Chapitre 2 : Méthode du Simplexe

J.-F. Scheid, B. Pinçon

v2.1

Plan du chapitre

I. Introduction

On a vu que pour résoudre un PL, il suffit de se restreindre aux solutions de bases réalisables.

Méthode du simplexe due à G. Dantzig (1947).

G. Dantzig, 1914-2005

Les deux phases de la méthode du simplexe :

1. **Phase 1 – Initialisation** : Trouver une solution de base réalisable (ou bien détecter l'impossibilité).
2. **Phase 2 – Progression** : On passe d'un sommet à un sommet voisin *pour augmenter la fonction objectif*

Remarque : On appelle n-simplexe ou simplement **simplexe**, l'enveloppe convexe d'un ensemble de $n+1$ points ($n = 1$: un segment, $n = 2$: un triangle, $n = 3$: un tétraèdre)

II. L'algorithme du simplexe proprement dit : Phase 2

PL sous forme standard

$$\max_{\mathrm{x}\in\mathbb{R}^n}\left[F(\mathrm{x}) = \mathrm{c}^\top \mathrm{x}\right]$$
$$\left\{\begin{array}{l} A\mathrm{x} = \mathrm{b} \\ \mathrm{x} \geq 0 \end{array}\right.$$

On dispose d'une base B et d'une solution de base réalisable $\underline{\mathrm{x}}$ avec (à une permutation près des colonnes de A)

$$A = (A_B \mid A_H) \quad \text{et} \quad \underline{\mathrm{x}} = \begin{pmatrix} \underline{\mathrm{x}}_B \\ \underline{\mathrm{x}}_H \end{pmatrix}$$

où A_B matrice $m \times m$, **inversible** (variables de base)
A_H matrice $m \times (n - m)$ (variables hors-base)

II. L'algorithme du simplexe proprement dit : Phase 2

PL sous forme standard

$$\max_{\mathrm{x}\in\mathbb{R}^n} \left[F(\mathrm{x}) = \mathrm{c}^\top \mathrm{x}\right]$$
$$\begin{cases} A\mathrm{x} = \mathrm{b} \\ \mathrm{x} \geq 0 \end{cases}$$

On dispose d'une base B et d'une solution de base réalisable $\underline{\mathrm{x}}$ avec (à une permutation près des colonnes de A)

$$A = (A_B \mid A_H) \qquad \text{et} \qquad \underline{\mathrm{x}} = \begin{pmatrix} \underline{\mathrm{x}}_B \\ \underline{\mathrm{x}}_H \end{pmatrix}$$

où A_B matrice $m \times m$, **inversible** (variables de base)
A_H matrice $m \times (n-m)$ (variables hors-base)

But : on veut trouver une autre base B^* et une solution de base réalisable $\underline{\mathrm{x}}^*$ telles que $\underline{\mathrm{x}}^*$ est meilleur que $\underline{\mathrm{x}}$ c-à-d

$$F(\underline{\mathrm{x}}^*) > F(\underline{\mathrm{x}})$$

Principe de la méthode du simplexe : faire rentrer une variable hors-base dans la nouvelle base (*variable entrante*) et faire sortir à la place une variable de base (*variable sortante*).

Principe de la méthode du simplexe : faire rentrer une variable hors-base dans la nouvelle base (*variable entrante*) et faire sortir à la place une variable de base (*variable sortante*).

1) Variable entrante - calcul des coûts réduits

Fonction objectif F exprimée en fonction des **variables hors-base**.

Ensemble des solutions réalisables $\mathcal{D}_R = \{\mathrm{x} \in \mathbb{R}^n | A\mathrm{x} = \mathrm{b},\ \mathrm{x} \geq 0\}$.

Exercice :

Montrer que pour tout $\mathrm{x} \in \mathcal{D}_R$, on a :

$$F(\mathrm{x}) = F(\underline{\mathrm{x}}) + \mathrm{L}_H^\top \mathrm{x}_H, \text{ où } \boxed{\mathrm{L}_H^\top = \mathrm{c}_H^\top - \mathrm{c}_B^\top A_B^{-1} A_H}$$

L_H est appelé vecteur des *coûts réduits* (par rapport à la base B).

Aide : dans la fonction objectif, exprimer les variables de base x_B en fonction des variables hors-base.

Solution :.

On a $\mathrm{b} = A\mathrm{x} = A_B\mathrm{x}_B + A_H\mathrm{x}_H$ avec A_B inversible donc $\mathrm{x}_B = A_B^{-1}(\mathrm{b} - A_H\mathrm{x}_H)$. On obtient donc

$$\begin{array}{rcl} F(\mathrm{x}) & = & \mathrm{c}^\top\mathrm{x} = \mathrm{c}_B^\top\mathrm{x}_B + \mathrm{c}_H^\top\mathrm{x}_H \quad \text{avec} \quad \mathrm{c} = \begin{pmatrix} \mathrm{c}_B \\ \mathrm{c}_H \end{pmatrix} \\ & = & \mathrm{c}_B^\top A_B^{-1}(\mathrm{b} - A_H\mathrm{x}_H) + \mathrm{c}_H^\top\mathrm{x}_H \\ & = & \mathrm{c}_B^\top A_B^{-1}\mathrm{b} + (\mathrm{c}_H^\top - \mathrm{c}_B^\top A_B^{-1}A_H)\mathrm{x}_H \end{array}$$

Or $\underline{\mathrm{x}}_B = A_B^{-1}\mathrm{b}$ (car $\underline{\mathrm{x}}_H = 0$) et $\mathrm{c}_B^\top A_B^{-1}\mathrm{b} = \mathrm{c}^\top\underline{\mathrm{x}} = F(\underline{\mathrm{x}})$ donc

$$F(\mathrm{x}) = F(\underline{\mathrm{x}}) + (\mathrm{c}_H^\top - \mathrm{c}_B^\top A_B^{-1}A_H)\mathrm{x}_H.$$

$\square$

Exercice : condition nécessaire d'optimalité

- Montrer que si $L_H \leq 0$ alors $\underline{x}$ (qui est la solution de base (réalisable) associée à la base B) est optimale, i.e. $\forall \mathrm{x} \in \mathcal{D}_R, F(\mathrm{x}) \leq F(\underline{\mathrm{x}})$.
- Montrer que si de plus $L_H < 0$ alors cette solution optimale est unique ($\forall \mathrm{x} \in \mathcal{D}_R$ tel que $x \neq \underline{x}$, on a $F(\mathrm{x}) < F(\underline{\mathrm{x}})$)

Aide : comme $x \neq \underline{x}$ au moins une des composantes de x est nécessairement strictement positive, pourquoi ?

Variable entrante

- Si les coûts réduits sont tous négatifs i.e. $\mathsf{L}_H^\top \leq 0$, il n'est alors pas possible d'augmenter la fonction objectif F: l'algorithme se termine *normalement* c'est-à-dire qu'on a trouvé une solution de base réalisable $\underline{x}$ *optimale*.

Variable entrante

- Si les coûts réduits sont tous négatifs i.e. $\mathsf{L}_H^\top \leq 0$, il n'est alors pas possible d'augmenter la fonction objectif F: l'algorithme se termine *normalement* c'est-à-dire qu'on a trouvé une solution de base réalisable $\underline{\mathrm{x}}$ *optimale*.
- Dans le cas contraire (i.e. $\exists(\mathsf{L}_H)_i > 0$), on a intérêt à faire entrer dans la base, la variable hors-base qui a le coût réduit *positif le plus grand possible*.

Variable entrante

- Si les coûts réduits sont tous négatifs i.e. $\mathsf{L}_H^\top \leq 0$, il n'est alors pas possible d'augmenter la fonction objectif F: l'algorithme se termine *normalement* c'est-à-dire qu'on a trouvé une solution de base réalisable $\underline{\mathsf{x}}$ *optimale*.
- Dans le cas contraire (i.e. $\exists (\mathsf{L}_H)_i > 0$), on a intérêt à faire entrer dans la base, la variable hors-base qui a le coût réduit *positif le plus grand possible*.

On note $e \notin B$ l'indice de la *variable entrante*. On choisit e tel que

$$(\mathsf{L}_H)_e = \max_j \left\{ (\mathsf{L}_H)_j,\ (\mathsf{L}_H)_j > 0 \right\}$$

ce qu'on note par

$$\boxed{e = \operatorname*{argmax}_j \left\{ (\mathsf{L}_H)_j,\ (\mathsf{L}_H)_j > 0 \right\}}$$

Remarque. Si on traite d'un problème de minimisation c'est-à-dire avec

$$\min F(\mathrm{x}),$$

alors la variable entrante x_e est déterminée par l'indice

$$e = \underset{j}{\operatorname{argmin}} \left\{ (\mathrm{L}_H)_j,\ (\mathrm{L}_H)_j < 0 \right\}$$

2) Variable sortante

Une fois l'indice e choisi, il faut déterminer la variable qui doit quitter la base. En maintenant la relation $A\mathrm{x} = \mathrm{b}$ avec $\mathrm{x} \geq 0$, on augmente la variable entrante x_e jusqu'à annuler une des variables de base. Cette variable sera alors la *variable sortante*.

2) Variable sortante

Une fois l'indice e choisi, il faut déterminer la variable qui doit quitter la base. En maintenant la relation $A\mathrm{x} = \mathrm{b}$ avec $\mathrm{x} \geq 0$, on augmente la variable entrante x_e jusqu'à annuler une des variables de base. Cette variable sera alors la *variable sortante*.

$$A\mathrm{x} = \mathrm{b} \quad \Leftrightarrow \quad A_B\mathrm{x}_B + A^e x_e = \mathrm{b} \qquad \text{où } A^e \text{ désigne la } e\text{-ième colonne de } A$$

2) Variable sortante

Une fois l'indice e choisi, il faut déterminer la variable qui doit quitter la base. En maintenant la relation $A\mathrm{x} = \mathrm{b}$ avec $\mathrm{x} \geq 0$, on augmente la variable entrante x_e jusqu'à annuler une des variables de base. Cette variable sera alors la *variable sortante*.

$$\begin{aligned} A\mathrm{x} = \mathrm{b} \quad &\Leftrightarrow \quad A_B \mathrm{x}_B + A^e x_e = \mathrm{b} \qquad \text{où } A^e \text{ désigne la } e\text{-ième colonne de } A \\ &\Leftrightarrow \quad \mathrm{x}_B = A_B^{-1}(\mathrm{b} - A^e x_e) \end{aligned}$$

2) Variable sortante

Une fois l'indice e choisi, il faut déterminer la variable qui doit quitter la base. En maintenant la relation $A\mathrm{x} = \mathrm{b}$ avec $\mathrm{x} \geq 0$, on augmente la variable entrante x_e jusqu'à annuler une des variables de base. Cette variable sera alors la *variable sortante*.

$$\begin{aligned} A\mathrm{x} = \mathrm{b} \quad &\Leftrightarrow \quad A_B\mathrm{x}_B + A^e x_e = \mathrm{b} \qquad \text{où } A^e \text{ désigne la } e\text{-ième colonne de } A \\ &\Leftrightarrow \quad \mathrm{x}_B = A_B^{-1}(\mathrm{b} - A^e x_e) \\ &\Leftrightarrow \quad \mathrm{x}_B = \underline{\mathrm{x}}_B - A_B^{-1}A^e x_e \end{aligned}$$

2) Variable sortante

Une fois l'indice e choisi, il faut déterminer la variable qui doit quitter la base. En maintenant la relation $A\mathrm{x} = \mathrm{b}$ avec $\mathrm{x} \geq 0$, on augmente la variable entrante x_e jusqu'à annuler une des variables de base. Cette variable sera alors la *variable sortante*.

$$
\begin{aligned}
A\mathrm{x} = \mathrm{b} \quad &\Leftrightarrow \quad A_B \mathrm{x}_B + A^e x_e = \mathrm{b} \qquad \text{où } A^e \text{ désigne la } e\text{-ième colonne de } A \\
&\Leftrightarrow \quad \mathrm{x}_B = A_B^{-1}(\mathrm{b} - A^e x_e) \\
&\Leftrightarrow \quad \mathrm{x}_B = \underline{\mathrm{x}}_B - A_B^{-1} A^e x_e \\
&\Leftrightarrow \quad \mathrm{x}_B = \underline{\mathrm{x}}_B - \mathrm{z} x_e
\end{aligned}
$$

avec

$$\boxed{\mathrm{z} = A_B^{-1} A^e \in \mathbb{R}^m.}$$

On doit avoir : $\mathrm{x}_B = \underline{\mathrm{x}}_B - \mathrm{z}x_e \geq 0$

On doit avoir : $\quad x_B = \underline{x}_B - z x_e \geq 0$

- Si $z \leq 0$, on peut augmenter x_e autant qu'on veut, on aura toujours la positivité de la variable de base x_B. La fonction objectif n'est pas majorée sur $\mathcal{D}_R$ ($\max F = +\infty$) $\Rightarrow$ arrêt de l'algorithme.

On doit avoir : $\mathrm{x}_B = \underline{\mathrm{x}}_B - \mathrm{z}x_e \geq 0$

- Si $\mathrm{z} \leq 0$, on peut augmenter x_e autant qu'on veut, on aura toujours la positivité de la variable de base x_B. La fonction objectif n'est pas majorée sur $\mathcal{D}_R$ ($\max F = +\infty$) $\Rightarrow$ arrêt de l'algorithme.
- Sinon (i.e. il existe $z_i > 0$), pour avoir la positivité $(\underline{\mathrm{x}}_B)_i - z_i x_e \geq 0$ pour tout i, on choisit la **variable sortante x_s** pour laquelle le rapport $(\underline{\mathrm{x}}_B)_i / z_i$ pour $i = 1, \cdots, m$ avec $z_i > 0$, est **le plus petit possible** :

On doit avoir : $\quad \mathrm{x}_B = \underline{\mathrm{x}}_B - \mathrm{z}x_e \geq 0$

- Si $\mathrm{z} \leq 0$, on peut augmenter x_e autant qu'on veut, on aura toujours la positivité de la variable de base x_B. La fonction objectif n'est pas majorée sur $\mathcal{D}_R$ ($\max F = +\infty$) $\Rightarrow$ arrêt de l'algorithme.

- Sinon (i.e. il existe $z_i > 0$), pour avoir la positivité $(\underline{\mathrm{x}}_B)_i - z_i x_e \geq 0$ pour tout i, on choisit la **variable sortante x_s** pour laquelle le rapport $(\underline{\mathrm{x}}_B)_i / z_i$ pour $i = 1, \cdots, m$ avec $z_i > 0$, est **le plus petit possible** :

Variable sortante (indice) :

$$\boxed{s = \underset{i}{\operatorname{argmin}} \left\{ \frac{(\underline{\mathrm{x}}_B)_i}{z_i},\ z_i > 0 \right\}}$$

On a, dans ce cas, $x_s = 0$ et $\mathrm{x}_B \geq 0$.

Remarque. La valeur de la variable entrante est donnée par

$$\boxed{x_e = \min_i \left\{ \frac{(\underline{x}_B)_i}{z_i},\ z_i > 0 \right\}}$$

Méthode du simplexe en phase 2 (progression)

1.
   - *Calcul des variables de base réalisables* :
     Etant donné $A = (A_B \mid A_H)$, on calcule $\underline{\mathrm{x}}_B = A_B^{-1}\mathrm{b} \geq 0$.
   - *Calcul des coûts réduits* :
     $\mathrm{L}_H^\top = \mathrm{c}_H^\top - \mathrm{c}_B^\top A_B^{-1} A_H \qquad \left(F(\mathrm{x}) = F(\underline{\mathrm{x}}) + \mathrm{L}_H^\top \mathrm{x}_H\right)$
   - Si $\mathrm{L}_H \leq 0$ alors $\underline{\mathrm{x}}_B$ est une solution optimale ($\rightarrow$ arrêt de l'algo.).

Méthode du simplexe en phase 2 (progression)

1. - *Calcul des variables de base réalisables* :
     Etant donné $A = (A_B \mid A_H)$, on calcule $\underline{x}_B = A_B^{-1} b \geq 0$.
   - *Calcul des coûts réduits* :
     $L_H^\top = c_H^\top - c_B^\top A_B^{-1} A_H \qquad \left(F(x) = F(\underline{x}) + L_H^\top x_H\right)$
   - Si $L_H \leq 0$ alors $\underline{x}_B$ est une solution optimale ($\rightarrow$ arrêt de l'algo.).
2. *variable entrante* : $e = \mathrm{argmax}_j \left\{ (L_H)_j,\ (L_H)_j > 0 \right\}$

Méthode du simplexe en phase 2 (progression)

1. - *Calcul des variables de base réalisables* :
     Etant donné $A = (A_B \mid A_H)$, on calcule $\underline{x}_B = A_B^{-1} b \geq 0$.
   - *Calcul des coûts réduits* :
     $L_H^\top = c_H^\top - c_B^\top A_B^{-1} A_H \qquad \left(F(x) = F(\underline{x}) + L_H^\top x_H\right)$
   - Si $L_H \leq 0$ alors $\underline{x}_B$ est une solution optimale ($\rightarrow$ arrêt de l'algo.).
2. *variable entrante* : $e = \operatorname{argmax}_j \left\{ (L_H)_j,\ (L_H)_j > 0 \right\}$
3. *variable sortante* :
   - Calcul de $z = A_B^{-1} A^e$ puis
   - $s = \operatorname{argmin}_i \left\{ \frac{(\underline{x}_B)_i}{z_i},\ z_i > 0 \right\}$.

Méthode du simplexe en phase 2 (progression)

1. - *Calcul des variables de base réalisables* :
     Etant donné $A = (A_B \mid A_H)$, on calcule $\underline{x}_B = A_B^{-1} b \geq 0$.
   - *Calcul des coûts réduits* :
     $L_H^\top = c_H^\top - c_B^\top A_B^{-1} A_H \qquad \left(F(x) = F(\underline{x}) + L_H^\top x_H\right)$
   - Si $L_H \leq 0$ alors $\underline{x}_B$ est une solution optimale ($\rightarrow$ arrêt de l'algo.).
2. *variable entrante* : $e = \operatorname{argmax}_j \left\{ (L_H)_j,\ (L_H)_j > 0 \right\}$
3. *variable sortante* : • Calcul de $z = A_B^{-1} A^e$ puis
   • $s = \operatorname{argmin}_i \left\{ \dfrac{(\underline{x}_B)_i}{z_i},\ z_i > 0 \right\}$.
4. On obtient une nouvelle base $\widetilde{B}$ et une nouvelle matrice $A_{\widetilde{B}}$ dans laquelle la colonne A^e remplace la colonne A^s. Calcul de $A_{\widetilde{B}}^{-1}$ et retour en 1.

III. Méthode des dictionnaires

PL sous forme standard $\max_{\mathrm{x}} F(\mathrm{x}) = \mathrm{c}^\top \mathrm{x}$

$$\begin{cases} A\mathrm{x} = \mathrm{b} \\ \mathrm{x} \geq 0 \end{cases}$$

Principe: on exprime les variables de base x_B ainsi que F en fonction des variables hors-base x_H. On obtient un système linéaire qu'on appelle **dictionnaire**.

III. Méthode des dictionnaires

PL sous forme standard $\max_{\mathrm{x}} F(\mathrm{x}) = \mathrm{c}^\top \mathrm{x}$

$$\begin{cases} A\mathrm{x} = \mathrm{b} \\ \mathrm{x} \geq 0 \end{cases}$$

Principe: on exprime les variables de base x_B ainsi que F en fonction des variables hors-base x_H. On obtient un système linéaire qu'on appelle **dictionnaire**.

Exemple du problème de production.

Forme standard (variables d'écart e_1, e_2, e_3)

$$\max F(x_1, x_2) = 6x_1 + 4x_2$$
$$\begin{cases} 3x_1 + 9x_2 + e1 = 81 \\ 4x_1 + 5x_2 + e2 = 55 \\ 2x_1 + x_2 + e3 = 20 \\ x_1, x_2 \geq 0,\ e_1, e_2, e_3 \geq 0 \end{cases}$$

⋆ Etape 1.

Solution de base réalisable initiale :

$$x_1 = 0,\ x_2 = 0,\ e_1 = 81,\ e_2 = 55,\ e_3 = 20 \text{ avec } F = 0.$$

Dictionnaire: On exprime les variables de base e_1, e_2, e_3 en fonction des variables hors-base x_1, x_2.

$$\boxed{\begin{array}{l} e_1 = 81 - 3x_1 - 9x_2 \\ e_2 = 55 - 4x_1 - 5x_2 \\ e_3 = 20 - 2x_1 - x_2 \\ \hline F = 6x_1 + 4x_2 \end{array}}$$

⋆ Etape 1.

Solution de base réalisable initiale :

$$x_1 = 0,\ x_2 = 0,\ e_1 = 81,\ e_2 = 55,\ e_3 = 20 \text{ avec } F = 0.$$

Dictionnaire: On exprime les variables de base e_1, e_2, e_3 en fonction des variables hors-base x_1, x_2.

$$\begin{array}{|l|}
\hline
e_1 = 81 - 3x_1 - 9x_2 \\
e_2 = 55 - 4x_1 - 5x_2 \\
e_3 = 20 - 2x_1 - x_2 \\
\hline
F = 6x_1 + 4x_2 \\
\hline
\end{array}$$

Variable entrante x_e: $\max_{>0}\{6,\ 4\} = 6 \Rightarrow \boxed{x_e = x_1}$.

⋆ Etape 1.

Solution de base réalisable initiale :

$$x_1 = 0,\ x_2 = 0,\ e_1 = 81,\ e_2 = 55,\ e_3 = 20 \text{ avec } F = 0.$$

Dictionnaire: On exprime les variables de base e_1, e_2, e_3 en fonction des variables hors-base x_1, x_2.

$$\boxed{\begin{array}{l} e_1 = 81 - 3x_1 - 9x_2 \\ e_2 = 55 - 4x_1 - 5x_2 \\ e_3 = 20 - 2x_1 - x_2 \\ \hline F = 6x_1 + 4x_2 \end{array}}$$

Variable entrante x_e: $\max_{>0}\{6,\ 4\} = 6 \Rightarrow \boxed{x_e = x_1}$.

Variable sortante x_s: on maintient $e_1 \geq 0,\ e_2 \geq 0,\ e_3 \geq 0$

$$\Rightarrow x_1 = \min_{>0}\{\frac{81}{3}, \frac{55}{4}, \frac{20}{2}\} = 10 \Rightarrow \boxed{x_s = e_3}.$$

⋆ Etape 1.

Solution de base réalisable initiale :

$$x_1 = 0,\ x_2 = 0,\ e_1 = 81,\ e_2 = 55,\ e_3 = 20 \text{ avec } F = 0.$$

Dictionnaire: On exprime les variables de base e_1, e_2, e_3 en fonction des variables hors-base x_1, x_2.

$$\begin{array}{|l|}
\hline
e_1 = 81 - 3x_1 - 9x_2 \\
e_2 = 55 - 4x_1 - 5x_2 \\
e_3 = 20 - 2x_1 - x_2 \\
\hline
F = 6x_1 + 4x_2 \\
\hline
\end{array}$$

Variable entrante x_e: $\max_{>0}\{6,\ 4\} = 6 \Rightarrow \boxed{x_e = x_1}$.

Variable sortante x_s: on maintient $e_1 \geq 0,\ e_2 \geq 0,\ e_3 \geq 0$

$$\Rightarrow x_1 = \min_{>0}\{\frac{81}{3}, \frac{55}{4}, \frac{20}{2}\} = 10 \Rightarrow \boxed{x_s = e_3}.$$

Nouvelle Solution de base réalisable :

$$x_1 = 10,\ x_2 = 0,\ e_1 = 51,\ e_2 = 15,\ e_3 = 0 \text{ avec } F = 60.$$

⋆ Etape 2.

Dictionnaire: On exprime la nouvelle variable de base x_1 en fonction de x_2 et e_3 (nouvelle variable hors-base). On utilise la 3ème équation du dictionnaire de l'étape 1 et on substitue x_1 dans les autres relations.

$$\begin{array}{l} x_1 = 10 - \frac{1}{2}x_2 - \frac{1}{2}e_3 \\ e_1 = 81 - 3(10 - \frac{1}{2}x_2 - \frac{1}{2}e_3) - 9x_2 \\ e_2 = 55 - 4(10 - \frac{1}{2}x_2 - \frac{1}{2}e_3) - 5x_2 \\ \hline F = 6(10 - \frac{1}{2}x_2 - \frac{1}{2}e_3) + 4x_2 \end{array}$$

On obtient ainsi le dictionnaire (étape 2)

$$
\begin{array}{|l|}
\hline
x_1 = 10 - \frac{1}{2}x_2 - \frac{1}{2}e_3 \\
e_1 = 51 - \frac{15}{2}x_2 + \frac{3}{2}e_3 \\
e_2 = 15 - 3x_2 + 2e_3 \\
\hline
F = 60 + x_2 - 3e_3 \\
\hline
\end{array}
$$

On obtient ainsi le dictionnaire (étape 2)

$$\boxed{\begin{array}{l} x_1 = 10 - \frac{1}{2}x_2 - \frac{1}{2}e_3 \\ e_1 = 51 - \frac{15}{2}x_2 + \frac{3}{2}e_3 \\ e_2 = 15 - 3x_2 + 2e_3 \\ \hline F = 60 + x_2 - 3e_3 \end{array}}$$

Variable entrante x_e: $\max_{>0}\{1, \ -3\} = 1 \Rightarrow \boxed{x_e = x_2}$.

On obtient ainsi le dictionnaire (étape 2)

$$\boxed{\begin{array}{l} x_1 = 10 - \frac{1}{2}x_2 - \frac{1}{2}e_3 \\ e_1 = 51 - \frac{15}{2}x_2 + \frac{3}{2}e_3 \\ e_2 = 15 - 3x_2 + 2e_3 \\ \hline F = 60 + x_2 - 3e_3 \end{array}}$$

Variable entrante x_e: $\max\limits_{>0}\{1,\ -3\} = 1 \Rightarrow \boxed{x_e = x_2}$.

Variable sortante x_s: on maintient $x_1 \geq 0,\ e_1 \geq 0,\ e_2 \geq 0$

$$\Rightarrow x_2 = \min_{>0}\{\frac{10}{1/2}, \frac{51}{15/2}, \frac{15}{3}\} = 5 \Rightarrow \boxed{x_s = e_2}.$$

On obtient ainsi le dictionnaire (étape 2)

$$\boxed{\begin{array}{l} x_1 = 10 - \frac{1}{2}x_2 - \frac{1}{2}e_3 \\ e_1 = 51 - \frac{15}{2}x_2 + \frac{3}{2}e_3 \\ e_2 = 15 - 3x_2 + 2e_3 \\ \hline F = 60 + x_2 - 3e_3 \end{array}}$$

Variable entrante x_e: $\max\limits_{>0}\{1,\ -3\} = 1 \Rightarrow \boxed{x_e = x_2}$.

Variable sortante x_s: on maintient $x_1 \geq 0,\ e_1 \geq 0,\ e_2 \geq 0$

$$\Rightarrow x_2 = \min_{>0}\{\frac{10}{1/2}, \frac{51}{15/2}, \frac{15}{3}\} = 5 \Rightarrow \boxed{x_s = e_2}.$$

Nouvelle Solution de base réalisable (étape 2) :

$$x_1 = \frac{15}{2},\ x_2 = 5,\ e_1 = \frac{27}{2},\ e_2 = 0,\ e_3 = 0 \text{ avec } F = 65.$$

⋆ Etape 3.

Dictionnaire: On exprime la nouvelle variable de base x_2 en fonction des variables hors-base e_2 et e_3. On utilise la 3ème équation du dictionnaire de l'étape 2 et on substitue x_2 dans les autres relations.

$$\boxed{\begin{array}{l} x_2 = 5 - \frac{1}{3}e_2 + \frac{2}{3}e_3 \\ x_1 = \frac{15}{2} + \frac{1}{6}e_2 - \frac{5}{6}e_3 \\ e_1 = \frac{27}{2} + \frac{5}{2}e_2 - \frac{7}{2}e_3 \\ \hline F = 65 - \frac{1}{3}e_2 - \frac{7}{3}e_3 \end{array}}$$

⋆ Etape 3.

Dictionnaire: On exprime la nouvelle variable de base x_2 en fonction des variables hors-base e_2 et e_3. On utilise la 3ème équation du dictionnaire de l'étape 2 et on substitue x_2 dans les autres relations.

$$\boxed{\begin{array}{l} x_2 = 5 - \frac{1}{3}e_2 + \frac{2}{3}e_3 \\ x_1 = \frac{15}{2} + \frac{1}{6}e_2 - \frac{5}{6}e_3 \\ e_1 = \frac{27}{2} + \frac{5}{2}e_2 - \frac{7}{2}e_3 \\ \hline F = 65 - \frac{1}{3}e_2 - \frac{7}{3}e_3 \end{array}}$$

Tous les coûts réduits sont ≤ 0 donc on ne peut plus augmenter F : l'optimum est atteint et la solution optimale est

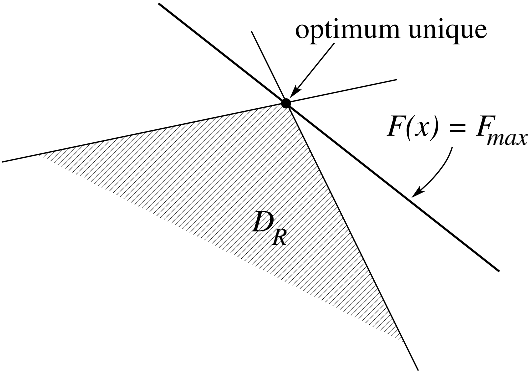
$$x_1^* = \frac{15}{2},\ x_2^* = 5,\ e_1^* = \frac{27}{2},\ e_2^* = 0,\ e_3^* = 0 \text{ avec } \max F = 65.$$

IV. Finitude du simplexe

A chaque étape de l'algorithme du simplexe (en phase 2), il y a des cas remarquables qui conduisent tous à l'arrêt de l'algorithme.

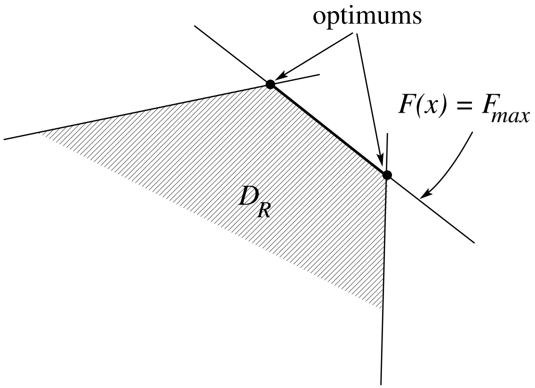
A chaque étape de l'algorithme du simplexe (en phase 2), il y a des cas remarquables qui conduisent tous à l'arrêt de l'algorithme.

1. Si les coûts réduits $L_H < 0$, alors la solution de base réalisable courante est l'unique optimum.

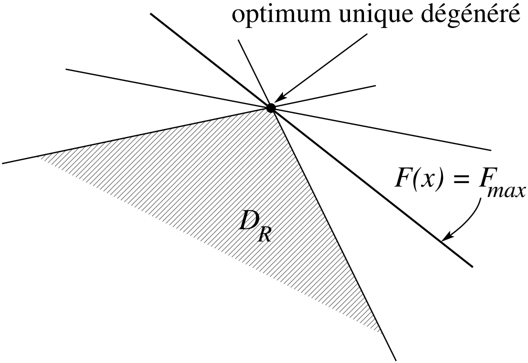


2. Si les coûts réduits $L_H \leq 0$, alors il y a deux cas remarquables :

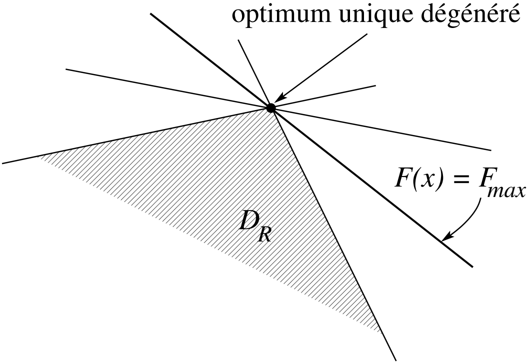
   i) si $(L_H)_e = 0$ et $x_e > 0$, alors l'optimum n'est pas unique.



ii) si $(L_H)_e = 0$ et $x_e = 0$, alors l'optimum est unique (a priori). Dans ce cas, la base est dite **dégénérée** c'est-à-dire qu'il existe une variable de base *nulle*.



ii) si $(L_H)_e = 0$ et $x_e = 0$, alors l'optimum est unique (a priori). Dans ce cas, la base est dite **dégénérée** c'est-à-dire qu'il existe une variable de base *nulle*.



3. Si $(L_H)_e > 0$ et x_e est non borné alors la fonction objectif F n'est pas majorée.

Finitude du simplexe

Une solution de base réalisable est dite **dégénérée** si au moins une des variables de base est *nulle*.

Théorème

Si au cours de l'algorithme du simplexe, aucune base rencontrée n'est dégénérée, alors l'algorithme se termine en un nombre fini d'itérations.

Finitude du simplexe

Une solution de base réalisable est dite **dégénérée** si au moins une des variables de base est *nulle*.

Théorème

Si au cours de l'algorithme du simplexe, aucune base rencontrée n'est dégénérée, alors l'algorithme se termine en un nombre fini d'itérations.

Démonstration. A une itération donnée de l'algorithme :

- soit on détecte une fonction objectif non majorée ($\rightarrow$ arrêt de l'algo.),
- soit elle est strictement croissante car $\widetilde{F}_{opt} - F_{opt} = (\mathrm{L}_H)_e x_e > 0$ puisque $(L_H)_e > 0$ et $x_e > 0$ (par hypothèse, aucune base rencontrée n'est dégénérée).

Par conséquent, on ne rencontre jamais une base déjà rencontrée à une itération précédente. Le nombre de solution de base réalisable étant fini ($\leq C_n^m$), l'algorithme s'arrête nécessairement en un nombre fini d'itérations. $\square$

Remarque: S'il existe une base dégénérée, alors on peut rencontrer un éventuel *cyclage* de l'algorithme : on retrouve une base déjà rencontrée et on boucle indéfiniment.
Pour traiter les cas de dégénérescence, on peut appliquer la règle de Bland (1977) qui assure l'arrêt de l'algorithme en un nombre fini d'itérations.

Règle de Bland

Lorsque plusieurs variables sont susceptibles d'entrer ou de sortir de la base, on choisit toujours celle qui a l'indice le plus petit.

1) Introduction

Pour un PL sous forme *canonique pure* avec les contraintes

$$Ax \leq b, \; x \geq 0,$$

on peut déterminer facilement une solution de base réalisable <u>dans le cas où $b \geq 0$</u>. En effet, sous forme standard les contraintes deviennent $Ax + e = b$, avec $x, e \geq 0$ où e sont les *variables d'écarts*.

Une solution de base réalisable évidente dans ce cas, est

$$x = 0, \quad e = b \geq 0.$$

1) Introduction

Pour un PL sous forme *canonique pure* avec les contraintes

$$Ax \leq b, \; x \geq 0,$$

on peut déterminer facilement une solution de base réalisable dans le cas où $b \geq 0$. En effet, sous forme standard les contraintes deviennent $Ax + e = b$, avec $x, e \geq 0$ où e sont les *variables d'écarts*.

Une solution de base réalisable évidente dans ce cas, est

$$x = 0, \quad e = b \geq 0.$$

Mais pour un PL *sous forme standard*, il n'y a pas toujours de solution de base réalisable évidente.

Construction des solutions de base réalisable = phase d'initialisation du simplexe (phase 1).

2) Variables auxiliaires

PL sous forme standard

$$(PL) \quad \begin{cases} \max\limits_{\mathrm{x} \in \mathcal{D}_{\mathcal{R}}} \left[F(\mathrm{x}) = \mathrm{c}^\top \mathrm{x} \right] \\ \mathcal{D}_R = \{x \in \mathbb{R}^n : Ax = b, x \geq 0\} \end{cases}$$

On ne suppose pas que la matrice $A \in \mathcal{M}_{m \times n}$ est de rang plein, ni qu'il existe bien des solutions réalisables.

Dans la suite on supposera que $b \geq 0$ sans perte de généralité (si $b_i < 0$, on multiplie la contrainte d'égalité correspondante par -1).

2) Variables auxiliaires

PL sous forme standard

$$(PL) \quad \begin{cases} \max\limits_{\mathrm{x} \in \mathcal{D}_{\mathcal{R}}} \left[F(\mathrm{x}) = \mathrm{c}^\top \mathrm{x} \right] \\ \mathcal{D}_R = \{x \in \mathbb{R}^n : Ax = b, x \geq 0\} \end{cases}$$

On ne suppose pas que la matrice $A \in \mathcal{M}_{m \times n}$ est de rang plein, ni qu'il existe bien des solutions réalisables.

Dans la suite on supposera que $b \geq 0$ sans perte de généralité (si $b_i < 0$, on multiplie la contrainte d'égalité correspondante par -1).

Pour obtenir une solution de base réalisable ou bien pour détecter l'impossibilité, on introduit un problème de programmation linéaire **auxiliaire** pour des variables supplémentaires appelées **variables artificielles**.

Programme auxiliaire

Le programme auxiliaire associé à (PL) s'écrit

$$(PLA) \quad \begin{cases} \min\limits_{(\mathrm{x},\mathrm{a})} F_A(x,a) = \sum\limits_{i=1}^{m} a_i \\ A\mathrm{x} + \mathrm{a} = \mathrm{b} \\ \mathrm{x} \geq 0, \quad \mathrm{a} \geq 0 \end{cases}$$

où $\mathrm{a} = (a_1, \cdots, a_m)$ sont appelées **variables artificielles**.

Il est clair que :

1. (PLA) dispose d'une base réalisable gratuite $(B = \{a_1, \ldots, a_m\}, H = \{x_1, \ldots, x_n\})$, la solution associée étant $(x := 0, a := b \geq 0)$.
2. Sa fonction objectif est minorée par 0 sur le domaine : $\forall (x, a)$ réalisable pour (PLA), on a $F_A(x, a) = \sum_i a_i \geq 0$.

Il est donc bien posé (domaine non vide et fonction objectif minorée $\Rightarrow$ il admet une solution optimale) et on peut démarrer le simplexe.

On a la propriété suivante.

Proposition

(PL) admet une solution réalisable si et seulement si le problème auxiliaire (PLA) associé admet une solution optimale avec $a = 0$.

On a la propriété suivante.

Proposition

(PL) admet une solution réalisable si et seulement si le problème auxiliaire (PLA) associé admet une solution optimale avec a $= 0$.

Preuve :

sens $\Rightarrow$ Supposons que (PL) admette une solution réalisable $\underline{x}$ (c.a.d. telle que $\underline{x} \geq 0$ et $A\underline{x} = b$). Alors $(\underline{x}, \underline{a} := 0)$ est optimale pour (PLA). En effet, elle est bien dans le domaine (elle vérifie $A\underline{x} + \underline{a} = b$, $\underline{x} \geq 0$ et $\underline{a} \geq 0$) et est telle que $F_A(\underline{x}, \underline{a}) = 0$ (or $F_A(x, a) \geq 0$ pour toute solution (x, a) réalisable pour (PLA)).

sens $\Leftarrow$ Si (PLA) admet une solution optimale $(\bar{x}, \bar{a})$ avec $\bar{a} = 0$ (et donc telle que $F_A(\bar{x}, \bar{a}) = 0$ par définition de F_A) alors on a bien trouvé $\bar{x}$ vérifiant $A\bar{x} = b$ et $\bar{x} \geq 0$, soit une solution réalisable pour (PL).

Détermination d'une solution de base réalisable via le problème auxiliaire :

On applique l'algorithme du simplexe au problème auxiliaire (PLA). Ce problème étant bien posé, on obtient à la fin des itérations de simplexe (en évitant d'éventuels cyclages avec la règle de *Bland*) une solution de base réalisable optimale. Appelons (x^*, a^*) cette solution. Deux possibilités :

1. le coût minimal est non nul $F_A(x^*, a^*) > 0$, on a alors détecté l'impossibilité ($\mathcal{D}_R = \emptyset$) ;
2. le coût minimal est nul, ainsi $a^* = 0$.

Dans ce dernier cas si x^* est bien une solution réalisable pour (PL), on a pas forcément obtenu tout de suite une solution de base réalisable. Il y a deux cas possibles :

1. On a réussi à faire sortir toutes les variables artificielles, c'est gagné.
2. Il reste une ou des variables artificielles dans la base, il reste un peu de travail à faire !

Dans le premier cas les variables hors-base sont formées des m variables artificielles a_i et de $n - m$ variables x_i (on note H les indices de ces dernières) et la base contient m variables x_i (on note B les indices de ces dernières). La décomposition base/hors base de $Ax + a = b$ s'écrit :

$$\sum_{i \in B} A^i x_i + \sum_{i \in H} A^i x_i + \sum_{i=1}^{m} e^i a_i = b \iff A_B x_B = b - A_H x_H - a$$

la partie haute du dictionnaire correspondant est donc :

$$x_B = (A_B)^{-1} b - (A_B)^{-1} A_H x_H - (A_B)^{-1} a$$

La solution de base associée est $(\bar{x}_B = (A_B)^{-1} b, (\bar{x}_H = 0, \bar{a} = 0))$. Il est clair que B est une base réalisable pour (PL), la solution de base associée étant $(\bar{x}_B = (A_B)^{-1} b, \bar{x}_H = 0)$ et, en rayant la partie $-(A_B)^{-1} a$, on a la partie haute du dictionnaire pour démarrer le simplexe sur (PL). Il y a un petit travail pour obtenir la partie basse du dictionnaire (il faut exprimer le coût $F(x) = c_B^\top x_B + c_H^\top x_H$ uniquement en fonction des variables hors-base).

Dans le deuxième cas, une ou plusieurs variables artificielles sont restées en base (la solution de base correspondante est dégénérée). Pour chacune de ces variables artificielles, la ligne correspondante du dictionnaire final s'écrit[1] :

$$a_i = 0 + \sum_{j \in H} m_{i,j} x_j + \sum_{j \in H'} m_{i,j} a_j$$

où H et H' sont respectivement les indices des variables initiales et des variables artificielles qui sont hors-base.

Deux possibilités :

1. s'il existe $j' \in H$ avec $m_{i,j'} \neq 0$, on peut faire entrer la variable $x_{j'}$ en base et a_i en hors-base.
2. sinon, on peut montrer que la ligne i du système initial $Ax = b$, c'est à dire l'équation i ($A_i x = b_i$) est redondante (elle est combinaison linéaire d'autres lignes), on peut la supprimer.

On revient ainsi au cas 1 avec éventuellement moins d'équations.

[1] On peut toujours s'arranger pour que l'équation donnant a_i soit la i-ème équation.

Résumé de la phase d'initialisation du simplexe (phase 1)

À la fin du simplexe phase 1[a] soit (x^*, a^*) la solution de base réalisation optimale et $F_A^* := F_A(x^*, a^*) = \sum_i a_i^*$ alors :

1. Si $F_A^* > 0$ il n'y a pas de solution réalisable pour (PL) $(\mathcal{D}_R = \emptyset)$.
2. Si $F_A^* = 0$ et $\nexists a_j \in X_B$ où X_B désigne l'ensemble des variables de base pour (PLA), alors fin normale de la phase 1. On passe à la phase 2 du simplexe.
3. Si $F_A^* = 0$ et $\exists a_j \in X_B$ avec $a_j = 0$, alors on revient au cas précédent via les manipulations expliquées page précédente[b].

---

[a] Rmq : (PLA) étant un pb de minimisation, la condition d'optimalité s'écrit $L_H \geq 0$.

[b] Pour chaque variable a_j en base, on essaie de la faire passer en hors-base, i.e. côté droit du dictionnaire, en l'échangeant avec une variable x_i en hors-base qui elle passe côté gauche, et si ce c'est pas possible, on supprime l'équation j qui est une contrainte redondante.

Remarques importantes pour une phase 1

1. selon le système initial $Ax = b$ on est pas obligé d'utiliser m variables auxiliaires. Par exemple si une équation provient d'une inégalité transformée en égalité grâce à une variable d'écart, et si cette variable d'écart est positive lorsqu'on met les variables initiales à 0, inutile d'ajouter une variable artificielle à cette équation). Voir exemple suivant.
2. Dès qu'une variable artificielle sort de la base, on peut très bien la rayer et ne plus la considérer (elle ne peut pas revenir en base). Cela simplifie les calculs !

Exercice

simplexe phase 1

On considère le PL suivant :

$$
(P) : \begin{array}{l}
\max_{x\in\mathbb{R}^5} f(x) = x_1 + 2x_2 + x_3 + 0x_4 + 0x_5 \\
x \geq 0 \\
\begin{array}{rcrcrcrcrcl}
x_1 & + & x_2 & - & x_3 & & & & & = & 4 \\
2x_1 & - & x_2 & & & + & x_4 & & & = & 5 \\
-x_1 & + & 2x_2 & & & & & + & x_5 & = & 5
\end{array}
\end{array}
$$

Trouver une solution de base réalisable pour (P) en utilisant un simplexe phase 1 (il suffit de ne rajouter qu'une seule variable auxiliaire). Puis écrire le dictionnaire de départ pour (P).

VI. Complexité du simplexe

Complexité = nombre d'itération dans le simplexe (phase2).

- On peut construire des exemples avec une complexité **exponentielle** en $\mathcal{O}(2^n)$ itérations (Klee-Minty, 1972).
- Mais dans la pratique la complexité du simplexe croît peu avec le nombre n de variables. En pratique, le nombre d'itérations est proportionnel au nombre m de contraintes (de m à $3m$ itérations).
- Si on tient compte de la résolution des systèmes linéaires avec une formule de mise à jour de l'inverse (Shermann-Morrison), on a $\mathcal{O}(m^2)$ opérations pour l'inverse.

VII. Quelques solveurs de PL

- COIN-OR (COmputational INfrastructure for Operations Research) Optimization Suite, logiciel OpenSource.
- GUROBI Code commercial, licence éducation gratuite.
- CPLEX (IBM ILOG). Code commercial, licence éducation gratuite.
- FICO Xpress Optimization Suite
- GLPK (GNU Linear Programming Kit), logiciel open source (GPL).
- AMPL (Automatic Mathematical Programming Language). Plateforme commerciale, version 'bridée' gratuite (limitation du nb de variables).

Comment utiliser ces solveurs :

- Via un code (C, C++,...) définition du PL puis appel au solveur.
- Via un fichier texte décrivant le PL. Formats usuels : MPS (historique mais illisible) et CPLEX/LP (voir un exemple ci-après).
- Via une interface avec un langage de script (python, matlab, ...).

Le format CPLEX/LP sur un exemple.

Soit à résoudre le PL suivant : $\begin{cases} \min f(x_1, x_2) = -x_1 - x_2 \\ 2x_1 - x_2 \leq 2 \\ -x_1 + x_2 \leq 2 \\ x_1 + 2x_2 \geq 5 \\ 5x_1 + 2x_2 \geq 10 \\ x \geq 0 \end{cases}$

```
\ format CPLEX/LP du PL précédent (tout texte après \ est un commentaire)

Minimize                  \ la fct à minimiser
   f : - x1 - x2
Subject To                \ les contraintes
   c1 : 2 x1 - x2 <= 2
   c2 : - x1 + x2 <= 2
   c3 : x1 + 2 x2 >= 5
   c4 : 5 x1 + 2 x2 >= 10
Bounds                    \ les bornes sur les variables
   x1 >= 0
   x2 >= 0
End
```

Quelques explications I

En fait comme par défaut les variables sont supposées être continue et positives et que nous n'avons pas besoin de nommer la fonction et les contraintes, le fichier pourrait être écrit de cette façon :

```
\ format CPLEX/LP du PL précédent en plus court

Minimize
   - x1 - x2
Subject To
   2 x1 - x2 <= 2
   - x1 + x2 <= 2
     x1 + 2 x2 >= 5
   5 x1 + 2 x2 >= 10
End
```

1. Il faut commencer par décrire la fonction (mot clé `Minimize` ou `Maximize`)
2. Puis les contraintes liant les variables (contraintes $\leq$, $\geq$ ou $=$) avec le mot clé `Subject To`.

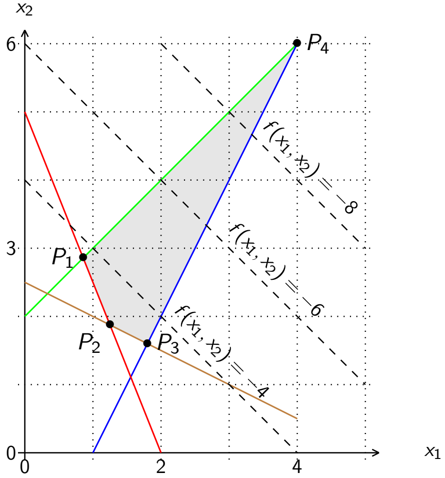
Quelques explications II

3. Ensuite on écrit les bornes sur les variables. Si rien n'est mentionné au sujet d'une variable on considère qu'elle doit être positive. Si `x` est le nom d'une variable et `l`, `u`, `t` des constantes numériques, les possibilités sont les suivantes :
   - `x >= l` ou `l <= x` pour une borne inférieure ;
   - `x <= u` pour une borne supérieure ;
   - `l <= x <= u` pour spécifier en même temps une borne inférieure et une borne supérieure ;
   - `x = t` pour forcer une variable à être égale à une constante (ce n'est plus vraiment une variable) ;
   - `x free` pour une variable sans contrainte.
4. Si certaines variables sont entières, on les nomme dans une section `General`, par exemple si les variables précédentes sont entières :

```
General     \ précise quelles sont la ou les variables entières
   x1
   x2
```

5. Pour préciser les variables binaires, le bon mot clé est `Binary`.

Résolution de l'exemple (graphique).



Résolution de l'exemple (solveur et fichier CPLEX LP).

Supposons que le fichier décrivant le PL en format CPLEX LP s'appelle `exemple.lp`. Il faut :

- rechercher le nom de l'exécutable (attention avec `pip install gurobipy` vous n'avez pas d'exécutable proprement dit installé) ;
- rechercher comment utiliser cet exécutable (internet ou encore `/path/to/executable --help`).

Avec gurobi (l'exécutable correspondant s'appelle `gurobi_cl`) :

```
gurobi_cl ResultFile=exemple.sol exemple.lp
```

Le fichier résultat :

```
# Solution for model f
# Objective value = -10
x1 4
x2 6
```