

1. Optimisation linéaire

L'optimisation linéaire est un domaine de la recherche opérationnelle. Elle consiste à modéliser des problèmes de recherche opérationnelle à l'aide d'inégalités linéaires et à résoudre de tels problèmes à l'aide d'une approche algorithmique spécifique.

1 Définitions et exemple

Un *problème (d'optimisation) linéaire (PL)* est un problème d'optimisation (minimisation ou maximisation) d'une fonction linéaire sous des contraintes linéaires, les variables étant continues. Voici un exemple de problème d'optimisation linéaire :

PL 1

$$\min 2x_1 - x_2 \quad (1.1)$$

$$x_1 + x_2 = 10 \quad (1.2)$$

$$5x_1 + x_2 \geq -2 \quad (1.3)$$

$$x_1 \leq 0 \quad (1.4)$$

Le problème **PL 1** possède deux variables continues nommées x_1 et x_2 . La ligne (1.1) est la fonction linéaire (à minimiser), appelée *fonction objective* ou *fonction coût*. Les lignes (1.2)-(1.4) correspondent aux inégalités devant être respectées par les variables x_1 et x_2 et sont appelées *contraintes* du problème.

L'optimisation linéaire permet de modéliser beaucoup de problèmes de recherche opérationnelle. À titre d'exemple, considérons le problème suivant ¹.

Énoncé :

Une entreprise de fabrication de yaourts souhaite produire des yaourts à la fraise. Il lui est possible de fabriquer deux types de yaourts à la fraise : allégé et normal. La fabrication d'un litre de chaque type de yaourt nécessite différentes matières premières : produire un litre de yaourt allégé nécessite 2 kilos de fraises et 1 litre de lait, alors que la production d'un litre de yaourt normal nécessite 1 kilo de fraises, 2 litres de lait et 1 kilo de sucre.

Un litre de yaourt allégé peut être vendu 40 euros alors qu'un litre de yaourt normal peut être vendu 50 euros.

Étant donné que l'entreprise possède 800 kilos de fraises, 700 litres de lait et 300 kilos de sucre, combien de litres de yaourt allégé et normal doit-elle fabriquer pour maximiser son revenu ?

Modélisation :

Dans ce problème, il faut déterminer le nombre de litres de yaourt de chaque type à produire. On a donc deux variables. On notera x_A la quantité (en litres) de yaourt allégé et x_N la quantité (en litres) de yaourt normal.

L'utilisation de chaque ressource (fraise, lait et sucre) donne lieu à une contrainte. Par exemple, la quantité de fraises nécessaire pour produire les deux types de yaourts ne doit pas dépasser la quantité en stock. Si l'on exprime la contrainte avec les variables x_A et x_N , on obtient :

$$2x_A + x_N \leq 800$$

Comme une quantité ne peut être négative, nous avons aussi les contraintes $x_A \geq 0$ et $x_N \geq 0$. Finalement, l'objectif consiste à maximiser le revenu correspondant à la vente des quantités de yaourts produits. On obtient alors le problème d'optimisation linéaire **PL 2** :

¹. Exemple issu du cours de P. Pesneau

PL 2

$$\max 40x_A + 50x_N \quad (\text{revenu}) \quad (1.5)$$

$$2x_A + x_N \leq 800 \quad (\text{fraise}) \quad (1.6)$$

$$x_A + 2x_N \leq 700 \quad (\text{lait}) \quad (1.7)$$

$$x_N \leq 300 \quad (\text{sucrer}) \quad (1.8)$$

$$x_A \geq 0 \quad (\text{quantité positive}) \quad (1.9)$$

$$x_N \geq 0 \quad (\text{quantité positive}) \quad (1.10)$$

2 Résolution graphique

Si l'on a un PL avec deux variables (deux dimensions), on peut alors représenter les contraintes dans le plan. L'axe des abscisses correspond à la première variable et l'axe des ordonnées à la deuxième variable. Chaque inégalité peut être représentée par un demi-espace défini par la droite correspondant à l'équation associée à l'inégalité. L'intersection de ces demi-espaces définit l'ensemble des solutions réalisables du problème et est appelé *domaine réalisable* du problème linéaire.

Dans le problème PL 2, on a 5 inégalités, définissant 5 demi-plans (Le demi-plan associé à une inégalité est celui qui ne contient pas le trait bleu). Le domaine réalisable est dessiné en vert sur la figure 1.1.

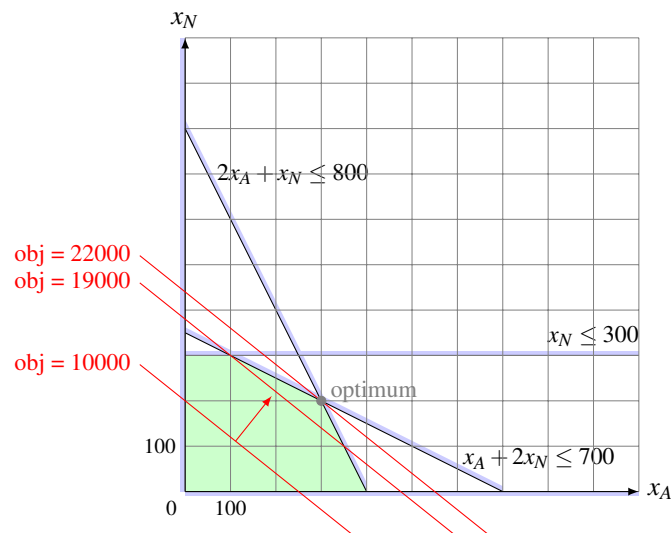


FIGURE 1.1 – Domaine réalisable et optimum

Définition 1 Un *sommet* du domaine réalisable est une solution vérifiant deux inégalités du problème linéaire à l'égalité.

Dans cet exemple, le domaine réalisable possède 5 sommets :

- $(0, 0)$ vérifiant les inégalités (1.9) et (1.10) à l'égalité,
- $(400, 0)$ vérifiant les inégalités (1.6) et (1.10) à l'égalité,
- $(300, 200)$ vérifiant les inégalités (1.6) et (1.7) à l'égalité,
- $(100, 300)$ vérifiant les inégalités (1.7) et (1.8) à l'égalité,
- $(0, 300)$ vérifiant les inégalités (1.8) et (1.9) à l'égalité.

Définition 2 Deux sommets du domaine réalisable sont dits *voisins* s'ils vérifient à l'égalité une même inégalité du PL.

Les sommets $(0,0)$ et $(400,0)$ sont voisins, de même que $(400,0)$ et $(300,200)$.

- R** Les concepts de sommets et sommets voisins existent également lorsque l'on a plus de deux variables (pour des dimensions supérieures à deux).

À toute solution est associé le coût de cette solution, correspondant à la valeur de la fonction objective pour cette solution. Plusieurs solutions ont le même coût. Ceci est représenté dans la figure par des droites correspondant aux solutions de même coût. Le vecteur indique la direction dans laquelle la valeur de la fonction objective augmente. On obtient la solution optimale (c'est-à-dire la meilleure solution) en suivant la direction donnée par ce vecteur jusqu'à la frontière du domaine réalisable.

Dans l'exemple, la solution optimale est la solution $(300,200)$. Pour optimiser son profit, l'entreprise doit donc produire 300 litres de yaourt allégé et 200 litres de yaourt normal.

- R** S'il existe une solution optimale, alors (au moins) un des sommets du domaine réalisable est une solution optimale.

3 Formes générale, standard et canonique

Dans sa forme générale, un problème linéaire peut contenir des contraintes données par des inégalités ($\leq$ ou $\geq$) ou des égalités. De plus, les variables peuvent être positives ou nulles, négatives ou nulles ou sans signe. L'objectif peut être une maximisation ou une minimisation. Le **PL 1** donné en début de chapitre est un exemple de PL de forme générale.

Définition 3 Un PL est dit sous forme *canonique* si :

- l'objectif est une maximisation,
- les contraintes sont toutes des inégalités de type $\leq$,
- les variables sont positives ou nulles.

- R** Tout PL (forme générale) peut être écrit sous forme canonique en appliquant quelques modifications :
- Si une variable x est négative ou nulle, on remplace cette variable par la variable x' de signe contraire (x' vaut $-x$).
 - Si une variable x est *libre* (sans contrainte de signe), on peut alors l'écrire comme la différence de deux variables $x^+ - x^-$ satisfaisant $x^+ \geq 0$ et $x^- \geq 0$.
 - Chaque égalité peut être remplacée par deux inégalités. On peut aussi multiplier une contrainte par -1 pour changer son signe.
 - Minimiser z se ramène à maximiser $-z$.

Le problème linéaire **PL 1** sous forme générale peut s'écrire, sous forme canonique, de la manière suivante :

PL 3

$$\begin{aligned} \max & 2x'_1 + x_2^+ - x_2^- \\ & -x'_1 + x_2^+ - x_2^- \leq 10 \\ & x'_1 - x_2^+ + x_2^- \leq -10 \\ & 5x'_1 - x_2^+ + x_2^- \leq 2 \\ & x'_1, x_2^+, x_2^- \geq 0 \end{aligned}$$

Définition 4 Un problème linéaire est dit sous forme *standard* si :

- l'objectif est une maximisation,
- les contraintes sont toutes des égalités,
- les variables sont positives ou nulles.

- R** On peut facilement passer d'une forme canonique à une forme standard en ajoutant, dans chaque contrainte, une *variable d'écart*, correspondant à la différence entre les valeurs du membre de droite et du membre de gauche.

Le **PL 3** s'écrit sous forme standard :

PL 4

$$\begin{aligned}
&\max 2x'_1 + x_2^+ - x_2^- \\
&-x'_1 + x_2^+ - x_2^- + e_1 = 10 \\
&x'_1 - x_2^+ + x_2^- + e_2 = -10 \\
&5x'_1 - x_2^+ + x_2^- + e_3 = 2 \\
&x'_1, x_2^+, x_2^-, e_1, e_2, e_3 \geq 0
\end{aligned}$$

4 Algorithme du simplexe

Pour résoudre un PL, on va utiliser un algorithme appelé *algorithme du simplexe* et introduit par George Dantzig en 1947. L'idée de cet algorithme est de partir d'une solution, puis de chercher à chaque itération une solution voisine qui améliore la fonction objective, jusqu'à ce que la solution obtenue soit optimale.

On supposera dans cette section que le PL à résoudre est sous forme canonique et que les membres de droite sont positifs ou nuls. La première étape consiste à écrire ce PL sous forme standard en ajoutant les variables d'écart. On a alors une première solution en fixant toutes les variables originales (*i.e.*, celles qui ne sont pas des variables d'écart) à 0 et chaque variable d'écart à la valeur du membre de droite de la contrainte.

À titre d'exemple, considérons le problème linéaire de l'exemple sur les yaourts. Celui-ci est sous forme canonique et les membres de droite sont tous positifs. La transformation de ce PL en forme standard donne :

PL 5

$$\begin{aligned}
&\max 40x_A + 50x_N + 0e_1 + 0e_2 + 0e_3 \\
&2x_A + x_N + e_1 = 800 \\
&x_A + 2x_N + e_2 = 700 \\
&x_N + e_3 = 300 \\
&x_A, x_N, e_1, e_2, e_3 \geq 0
\end{aligned}$$

Une solution du problème est $(0, 0, 800, 700, 300)$ avec un coût (de la fonction objective) de 0.



Si un membre de droite est strictement négatif, alors cette solution n'est pas réalisable puisque les variables d'écart doivent être positives ou nulles. Nous verrons par la suite comment traiter ce cas.

On peut réécrire le PL en mettant à gauche les variables non nulles (e_1 , e_2 et e_3). On obtient ainsi le problème linéaire, appelé *dictionnaire associé* à e_1 , e_2 et e_3 :

Dictionnaire 1

$$\begin{aligned}
&\max 40x_A + 50x_N \\
&e_1 = 800 - 2x_A - x_N \\
&e_2 = 700 - x_A - 2x_N \\
&e_3 = 300 - x_N \\
&x_A, x_N, e_1, e_2, e_3 \geq 0
\end{aligned}$$

Les variables à gauche sont appelées variables *de base* alors que celles à droite sont appelées variables *hors base*. Les variables de base ainsi que la fonction objective sont exprimées en fonction des variables hors base. La solution est donnée en fixant les variables hors base à 0. Les variables de base sont les variables ayant une valeur non nulle². Cette solution est la *solution associée au dictionnaire*.

Il est clair que cette solution n'est pas optimale puisque la valeur de la fonction objective associée est 0. Comme le coefficient associé à la variable hors base x_A dans la fonction objective est strictement positif, si l'on trouve une autre solution avec $x_A > 0$, alors cette solution sera meilleure (puisque son coût sera

2. Certaines variables de base peuvent être nulles dans le cas d'un PL dégénéré.

strictement plus grand que 0). On va donc chercher une solution avec $x_A > 0$ mais en laissant les autres variables hors base (ici x_N) à 0.

R On a choisi x_A de manière arbitraire. On aurait aussi bien pu chercher une solution avec $x_N > 0$ (et x_A à 0).

Quelle valeur donner à x_A ?

- Si l'on fixe $x_A = 100$ (et $x_N = 0$), les valeurs des variables d'écart sont alors $e_1 = 800 - 200 = 600$, $e_2 = 700 - 100 = 600$, et $e_3 = 300$. On obtient alors la solution $(100, 0, 600, 600, 300)$ de coût $z = 4000$.
- Si l'on fixe $x_A = 500$, on obtient alors le point $(500, 0, -200, 200, 300)$ de coût $z = 20000$. Cependant, ce point n'est pas une solution puisque la contrainte $e_1 \geq 0$ n'est pas satisfaite !

Il faut donc trouver la valeur maximale que peut prendre x_A tout en satisfaisant les contraintes. La contrainte $e_1 \geq 0$ implique que $800 - 2x_A \geq 0$, c'est-à-dire $x_A \leq 400$. La contrainte $e_2 \geq 0$ implique que $x_A \leq 700$.

Pour que le nouveau point soit une solution, il faut donc que x_A soit au plus égal à 400. Comme on cherche à maximiser l'objectif, on pose $x_A = 400$. On obtient alors la solution $(400, 0, 0, 300, 300)$ de coût $z = 16000$.

Dans la nouvelle solution, la variable x_A étant non nulle, elle devient une variable de base. On dit que x_A *entre en base*. Comme $x_A = 400$, la variable e_1 passe à 0 et devient une variable hors base. On dit que e_1 *sort de la base*. Il faut maintenant écrire le dictionnaire associé aux variables de base x_A , e_2 et e_3 . L'équation contenant e_1 permet d'exprimer la variable x_A en fonction des variables hors base. De plus, on remplace dans les autres contraintes la variable x_A par la valeur donnée par cette première équation. On obtient alors :

Dictionnaire 2

$$\max 16000 + 30x_N - 20e_1$$

$$x_A = 400 - \frac{1}{2}x_N - \frac{1}{2}e_1$$

$$e_2 = 300 - \frac{3}{2}x_N + \frac{1}{2}e_1$$

$$e_3 = 300 - x_N$$

$$x_A, x_N, e_1, e_2, e_3 \geq 0$$

On recommence une nouvelle itération. Trouver une solution avec $x_N > 0$ améliorerait l'objectif. Pour obtenir une solution valide, on doit satisfaire :

$$x_A = 400 - \frac{1}{2}x_N \geq 0, \text{ ce qui implique } x_N \leq 800$$

$$e_2 = 300 - \frac{3}{2}x_N \geq 0, \text{ ce qui implique } x_N \leq 200$$

$$e_3 = 300 - x_N \geq 0, \text{ ce qui implique } x_N \leq 300$$

On pose donc $x_N = 200$. La variable x_N entre en base et e_2 en sort. On obtient alors la solution $(300, 200, 0, 0, 100)$ de coût $z = 40 * 300 + 50 * 200 = 22000$.

R Si aucune borne supérieure n'est trouvée pour la variable entrante, alors le problème linéaire est non borné (la valeur de la fonction objective tend vers l'infini).

Le nouveau dictionnaire associé est :

Dictionnaire 3

$$\max 22000 - 10e_1 - 20e_2$$

$$x_A = 300 - \frac{2}{3}e_1 + \frac{1}{3}e_2$$

$$x_N = 200 + \frac{1}{3}e_1 - \frac{2}{3}e_2$$

$$e_3 = 100 - \frac{1}{3}e_1 + \frac{2}{3}e_2$$

$$x_A, x_N, e_1, e_2, e_3 \geq 0$$

Toute solution voisine avec $e_1 > 0$ ou $e_2 > 0$ serait de moins bonne qualité puisque les coefficients de la fonction objective associés à e_1 et e_2 sont négatifs. La solution $(300, 200, 0, 0, 100)$ est donc optimale. Le dictionnaire associé à une solution optimale est dit *optimal*.



Un dictionnaire est optimal si tous les coefficients des variables hors-base dans la fonction objective sont négatifs ou nuls.

L'algorithme du simplexe peut donc être écrit de la manière suivante.

Algorithme 1 : Algorithme du simplexe

Entrées : PL canonique avec membres de droite positifs ou nuls

Sorties : Solution optimale ou problème non borné

début

Écrire le PL sous forme standard

$D \leftarrow$ dictionnaire associé aux variables d'écart

tant que D n'est pas optimal **faire**

$x_j \leftarrow$ variable entrante de D

$x_i \leftarrow$ variable sortante de D et x_j

si x_i n'existe pas **alors**

retourner Problème non borné

 Mettre à jour D (x_j entre, x_i sort)

retourner Solution optimale de D

Choix des variables entrante et sortante

À chaque itération, on doit faire entrer en base une variable dont le coefficient dans la fonction objective du dictionnaire est positif. Dans le cas où plusieurs variables peuvent entrer en base, on choisira la première dans l'ordre "lexicographique" (l'ordre dans lequel apparaissent les variables dans le PL). De la même manière, si plusieurs variables peuvent sortir de base, on choisira la première dans l'ordre lexicographique. Ce choix permet de s'assurer que l'algorithme du simplexe s'arrête dans tous les cas³.

Interprétation géométrique

Les solutions associées aux différents dictionnaires obtenus avec l'algorithme du simplexe correspondent à des sommets du domaine réalisable. Ainsi, les solutions obtenues par l'algorithme du simplexe dans l'exemple sont $(0, 0, 800, 700, 300)$, $(400, 0, 0, 300, 300)$ et $(300, 200, 0, 0, 100)$. Ces solutions correspondent aux sommets du domaine réalisable $(0, 0)$, $(400, 0)$ et $(300, 200)$.

L'algorithme du simplexe part d'un sommet du domaine réalisable. À chaque étape, il se déplace vers un sommet voisin améliorant l'objectif.

3. En effet, si le PL est dégénéré, il est possible, en prenant d'autres règles pour le choix des variables entrante et sortante, que l'algorithme du simplexe ne s'arrête jamais. Ceci est cependant très rare et l'explication de la dégénérescence des problèmes linéaires dépasse le cadre de ce cours.

5 Recherche d'une solution initiale

L'algorithme du simplexe a besoin d'une solution initiale (un sommet) pour commencer. Si tous les membres de droite sont positifs ou nuls, alors une solution peut facilement être trouvée (c.f. partie précédente). Dans le cas contraire, la recherche d'une solution initiale n'est pas forcément facile.

Une méthode pour trouver une telle solution initiale consiste à résoudre une version modifiée du PL. On considère une nouvelle variable, disons x_0 , et on ajoute $-x_0$ dans chaque contrainte où le membre de droite est strictement négatif. On ajoute également la contrainte $x_0 \geq 0$ et on pose comme objectif $\min x_0$.

À titre d'exemple, supposons que l'on ait le PL (sous forme standard) :

PL 6

$$\begin{aligned} \max & -x_1 - 2x_2 \\ & -x_1 - x_2 + e_1 = -1 \\ & x_1 + x_2 + e_2 = 2 \\ & -4x_1 - x_2 + e_3 = -4 \\ & x_1, x_2, e_1, e_2, e_3 \geq 0 \end{aligned}$$

Le PL auxiliaire de ce problème est alors :

PL 7

$$\begin{aligned} \max & -x_0 \\ & -x_1 - x_2 + e_1 - x_0 = -1 \\ & x_1 + x_2 + e_2 = 2 \\ & -4x_1 - x_2 + e_3 - x_0 = -4 \\ & x_1, x_2, e_1, e_2, e_3 \geq 0 \end{aligned}$$

Une solution peut facilement être trouvée. On fixe x_0 égal à la valeur absolue du membre de droite le plus petit (ici, 4). On fixe les variables originales à 0. Les valeurs des variables d'écart se calculent alors en fonction des membres de droite et de x_0 . Dans l'exemple, on obtient la solution $(4, 0, 0, 3, 2, 0)$. Les variables de base associées à cette solution sont x_0 , e_1 et e_2 . Comme on a une solution initiale, on peut alors appliquer l'algorithme du simplexe.

On part du dictionnaire donné par les variables en base de la solution initiale trouvée. Dans l'exemple, on part du dictionnaire suivant :

Dictionnaire 4

$$\begin{aligned} \max & -4 + 4x_1 + x_2 - e_3 \\ & e_1 = 3 - 3x_1 + e_3 \\ & e_2 = 2 - x_1 - x_2 \\ & x_0 = 4 - 4x_1 - x_2 + e_3 \\ & x_A, x_N, e_1, e_2, e_3 \geq 0 \end{aligned}$$

Si la solution optimale du problème auxiliaire a pour coût 0 (i.e., x_0 vaut 0 dans cette solution), alors elle est solution du PL original et peut être utilisée comme solution initiale. Dans le cas contraire, le PL original n'a pas de solution.

Cette phase de recherche d'une solution initiale à l'aide de l'algorithme du simplexe est appelée *phase I du simplexe*. La résolution d'un PL à l'aide du simplexe avec recherche d'une solution initiale est appelée dans ce cas *simplexe en deux phases*.

2. Optimisation linéaire en nombres entiers

Certains problèmes ne peuvent pas être modélisés sous forme de problèmes d'optimisation linéaire. En effet, certaines décisions ne peuvent être traduites que par des variables acceptant un nombre restreint de valeurs. Par exemple, lorsqu'une usine souhaite déterminer quels produits fabriquer pour maximiser son gain étant donné un ensemble de ressources, la réponse ne peut pas être 3.7 voitures et 2.1 motos ! Les variables représentant ces décisions ne peuvent alors accepter que des valeurs entières.

1 Définition

Un problème d'optimisation linéaire dont les variables ne peuvent prendre que des valeurs entières est appelé *problème d'optimisation linéaire en nombres entiers (PLNE)*. C'est donc un problème avec une fonction objective linéaire, des contraintes linéaires et des *contraintes d'intégralité* (contraintes indiquant qu'une variable doit prendre des valeurs entières). Voici un exemple de PLNE :

PLNE 1

$$\max 3x_1 + 8x_2 \quad (2.1)$$

$$x_1 + x_2 \leq 3 \quad (2.2)$$

$$5x_1 + 2x_2 \leq 10 \quad (2.3)$$

$$x_1 + 3x_2 \leq 8 \quad (2.4)$$

$$x_1, x_2 \geq 0 \quad (2.5)$$

$$x_1, x_2 \text{ entiers} \quad (2.6)$$

Le problème **PLNE 1** contient deux variables, trois contraintes linéaires et deux contraintes d'intégralité.



Un problème d'optimisation linéaire peut contenir à la fois des variables continues et des variables entières. On parle alors de *problème d'optimisation linéaire en variables mixtes*.

1.1 Interprétation graphique

Lorsqu'un PLNE contient deux variables, il est possible, comme pour un PL, de représenter les solutions dans le plan. Pour chaque contrainte, on trace le demi-plan associé séparant l'ensemble des points satisfaisant la contrainte de ceux qui ne la satisfont pas. L'ensemble des solutions du PLNE est l'ensemble des points à coordonnées entières appartenant à l'intersection des demi-plans des contraintes linéaires.

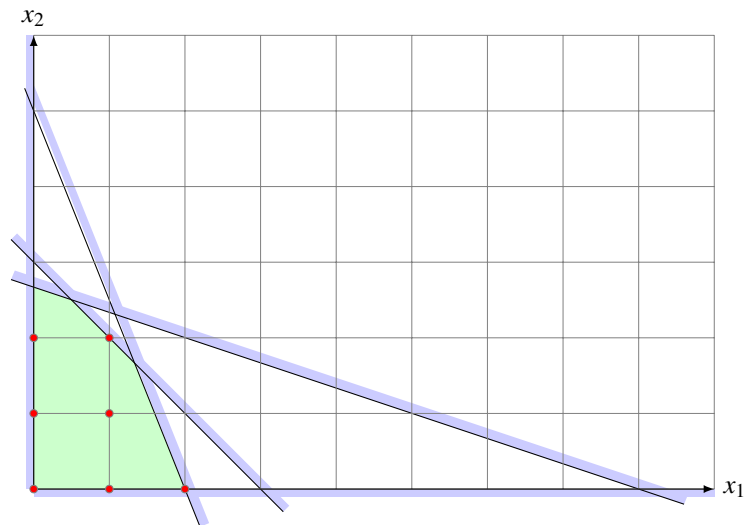
La représentation graphique du **PLNE 1** est donné par la figure 2.1. Dans cette figure, l'ensemble des points (réels) satisfaisant toutes les contraintes est dessiné en vert. Parmi ces points, ceux à coordonnées entières – représentés par les points rouges – correspondent aux solutions du problème **PLNE 1**. Le problème contient donc 7 solutions. La solution optimale est la solution (1,2) de coût 19.

1.2 Différences avec l'optimisation linéaire

Le fait que certaines variables prennent des valeurs entières modifie complètement la structure du problème. On distingue 3 grandes différences avec l'optimisation linéaire.

Nombre fini de solutions

Un PLNE borné contient un nombre fini (mais généralement très grand) de solutions. Il est donc possible d'énumérer les solutions du problème.

FIGURE 2.1 – Solutions du problème **PLNE 1** (en rouge)

Différents systèmes d'inégalités

Puisque les solutions d'un PLNE sont des points entiers, un ensemble de solutions peut être représenté par différents ensembles de contraintes linéaires. Les contraintes du **PLNE 2** définissent le même ensemble de solutions entières que les contraintes du **PLNE 1**. Ces deux PLNE sont donc équivalents.

PLNE 2

$$\begin{aligned} \max & 3x_1 + 8x_2 \\ & 2x_1 + x_2 \leq 4 \\ & x_2 \leq 2 \\ & x_1, x_2 \geq 0 \\ & x_1, x_2 \text{ entiers} \end{aligned}$$

Sommets et solutions

Un sommet du domaine réalisable défini par les contraintes linéaires du PLNE peut ne pas correspondre à une solution si ce n'est pas un point entier. Par exemple, trois sommets du **PLNE 1** sont des points fractionnaires (ayant au moins une coordonnée non entière) et deux sont des points entiers.

Cela implique qu'un PLNE ne peut plus être résolu par l'algorithme du simplexe. En effet, ce dernier, visitant uniquement les sommets du domaine réalisable, peut tomber sur des points qui ne sont pas solutions. De plus, l'algorithme ne "visite" dans ce cas pas les bons points puisque parmi les deux sommets entiers qu'il est susceptible de visiter ((0,0) et (2,0)), le meilleur a un coût de 6, ce qui est très loin de l'optimum égal à 19!

On ne peut donc pas utiliser simplement l'algorithme du simplexe pour résoudre un PLNE, même si le simplexe sera très utile lors de la résolution.

1.3 Énumération explicite

Puisque le nombre de solutions est fini, il est possible de les énumérer (contrairement au cas continu). Un algorithme consiste alors à énumérer toutes les solutions et à retourner la meilleure solution.

Comment énumérer les solutions?

Supposons pour simplifier que le problème contient n variables binaires (soit 0, soit 1) et qu'il n'y a pas d'autres variables (continues ou entières). Une façon d'énumérer les solutions consiste à énumérer tous les points à coordonnées 0 ou 1 et à vérifier si chaque point est une solution du problème.

Efficacité de cet algorithme ?

Pour avoir une idée de l'efficacité de cet algorithme, nous allons déterminer le temps moyen que mettrait cet algorithme en fonction du nombre de points à tester. Supposons que l'on puisse traiter (vérifier si un point est une solution et calculer son coût) 1 milliard de points en une seconde. Voici alors les temps d'exécution de l'algorithme en fonction de différentes valeurs de n .

n	Temps d'exécution de l'algorithme
10	1 microseconde
20	1 milliseconde
30	1 seconde
40	18.3 minutes
50	13 jours
60	36,5 années
70	374,3 siècles

Conclusion : Il est impossible d'utiliser un tel algorithme !

2 Algorithme de séparation et évaluation

L'algorithme de séparation et évaluation (Branch&Bound (B&B) en anglais) est un algorithme générique de résolution de PLNE (et PL mixtes). Il est basé sur une énumération implicite des solutions, c'est-à-dire que l'algorithme a pour objectif de ne pas énumérer toutes les solutions mais cela peut se produire dans le pire des cas !

Pour l'instant, nous considérons l'algorithme de séparation et évaluation lorsque le PLNE à résoudre est un problème à variables binaires. De plus, nous supposons que la fonction objective est à maximiser. L'exploration ou énumération des solutions se fait sous la forme d'un arbre appelé *arbre de branchements*. Cet arbre est défini de la manière suivante :

- la racine correspond au problème initial,
- les deux enfants d'un nœud correspondent au problème parent mais en fixant une variable soit à 1 (premier enfant) soit à 0 (deuxième enfant),
- les feuilles correspondent au problème initial dans lequel toutes les variables ont été fixées. Elles correspondent donc à un point entier qui est une solution du problème s'il vérifie toutes les contraintes linéaires.

Supposons que le PLNE contienne 3 variables binaires x_1 , x_2 et x_3 . L'arbre de branchements peut alors être celui donné par la figure 2.2. Les nœuds $P_7, \dots, P_{14}$ sont les feuilles de l'arbre et correspondent aux 8 points à coordonnées 0/1 dans $\mathbb{R}^3$.

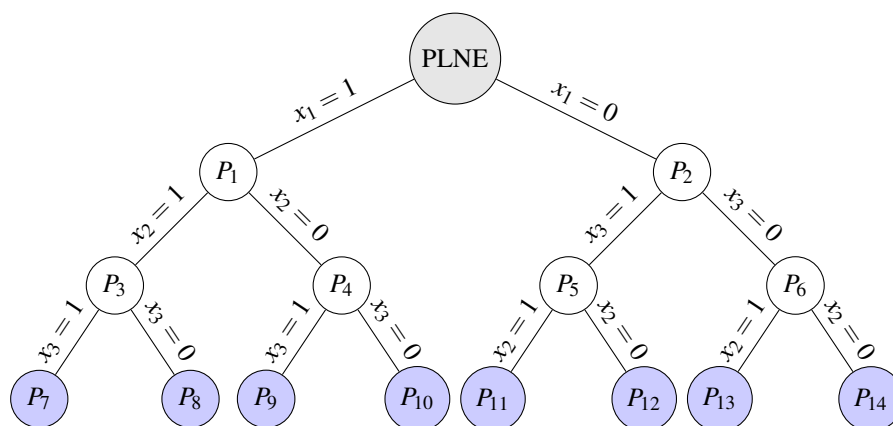


FIGURE 2.2 – Arbre de branchements pour 3 variables

Définition 5 Un nœud de l'arbre de branchements est également appelé *sous-problème*.

Si l'algorithme explore tous les nœuds de l'arbre de branchements (notamment toutes les feuilles), alors cela revient à une exploration totale, ce qui n'est pas possible dès lors que l'on a quelques dizaines de variables. **Le but de l'algorithme de séparation et évaluation est de ne pas explorer tout l'arbre.** Pour cela, il repose sur les deux questions suivantes :

- Quelle est la meilleure solution obtenue pour l'instant ?
- Peut-on espérer obtenir mieux (une meilleure solution) en explorant cette partie de l'arbre ?

À titre d'exemple, imaginons que l'on ait exploré toute la partie gauche de l'arbre (partie $x_1 = 1$) de la figure 2.2. Si, après cette exploration, on sait que l'on ne peut pas espérer mieux dans la deuxième partie de l'arbre (c'est-à-dire que l'on sait que les solutions avec $x_1 = 0$ ne sont pas mieux que celles que l'on a déjà trouvées), alors cette partie n'est pas explorée et l'algorithme s'arrête. On a alors exploré seulement 4 solutions sur 8 (même si l'on a exploré 8 nœuds en comptant la racine) !

2.1 Calcul de bornes

L'algorithme de séparation et évaluation calcule des bornes pour ne pas explorer tout l'arbre. Ces deux bornes permettent de répondre aux deux questions précédentes.

Définition 6 — Bornes primales du PLNE. Toute solution du PLNE a un coût inférieur ou égal à l'optimum (dans le cas d'une maximisation). Son coût constitue donc une borne inférieure sur l'optimum. Cette borne est appelée *borne primale*.

Définition 7 — Relaxation continue d'un sous-problème. Un sous-problème de l'arbre de branchements est un PLNE correspondant au PLNE initial avec en plus des équations fixant la valeur de certaines variables. Si l'on enlève les contraintes d'intégralité dans ce sous-problème, on obtient alors un PL. Ce PL est appelé *relaxation continue* ou *relaxation linéaire* du sous-problème.

La relaxation continue permet d'obtenir une borne supérieure sur le coût de la meilleure solution du sous-problème. En effet, la relaxation continue contient toutes les solutions du sous-problème plus d'autres correspondant aux points fractionnaires satisfaisant les contraintes. La fonction objective (à maximiser) étant la même dans les deux cas, l'optimum de la relaxation continue fournit une borne supérieure sur l'optimum du sous-problème.

Définition 8 — Borne duale d'un sous-problème. L'optimum de la relaxation continue d'un sous-problème P de l'arbre de branchements est appelé *borne duale* de P .

■ **Exemple 1** La relaxation continue du **PLNE 1** correspond au PL donné par (2.1)-(2.5). La solution optimale de cette relaxation est $(\frac{1}{2}, \frac{5}{2})$ et son coût est 21.5. Cette valeur correspond à la borne duale du PLNE. Cette borne est supérieure ou égale à l'optimum du problème qui est 19. ■

■ **Exemple 2** Considérons l'arbre de branchements de la figure 2.2. Si la borne duale du sous-problème P_2 vaut 18.7, cela signifie que les points correspondant à P_{11} , P_{12} , P_{13} et P_{14} (s'ils sont solutions) ne peuvent avoir un coût supérieur à 18.7. De même, si la borne duale du sous-problème P_4 est 8.4, alors le coût des points correspondant à P_9 et P_{10} , s'ils sont solutions, ne peut dépasser 8.4. ■

2.2 Algorithme

L'algorithme de séparation et évaluation considère une liste $\mathcal{L}$ de sous-problèmes à évaluer. La liste $\mathcal{L}$ contient au départ un seul sous-problème correspondant au PLNE à résoudre (racine de l'arbre de branchements). Tant que la liste n'est pas vide, l'algorithme retire un sommet de cette liste pour l'évaluer. Quand la liste est vide, l'algorithme retourne la meilleure solution trouvée. Cette solution est alors une solution optimale du PLNE.

Évaluation d'un sous-problème

L'évaluation d'un sous-problème consiste à calculer sa borne duale et la solution associée (solution optimale de la relaxation continue). Ceci peut être fait en appliquant par exemple l'algorithme du simplexe. Quatre cas de figure sont possibles :

- La solution optimale de la relaxation continue est entière. Dans ce cas, elle correspond à la solution optimale du sous-problème. Il n'est donc pas nécessaire de poursuivre l'exploration de l'arbre à partir

de ce sous-problème.

- La relaxation continue n'admet pas de solution. Dans ce cas, cela signifie que le sous-problème n'a pas non plus de solution. Il n'est donc pas nécessaire de poursuivre l'exploration de l'arbre à partir de ce sous-problème.
- La borne duale est inférieure ou égale à la meilleure borne primale trouvée par l'algorithme. Dans ce cas, cela signifie que l'on ne peut pas espérer trouver une solution dans le sous-problème meilleure que celle que l'on connaît. Il n'est donc pas nécessaire de poursuivre l'exploration de l'arbre à partir de ce sous-problème.
- La borne duale est supérieure à la meilleure borne primale et la solution optimale associée est fractionnaire. Dans ce cas, on ne peut rien dire. Il faut poursuivre l'exploration du sous-arbre associé au sous-problème. On procède alors à une étape de séparation de ce sous-problème.

2.3 Séparation d'un sous-problème

L'étape de séparation consiste à ajouter dans la liste $\mathcal{L}$ deux nouveaux sous-problèmes, disons P_0 et P_1 correspondant aux fils du sous-problème P que l'on considère. Puisque l'on sépare P , cela signifie que la solution optimale $\bar{x} = (\bar{x}_1, \dots, \bar{x}_n)$ de sa relaxation continue est fractionnaire. On choisit une variable $\bar{x}_j$ fractionnaire. Le sous-problème P_0 est le sous-problème P dans lequel on ajoute l'équation $x_j = 0$. Le sous-problème P_1 est le sous-problème obtenu en ajoutant dans P l'équation $x_j = 1$.

2.4 Spécifications de l'algorithme

L'algorithme de séparation et évaluation dépend de plusieurs paramètres. Pour appliquer cet algorithme, il faut déterminer :

- comment choisir la variable fractionnaire lors de la séparation (s'il en existe plusieurs) ?
- Quel sous-problème retirer de la liste $\mathcal{L}$ pour évaluation ?

Ces deux paramètres permettent d'obtenir plusieurs versions de l'algorithme de séparation et évaluation. Par défaut, nous choisirons les règles suivantes :

- S'il existe plusieurs variables fractionnaires pour la séparation, nous choisirons la variable fractionnaire de plus petit indice.
- S'il existe plusieurs sous-problèmes à évaluer, nous choisirons le sous-problème le plus "bas" dans l'arbre (avec le plus de variables fixées). S'il en existe plusieurs, nous choisirons celui contenant le plus de variables fixées à 1. On appelle cette règle un *parcours en profondeur*. Concrètement, cela correspond à considérer que la liste $\mathcal{L}$ est une pile. À chaque séparation, on empile d'abord le sous-problème avec la nouvelle variable fixée à 0.



Il existe des variantes dans lesquelles la borne duale n'est pas donnée par la relaxation continue mais par une autre relaxation.

2.5 Comment faire lorsque les variables sont entières ?

Lorsque les variables sont entières (et non plus binaires), il n'est plus possible, lors de la séparation sur la variable fractionnaire x_j , de créer deux sous-problèmes, l'un avec $x_j = 0$ et l'autre avec $x_j = 1$.

Une technique consiste à créer deux sous-problèmes, l'un en rajoutant $x_j \leq \lfloor \bar{x}_j \rfloor$ et l'autre avec $x_j \geq \lceil \bar{x}_j \rceil$, où $\bar{x}_j$ désigne la valeur fractionnaire de la variable x_j dans la solution optimale de la relaxation continue.