_3r_4r_5 \ge 1.05$ 

This problem is not linear. We can transform it into a linear problem if we use the logarithm function. Indeed, let  $y_i = log(r_i) \ \forall i = 1,...,5$ . We can do this because the logarithm is a monotonic increasing function. The problem can be written as:

$$min y_1 + y_2 + y_3 + y_4 + y_5$$

under the constraints:

$$y_1 + y_2 + y_3 \ge log(1.03)$$

$$y_2 + y_3 + y_4 \ge log(1.04)$$

$$y_3 + y_4 + y_5 \ge log(1.05)$$

This problem is now linear (and thus convex).

#### Remark

If we don't make the change of variable  $y_i = log(r_i) \ \forall i = 1, ..., 5$ , the variables  $r_i$  appear in log so the problem isn't linear.

#### ii. Change of variables

We use change of variables in order to transform the problem into a linear or convex problem. For example, if every variable appears in a logarithm, then we can use the change of variable to remove the logarithm.

**NB**: every appaerance of the variables needs to match the change of variables!

For example, signomials <sup>8</sup> can be converted into convex functions thanks of this trick. Let's consider the signomial  $\frac{x_1x_2^2}{x_3^2}$ . Let  $x_i = e^{y_i}$ , we obtain the following expression by

change of variable  $e^{y_1+2y_2-\frac{1}{2}y_3}$ . Moreover, the exponential of a linear function is convex and the change of variables conserves the convexity (See later).

#### iii. Misleading/Deceptive appearances

In this part, we are looking for a polynomial p(x),  $x \in \mathbb{R}$  of degree D which fulfils some characteristic (See Figure A.6):

- . For  $x \in [0; f_1], p(x) \ge 3$
- . For  $x \in [f_2; f_3], p(x) \le 0.5$

Let  $p(x) = \sum_{i=0}^{D} a_i x^i$ . On the Figure A.6, we observe that the polynomial will not be linear or convex. We then have the impression that the optimization problem is not linear. But it is not the case, given that the variables are not the  $x_i$  but the  $a_i$ . Furthermore, the constraints (2 examples of constraints are given below) are linear.

$$\sum_{i=0}^{D} a_i 50^i \ge 3 \Leftrightarrow p(50) \ge 3$$

$$\sum_{i=0}^{D} a_i 100^i = 1 \Leftrightarrow p(100) = 1$$

#### iv. Flexibility

The flexibility of an optimization problem is its capacity to be solved in different ways depending on which criterion we really want to optimize. Let us consider an approximation problem. We have a set of points  $(x_i, y_i)$  in  $\mathbb{R}^2$ , and we want to find the function (which is often a polynomial) which approximates at best these points. In other words, we want to find a function f that minimizes the errors  $\epsilon_i$  at the approximation points:

$$\epsilon_i = |y_i - f(x_i)|$$

<sup>&</sup>lt;sup>8</sup>A signomial is an algebraic function of one or more variables of the form:  $f(x_1, x_2, ... x_n) = \sum_i \left( c_i \prod_j x_j^{a_{ij}} \right)$ 

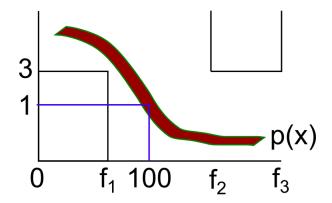


Figure A.6.: Polynomial p(x)

The question now is to choose the way to minimize errors  $\epsilon_i$ . Indeed, we could choose to minimize overall errors with a least square criterion:

$$\min ||\epsilon||_2 = \sqrt{\sum_i \epsilon_i^2}$$

Another choice would be to minimize the sum of the errors:

$$\min ||\epsilon||_1 = \sum_i |\epsilon_i|$$

Finally, a last (often used) way to minimize errors is to minimize the maximum error:

$$\min_{i} \max_{i} \epsilon_{i}$$

All these considerations prove that, depending on the criterion we choose to optimize (i.e. depending on the context), the result could be different. For example, the problem

$$\begin{array}{ccc} \max & safety \\ cost & \leq & m \end{array}$$

will not lead to the same solution than

$$\begin{array}{ccc}
\min & cost \\
safety & \geq & b
\end{array}$$

#### v. Charnes and Cooper

Consider the non-linear problem which is a division of two linear expression :

$$\min \ \frac{c^T x + d}{f^T x + g}$$

under the constraints:

$$Ax \leq b$$

In this case, taking the logarithm of the objective function will not change anything: we will have the same problem.

We make the hypothesis that  $f^Tx + g > 0 \ \forall x \ such that \ Ax \leq b$ . If not, we have the solution  $-\infty$  which is not an interesting solution.

To solve this problem and make it linear, we are going to homogenize the objective function. Let  $x = \frac{y}{t}$  with  $y \in \mathbb{R}^n$  and  $t > 0 \in \mathbb{R}$ . If we take t = 1 then we get back to the original problem. So, one solution in x correspond to several solutions in (y,t) (for example :  $(x,1), (2x,2), ... (\lambda x, \lambda)$ ).

We can write the problem as:

$$\min \frac{\frac{c^T y}{t} + d}{\frac{f^T y}{t} + g}$$

with:

$$A\frac{y}{t} \le b$$

By simplification, we obtain the following problem:

$$\min \ \frac{c^T y + dt}{f^T y + gt}$$

with:

$$Ay \le bt$$

Now, we notice that the objective function's numerator and denominator are linear as the constraint. This problem has a property called homogeneity. If you take any solution, you can multiply any component by the same constant and nothing changes. Mathematically

(y,t) solution  $\Rightarrow (\lambda y, \lambda t)$  solution  $\forall \lambda \ 0$ 

We are gooing to choose solutions satisfying:  $f^Ty + gt = 1$ . We can do this using the property of homogeneity. This step results in selecting one solution among the collection of solutions multiple of each other. The objective function gets simpler and linear and we add one constraint. So, we have the following linear optimization problem:

$$\min c^T y + dt$$

with:

$$Ay - bt \le 0$$
$$f^T y + gt = 1$$
$$t > 0$$

We compute  $y^*$  and  $t^*$  from this problem and take  $x^* = \frac{y^*}{t^*}$  the solution of the original problem.

NOTE: If we have t=0 at the optimum then the problem is unbounded and  $x^* \to \infty$ . Example:

 $\min \frac{1}{x}$ 

with:

This problem have an optimal value of 0 so the solution  $x^* \to +\infty$ .

#### d. Convex Optimization: Theorems and properties

#### i. Convex sets

#### Definition and examples

Definition 2. A set X is convex if and only if

$$x, y \in X \Rightarrow \lambda x + (1 - \lambda)y \in X \quad \forall \ 0 \le \lambda \le 1$$

Thus, a set is convex if and only if it contains all the segments joining any pair of its points.

#### **Examples:**

- $\mathbb{R}^n$ ,  $\mathbb{R}^n_+$ ,  $\emptyset$
- hyperplans  $(\{x | b^T x = \beta\})$
- open or closed half-spaces ( $\{x | b^T x < \beta\}$  and  $\{x | b^T x \le \beta\}$ )
- open and closed balls  $(\{x| \|x-a\| < r\} \text{ and } \{x| \|x-a\| \le r\})$

#### Properties

Property 1. Given a collection of convex sets  $\{C_i\}_{i\in I}\subseteq \mathbb{R}^n$  (I can be arbitrary), then  $\bigcap_{i\in I}C_i$  is convex too.

It follows that polyhedrons are convex because they are intersection of half-spaces.

Property 2. Given a collection of convex sets  $C_1$ ,  $C_2$ ,  $C_3$ , ...,  $C_n$ , their cartesian product  $C_1 \times C_2 \times C_3 \times ... \times C_n$  is convex too.

Property 3. If  $X \subseteq \mathbb{R}^n$  and  $X \subseteq \mathbb{R}^n$  are convex then the Minkowski sum of X and Y,  $X + Y = \{x + y \mid x \in X \text{ and } y \in Y\}$  is convex too.

#### Remark

The union of convex sets is not always convex!

#### ii. Convex functions

#### Definition and examples

Definition 3. A function f with domain D is a convex function of and only if

D is convex and

$$x, y \in D \Rightarrow f(\lambda x + (1 - \lambda)y) \le \lambda f(x) + (1 - \lambda)f(y) \quad \forall \ 0 \le \lambda \le 1$$

#### **Examples:**

- linear and affine functions are convex  $(x \to \alpha x, x \to b^T x \text{ and } x \to b^T x + \alpha)$
- the norm function and the square of the norm function are convex functions  $(x \to ||x||$  and  $x \to ||x||^2)$
- quadratic forms  $(x \to x^T Q x)$  are convex functions when the matrix  $Q \in \mathbb{R}^{n \times n}$  is semi positive definite
- the functions  $x \to e^x$ ,  $x \to log(x)$  and  $x \to |x|^p$   $(1 \le p)$  are convex Definition 4. A function f is concave  $\Leftrightarrow -f$  is convex.

#### Remark

Linear and affine functions are convex and concave.

#### **Properties**

To know whether a function is convex or not, we have to transform it into its epigraph and check if it is convex or not. But there are some useful properties of convex functions that we can use to spare time.

Property 4. If f is a convex function and  $c \in \mathbb{R}_0^+$ , then cf is convex.

Property 5. If f and g are convex functions, then f + g is convex.

Property 6. Given a collection of convex functions  $\{f_i\}_{i\in I}: \mathbb{R}^n \to \mathbb{R}$ ,  $\sup_{i\in I} f_i$  is convex too.

with 
$$\left[\sup_{i\in I} f_i\right](x) = \sup_{i\in I} f_i(x)$$

Property 7. Given f(x,s) (with  $x \in \mathbb{R}^n$  and  $s \in \mathbb{R}$ , a parameter) such that  $x \to f(x,s)$  is convex for any s,

$$\int_{s \in S} f(x, s) ds \text{ is convex}$$

#### Convexity and differential calculus

 $\overline{Property \ 8. \ \text{Let } f \text{ be a differentiable function of which the domain D is open. } f \text{ is convex if and only if D is convex and}$ 

$$\forall x, y \in D, \ f(y) \ge f(x) + \nabla f(x)^T (y - x)$$

This property means that, if f is convex, it will be above all its Taylor approximations of first order. This signifies that at any point, the tangent of the function is under the function. This is useful to make a piecewise approximation of f by linear functions (an example of such an approximation is shown on Figure A.7). In order to obtain such an approximation, we have to choose n points at which we calculate the tangent of the function f and then take the maximum of the n tangents (in the example, n = 3).

Besides, the value of the approximation is lower than the real value on any point. It allows us to obtain lower bounds.

Property 9. Let f be a twice differentiable function of which the domain D is open. f is convex if and only if D is convex and

$$\forall x \in D, \ \nabla^2 f(x) \ge 0$$

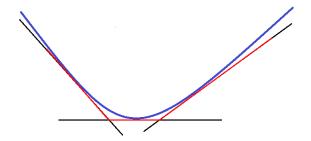


Figure A.7.: Illustration of the piecewise linear approximation of a convex function

#### iii. Convexity and linear transformations

Linear transformations preserve convexity. Indeed,

Property 10. If  $S \subseteq \mathbb{R}^n$  is convex and  $\Phi: \mathbb{R}^n \to \mathbb{R}^m: x \to Ax + b$  is a linear transformation, then the image of S by  $\Phi$ ,

$$\Phi(S) = \{\Phi(x) | x \in S\}$$

is convex too.

Property 11. If  $\Phi: \mathbb{R}^n \to \mathbb{R}^m: x \to Ax + b$  is a linear transformation and  $f: x \to f(x)$  is a convex function, then the composition

$$f \circ \Phi = f(\Phi(x)) = f(Ax + b)$$

is convex too.

Property 12. If  $S \subseteq \mathbb{R}^n$  is convex and  $\Theta : \mathbb{R}^m \to \mathbb{R}^n : x \to ax + b$  is a linear transformation, then the image of S by the inverse of  $\Theta$ ,

$$\Theta^{-1}(S) = \{ x | \Theta(x) \in S \}$$

is convex too.

## B. First-order methods

# C. Conic modeling and duality

# D. Interior-point methods

Part II.

Labs

#### 1. Introduction: AMPL

AMPL (which stands for "A mathematical programming language") is an algebraic modelling language which enables the solving of high complexity problems for large (or small) scale models and will be used throughout this course. Knowing or identifying what type of problem we're dealing with is important and allows us to decide which will be the most adequate method to solve it. Certain methods work well with certain types of models, some are better than others, etc.

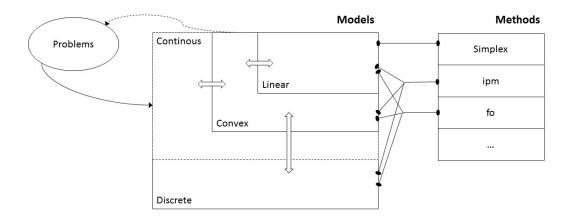


Figure D.1.: Visualization of the relation between models and methods

AMPL works by first reading a text file which contains all the useful information of the model, this file usually ends in ".mod", it then parses it and tries to solve the problem. Parameters, variables, objective function and constraints are all defined in this file. The solving part is done by communicating with a solver, AMPL gives it all the information, the solver then sends back the solution. Consider the following optimization problem .

$$\min c^T x$$
$$Ay \le b$$
$$l \le y \le u$$

For this problem AMPL will give the solver the usefull values (c,A,b,l,u,...) it needs to solve the problem. There exists quite a few of these solvers, some mode adapted to certain model types (linear, convex,...). We have for example:

- minos (basic solver for linear and nonlinear problems)
- cplex (can be used for linear, convex, mixted integer models)
- gurobi (very similar to cplex)
- knitro (good for nonlinear models)
- ...

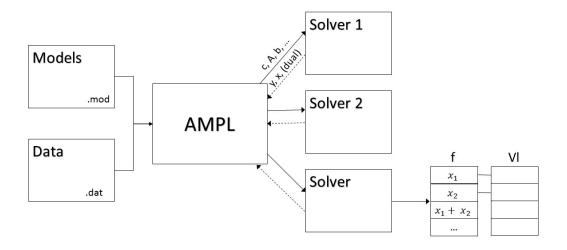


Figure D.2.

AMPL can also work with an additionnal data file (".dat") which is used when parameters are left in the model file. This allows us to avoid changing the entire file when looking at different values of parameters, and only having to change them once in the data file.

A few examples and the basic syntax for AMPL can be found in the Tutorials Dropbox, given on Moodle.

Everything in AMPL has a name, whether it be variables, constants, or even constraints (which represent a dual variable) and each command and declaration ends with a semi-colon (";"). Certain commands are worth being reminded here, for example:

- Changing solvers : option solver ... ;
- Displaying dual variable : display Protein;
- Displaying variable : display Protein.body;
- Reset the whole model: reset;
- Chosing a model file : data data1.dat;
- Chosing a data file: model data1.mod;

• Solving the chosen model : solve;

Part III.

**Exercices**