Introduction à la Programmation linéaire

Recherche opérationnelle versus programmation mathématique

- La Recherche Opérationnelle (RO) inclut des domaines de recherche tels que la théorie et l'algorithmique des graphes et l'optimisation combinatoire, etc.

- La Programmation Mathématique (PM) recouvre les méthodes qui servent à déterminer l'optimum d'une fonction sous des contraintes données et l'optimisation combinatoire en fait partie.
  - On parle de Programmation Linéaire (PL) quand la fonction à optimiser et les contraintes sont des fonctions linéaires.

- La théorie et l'algorithmique des graphes permet de déterminer des algorithmes spécifique pour résoudre des problèmes combinatoire tandis que la PM, en propose des formulation mathématique génériques pour résoudre les problèmes combinatoire efficacement (surtout avec la PL);

Introduction à la programmation mathématique

- Un *programme mathématique* est un problème d'optimisation d'une fonction-objectif de *n* variables, en présence de *m* contraintes. Si au moins une contrainte ou l'objectif est non-linéaire, on a un *programme non-linéaire*.

  Un programme est convexe si la fonction d'optimisation est convexe et le domaine défini par les contraintes est aussi convexe : dans ces conditions tout optima local est également un optima global et il est atteint dans un point extrême du domaine.

Programmation Linéaire versus programmation mathématique

- **Programmation Linéaire** (voir poly RO04)
  - On entends donc par programmation linéaire l'optimisation d'une fonction linéaire de variables x_1, x_2, …x_n, soumises à des contraintes linéaires sous forme d'égalités ou inégalités non strictes.

- **Le problème :**
  - Un *programme linéaire* est un problème qui consiste à trouver un extremum (maximum ou minimum) d'une fonction linéaire de plusieurs variables (la *fonction économique*), ces variables devant en outre vérifier un système d'équations et/ou d'inéquations linéaires (les *contraintes*).

- **L'avantage de la PL versus la PNL**
  - Travailler avec la PL au lieu de Programme Non Linéaire facilite la modélisation, la résolution et l'interprétation des solutions. Des solveurs puissants sont disponibles.

La place de l'optimisation (PM, PL) dans la formation d'un ingénieur

- Pour les ingénieurs
  - Savoir modéliser
  - Connaitre les outils/solveurs du marché
  - Comprendre les méthodes de résolution

- Pour les ingénieurs en optimisation et data science, ingénieurs dans des centres d'étude et de recherche, chercheurs
  - Maitriser les méthodes de résolution
  - Concevoir des méthodes spécifiques pour des problèmes difficiles
  - Améliorer les solveurs…

Brève description de l’UV RO04

- Contenu du RO04
  - Programmation/optimisation linéaire
    - Bases de modélisation
    - Méthodes de résolution
      - Méthode Simplex pour les PL continue
      - Méthodes spécifiques pour PLNE (PL en Nombres Entiers)
  - Programmation/optimisation non linéaire (2eme partie de RO04)
    - Etude des fonctions/domaines convexes et non convexes, étude des conditions d’optimalité, méthodes de résolution…

Contenu du cours

- Présentation générale de la PL
- Eléments de modélisation, exemples de problème combinatoire
- Brève description des méthodes de résolution

Présentation générale d'un PL

Formulation standard d'un PL

La formulation standard d'un PL:

maximiser $f(x)$

sous les contraintes:

$h_i(x) = 0, i = 1, 2, \ldots, m$

$g_j(x) \leq 0, j = 1, 2, p$

$x \in S$.

Où x est un vecteur *n-dimensionnel des variables* $x = (x_1, x_2, \ldots, x_n)$, et f, h_i, $i = 1, 2, \ldots, m$, *et* g_j, $j = 1, 2, \ldots, p$, *sont des fonctions réels des* variables $x_1, x_2, \ldots, x_n$.

S est un sous-ensemble de l'espace R^n. *La fonction f est la fonction objective et les équitations, inéquations et le domaine d'appartenance définissent les contraintes.*

LA FORMULATION STANDARD

maximiser $c_1x_1 + c_2x_2 + \ldots + c_nx_n$

s.c. $A_{11}x_1 + A_{12}x_2 + \ldots + A_{1n}x_n = b_1$

$A_{21}x_1 + A_{22}x_2 + \ldots + A_{2n}xn = b_2$

...

$A_{m1}x_1 + A_{m2}x_2 + \cdots + A_{mn}x_n = b_m$

et $x_1 \geq 0, x_2 \geq 0, \ldots, x_n \geq 0$.

Formulation matricielle :

maximiser $c^T x$

s.c.: $Ax = b$ and $x \geq 0$.

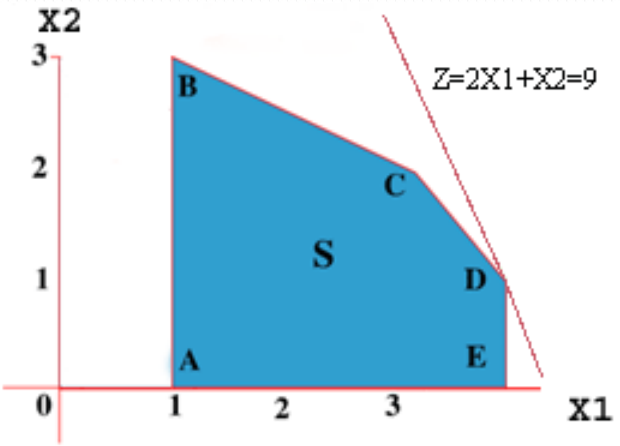
Note: vector $b \geq 0$

Interprétation géométrique

maximiser: $Z = 2x_1 + x_2$

sous les contraintes:
$$x_1 + 2x_2 \leq 7$$
$$x_1 + x_2 \leq 5$$
$$x_1 \geq 1$$
$$x_1 \leq 4$$
$$x_i \geq 0$$

On voit grâce au dessin que la valeur du maximum de Z est égale à neuf sur cet exemple :



MODELISATION

Formulation

- Dans un PL on peut distinguer:
  - Variables de décision
  - Paramètres (les données du problème)
  - Fonction Objective (ou d’optimisation)
  - Contraintes;
  - Domain des variables;

    Note : dans un PL le domaine des variables correspond aux nombres réels positifs et dans un PLNE il correspond aux nombres entiers positifs.

Un premier cas d'étude
le problème d'allocation de ressources

Cas général

On suppose que le programme linéaire est présenté sous une forme telle que :

- La fonction économique est en *maximisation* ;

- Les contraintes comprennent des *conditions de positivité* sur *toutes* les variables.

C'est-à-dire, pour un programme linéaire à ***n*** variables $\boldsymbol{x_j}$:

Interprétation économique :

n activités, m ressources…

$$Max \quad z = \sum_{j=1}^{n} c_j x_j$$

$$s.c. \quad A_{11}x_1 + A_{12}x_2 + \ldots + A_{1n}x_n \leq b_1;$$

$$A_{21}x_1 + A_{22}x_2 + \ldots + A_{2n}x_n \leq b_2;$$

$$\ldots$$

$$A_{m1}x_1 + A_{m2}x_2 + \ldots + A_{mn}x_n \leq b_m;$$

$$x_{1,} x_{2,} \ldots, x_{n,} \geq 0.$$

Modélisation, exemple

Exemple. Examinons le cas d'un programme de production où il faut déterminer quelles seront les quantités à fabriquer pour optimiser le résultat, et ce en tenant compte des contraintes de production. Supposons une entreprise fabriquant deux produits A et B, passant tous deux dans les ateliers découpe (D) et finition (F) :

Atelier	**Découpe**	**Finition**
Temps de fabrication de A	2 heures	3 heures
Temps de fabrication de B	2 heures	1 heure
Capacité maximale de production	200 heures	100 heures

Sachant que les marges unitaires des produits A et B sont respectivement de 20 € et 10 €, quelles seront les quantités à produire pour maximiser le résultat ?

x_A, x_B

$2x_A + 2x_B \leq 200$

$3x_A + x_B \leq 100$

$\max\ 20x_A + 10x_B,\quad x_A, x_B \in \mathbb{N}$

Modélisation, exemple

Le problème de gestion d'entrepôt

Considérons le problème d'un entrepôt quand on doit acheter et vendre de la marchandise tout en maximisant le profit des ventes. L' entrepôt a une certaine capacité C il y a un cout unitaire r pour l'emmagasinage du stock par mois. Les prix d'achat et de vente varient avec le temps (indexé par mois i). Pour chaque mois i le prix d'achat est f_i et de vente p_i. L'entrepôt est initialement vide et il doit être aussi à la fin de la période.

$$\text{Max} \sum_i (p_i s_i - f_i u_i - r x_i)$$

$$x_{i+1} = x_i + u_i - s_i$$

$$0 = x_n + u_n - s_n$$

$$x_i \leq C$$

$$x_i \geq 0,\ u_i \geq 0,\ s_i \geq 0$$

Modélisation, exemple

Surveillance des rues par des caméras.

- Une municipalité souhaite installer des caméras dans les carrefours d'une zone industrielle suite à des vols à répétition. On suppose que chaque portion de rue entre deux carrefours est en ligne droite. Une caméra installée à un carrefour peut pivoter à 360 degrés et voir toutes les rues adjacentes. La commune souhaite installer un nombre minimal de caméras. Proposez un modèle mathématique pour ce problème. On peut représenter la zone de surveillance par un graphe où les nœuds sont les carrefours et les arcs les rues.
  - C'est un problème de la théorie des graphes : le problème de couverture des nœuds.

$$x_i = \begin{cases} 1 & \text{si } i \in \text{Couverture} \\ 0 & \text{sinon} \end{cases}$$

$$G = (X, U)$$

$$\begin{cases} \text{Min} \sum_{i=1}^{n} x_i \\ \forall (i,j) \in U \quad x_i + x_j \geq 1 \\ x_i \in \{0,1\} \quad \forall i \in 1..n \end{cases}$$

Formulation des problèmes combinatoires vus en cours

- Problème de numérotation d’un graph sans circuit
- Problème de couplage maximal
- Problème du circuit hamiltonien
- Problème du plus court chemin

Note : ce sont tous des problèmes combinatoires, et par conséquent des PLNE.

Formulation de problèmes combinatoires

Numérotation du graphe

- Variables de décision
  - x_i -> numéro du sommet i,
- Paramètres
  - Graphe G=(X,U) de n sommets
- Contraintes
  - $\forall\ (i,j)\in U \qquad x_i \leq x_j + 1;$
  - $\forall\ i\in X \qquad x_i \leq n;$
  - $\forall\ i\in X \qquad x_i \in N;$
- Fonction objective
  - Max 0 X

Couplage maximal

- Variables de décision
  - x_{ij} -> 1 si arête $(i,j)\in$Couplage et 0 sinon.
- Paramètres
  - Graphe G=(X,U) non orienté de n sommets
- Contraintes
  - $\forall\ i\in X \qquad \sum_{j\in D(i)} x_{ij} \leq 1;$
  - $\forall\ (i,j)\in U \qquad x_{ij} \in \{0,1\};$
- Fonction objective
  - Max $\sum_{(i,j)\in U} x_{ij}$

Le problème de voyageur de commerce

Historique

- **19ième siècle**
  - Les premières approches mathématiques exposées pour le problème du voyageur de commerce ont été traitées au **19ième** siècle par les mathématiciens Sir William Rowan Hamilton et Thomas Penyngton Kirkman.
- **années 1930**
  - Le PVC est traité plus en profondeur par Karl Menger à Harvard. Une attention particulière est portée sur les connections par Menger et Whitney.
- **1954**
  - Solution du PVC pour 49 villes par Dantzig, Fulkerson et Johnson.
- **1975**
  - Solution pour 100 villes par Camerini, Fratta et Maffioli
- **1987**
  - Solution pour 532, puis 2392 villes par Padberg et Rinaldi
- **1998**
  - Solution pour les 13 509 villes des Etats-Unis.
- **2001**
  - Solution pour les 15 112 villes d'Allemagne par Applegate, Bixby, Chvàtal et Cook des universités de Rice et Princeton.

Le problème de voyageur de commerce: formulation mathématique

Formulation en PLNE:

– A chaque arc (i,j), associons les variables x_{ij} prenant la valeur 1 s'il est dans le circuit et 0 sinon.

$$Min \quad \sum_{i,j} d_{i,j} x_{i,j}$$

$$\sum_{j} x_{i,j} = 1 \qquad i = 1, \dots, n \quad (a)$$

$$\sum_{i} x_{i,j} = 1 \qquad j = 1, \dots, n \quad (b)$$

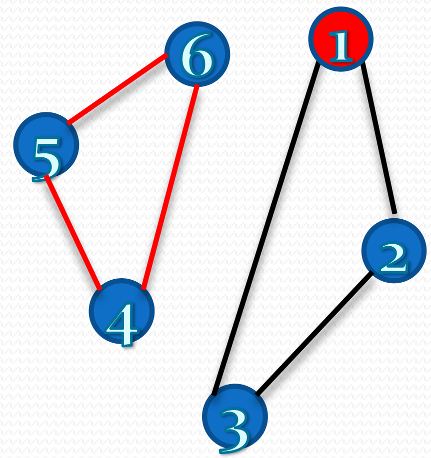
$$x_{i,j} \in \{0,1\} \qquad \forall i, \forall j.$$

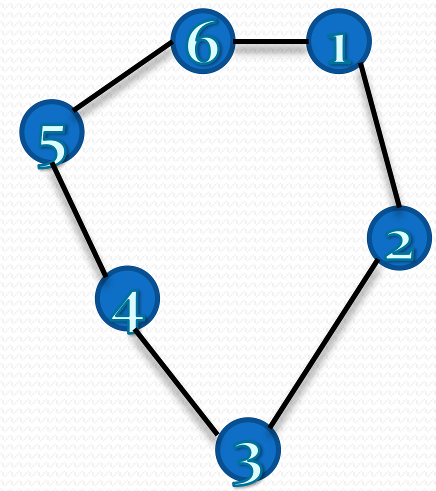
$$Min \quad \sum_{i,j} d_{i,j} x_{i,j}$$

$$\sum_{j} x_{i,j} = 1 \qquad i = 1, \dots, n \qquad (a)$$

$$\sum_{i} x_{i,j} = 1 \qquad j = 1, \dots, n \qquad (b)$$

$$x_{i,j} \in \{0,1\} \qquad \forall i, \forall j.$$





Différentes possibilités existent pour « briser » ces cycles:

- Ordonner les sommets en introduisant les variables entières u_i

$$u_i - u_j + n x_{i,j} \le n - 1 \quad 2 \le i \ne j \le n, \quad u_i \in N \quad \text{(C)}$$

$$u_4 - u_5 + n * 1 \le n - 1$$
$$u_5 - u_6 + n * 1 \le n - 1$$
$$u_6 - u_4 + n * 1 \le n - 1$$
$$3(n) \le 3(n-1)$$

??

Le problème du plus court chemin : formulation mathématique

Formulation en PLNE du problème du plus court chemin de s vers t dans un graphe valué

Variables de décision

x_{ij} -> 1 s'il est dans le plus court chemin et 0 sinon

Paramètres

Graphe G=(X,U,w, s, t)

Contraintes

$\forall \; i \notin \{s,t\}$ $\quad \sum_{j \in U+(i)} x_{ij} = \sum_{k \in U-(i)} x_{ki}$;

$\sum_{j \in U+(i)} x_{sj} = \sum_{k \in U-(t)} x_{kt} = 1$;

$\forall \; (i,j) \in U$ $\quad x_{ij} \in \{0,1\}$;

Fonction objective

Min $\sum_{(i,j) \in U} w_{ij} x_{ij}$

$x_{ij} \in \{0, 1\}$

METHODES DE RESOLUTION

Méthodes de résolution pour la PL

- **Comment résoudre un PL :**
  - Algorithme de Simplexe Dantzig (1951, 1963), très efficace (mais non polynomial). L'apparition de logiciels commerciaux très puissants (CPLEX, EXPRESS-MP, etc.) permet, aujourd'hui, de mettre cet outil à la disposition d'un large public.

- **Méthodes alternatives au simplexe :**
  - Méthode de l'ellipsoïde (Khachyan 1979), polynomial mais pas vraiment efficace;
  - Méthode des points intérieurs (Algorithme de Karmarkar 1984) : polynomial et pratiquement efficace

Algorithme de Simplexe : principes de base

La zone des solutions définie par les contraintes forme un polyèdre, et la recherche de l'optimum peut être restreinte à l'ensemble de ses points extrêmes.

La méthode du simplexe explore les points extrêmes en améliorant à chaque itération la valeur du critère.

Avantages :

- La complexité de la méthode n'est pas polynomiale mais pratiquement elle donne de très bons résultats.
- Elle donne également de résultats théoriques (dualité, géométrie des polyèdres).

Programme Linéaire en Nombres Entières (PLNE)

Dans la classe des Programmes Linéaires on distingue le cas avec des variables en nombres entières (PLNE). Ce-dernier est d'habitude plus dur à résoudre et les problèmes représentés sont quasi toujours NP-Complets.

L'optimum n'est plus en général un sommet du polyèdre et ne peut être calculé par l'algorithme de Simplexe. Il faut des méthodes arborescentes lourdes (B&B, B&C, B&P), limitées à quelques centaines de variables et contraintes.

Les PL en 0-1, sont des cas particuliers de PLNE. Les variables binaires correspondent souvent à des variables de décision.

Méthodes exactes de résolution

- Énumération complète des solutions : *Branch and Bound method (B&B),* séparation et évaluation.
  - *Principe : On construit un arbre de recherche dont le problème initial (problème de minimisation) est la racine.*
  - *On divise le problème en sous-problèmes : l'optimum peut appartenir à l'un quelconque de ces sous-problèmes.*
    - *tout sous-problème infaisable sera éliminé;*
    - *si possible on calcul la solution du sous-problème.*
    - *sinon on calcule une borne inférieure, si elle est supérieure de la meilleure solution déjà obtenue, on éliminé le sous-problème.*
    - *Dans le cas restant, on sub-divise à nouveau le domaine.*

- *La méthode est particulièrement adaptée à la résolution des PL01 et des PLNE dans un cadre plus général.*

Branch and bound
un problème de maximisation

Un exemple

Considérons le programme linéaire en nombres entiers suivant :

Max $F = 3x_1 + 8x_2$

$x_1 + 4x_2 \leq 20$ $(1) \Rightarrow x_2 \leq 5$

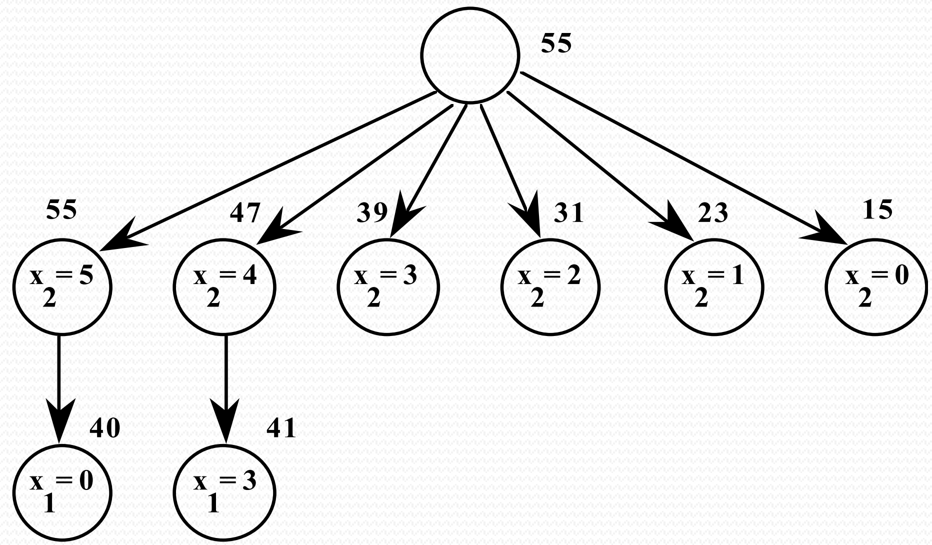
$x_1 + 2x_2 \leq 11$

$3x_1 + 2x_2 \leq 19$

$2\,(4x_1 - x_2 \leq 22)$ $\quad 11x_1 \leq 63$

$x_1, x_2 \in \mathbb{N}$ $\quad x_1 \leq 5$

Schéma de branchement et de valuation



Exemple

Soit donné le programme linéaire P en nombres entiers suivant :

Max $4x_1 + 4x_2 - 3x_3$

$4x_1 + 2x_2 - x_3 \leq 7$

$6x_1 + 2x_2 + 3x_3 \leq 10$

$x_1 + x_2 - x_3 \leq 2$

$x_1, x_2, x_3 \in \mathbb{N}$.

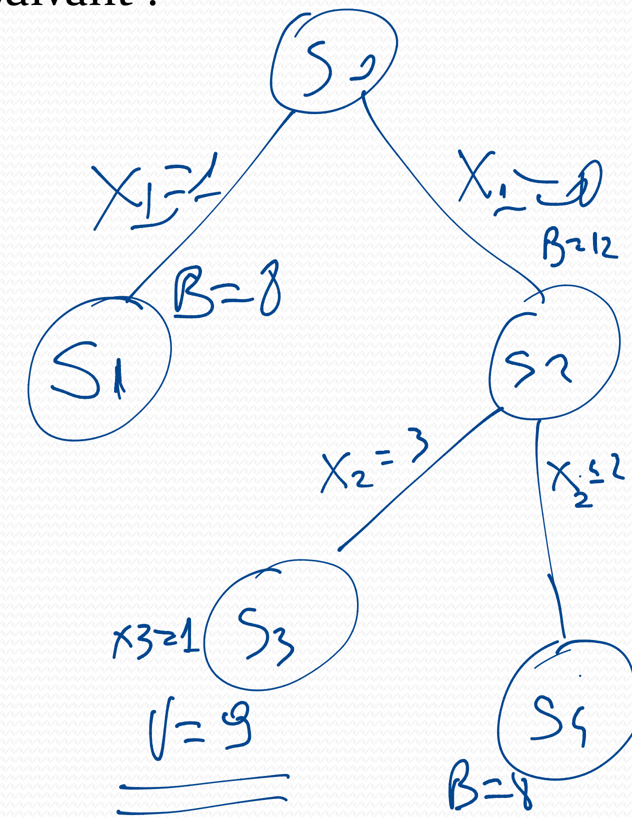
$9x_1 + 5x_2 \leq 16$

$x_1 \leq 1$

$x_2 \leq 3$

$5x_2 \leq 7$

$\Rightarrow x_2 \leq 1$



$x_1 = 0, x_2 = 3, x_3 = 1$

Programmation linéaire en variables booléennes

Méthode de FAURE-MALGRANGE

Principes :

1. ***Réécrire le model en utilisant des variables complémentaires tels que xi + ¬xi = 1.***
2. ***Construire une arborescence de recherche dichotomique séparant sur les variables ayant le plus grand coefficient dans la fonction objective;***

EXEMPLE:

Max $F = 3x1 + 5x2 - x3 + 2x4$

$3x1 + 4x2 - 2x3 + x4 \leq 5$

$2x1 - x2 + 3x3 - x4 \leq 4$

$4x1 - x2 - x3 + 2x4 \leq 5$

$x1, x2, x3, x4 \in \{0, 1\}$

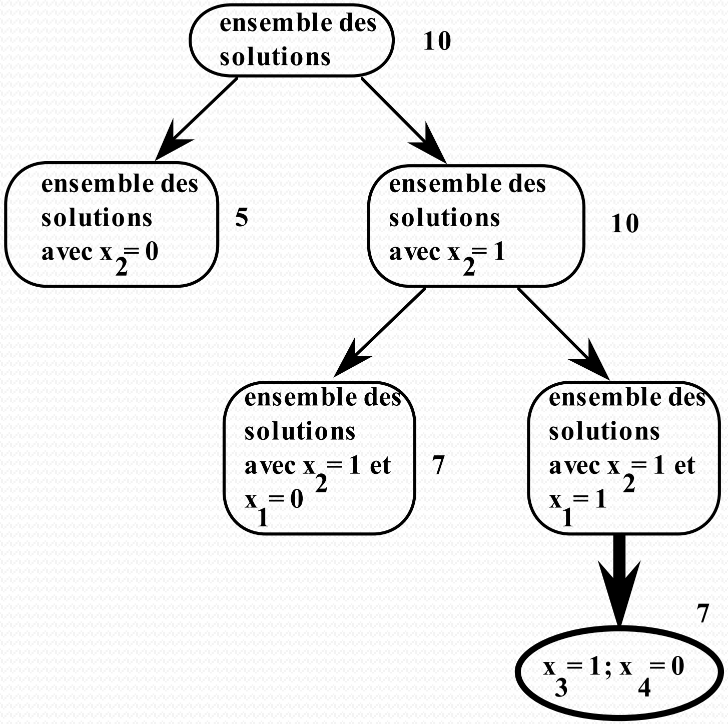
Nous introduisons ¬x1, ¬x2, ¬x3, ¬x4 définies par xi + ¬xi = 1, et on obtient :

$F = 10 - (3\neg x1 + 5\neg x2 + x3 + 2\neg x4)$

$3x1 + 4x2 + 2\neg x3 + x4 \leq 7$

$2x1 + \neg x2 + 3x3 + \neg x4 \leq 6$

$4x1 + \neg x2 + \neg x3 + 2x4 \leq 7$



Les solveurs

Il existe de nombreux solveurs de PL :

des solveurs commerciaux Cplex (IBM), Xpress, Gurobi, et même Matlab ou Excel… ;

des solveurs académiques Lp de COIN-OR, Soplex de la ZIB ;

des solveurs libres comme Glpk (gnu)…

Concernant les problèmes PLNE, les solveurs commerciaux Cplex ou Gurobi sont les plus performants (Xpress est un peu en-dessous) pouvant réussir parfois quelques milliers de variables/contraintes ; il y a aussi un solveur "universitaire" efficace, le SCIP de la ZIB.

FIN