

Programmation Mathématique Avancée

L'optimisation quadratique

Amélie Lambert

Cnam

MPRO

1 Introduction

- Les classes de problèmes non-linéaires
- Les formes quadratiques
- L'optimisation quadratique

2 L'optimisation quadratique convexe

- Les conditions d'optimalité
- Différence avec la résolution d'un PLNE
- Cas d'une fonction quadratique convexe sous des contraintes linéaires

3 L'optimisation quadratique non convexe

- Reformulations linéaires de produits de variables binaires
- Reformulation linéaire de produits mixtes
- Reformulation linéaire de produits de variables entières
- Relaxation linéaire de produits de variables continues

1 Introduction

- Les classes de problèmes non-linéaires
- Les formes quadratiques
- L'optimisation quadratique

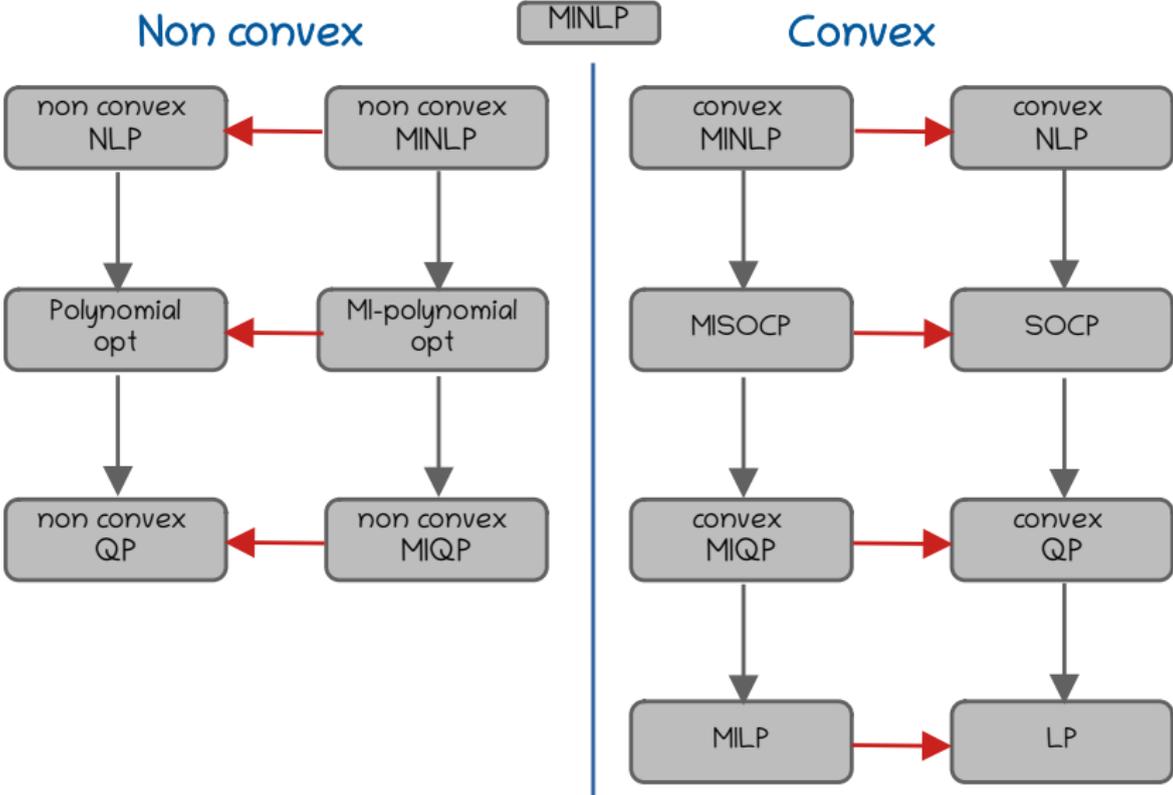
2 L'optimisation quadratique convexe

- Les conditions d'optimalité
- Différence avec la résolution d'un PLNE
- Cas d'une fonction quadratique convexe sous des contraintes linéaires

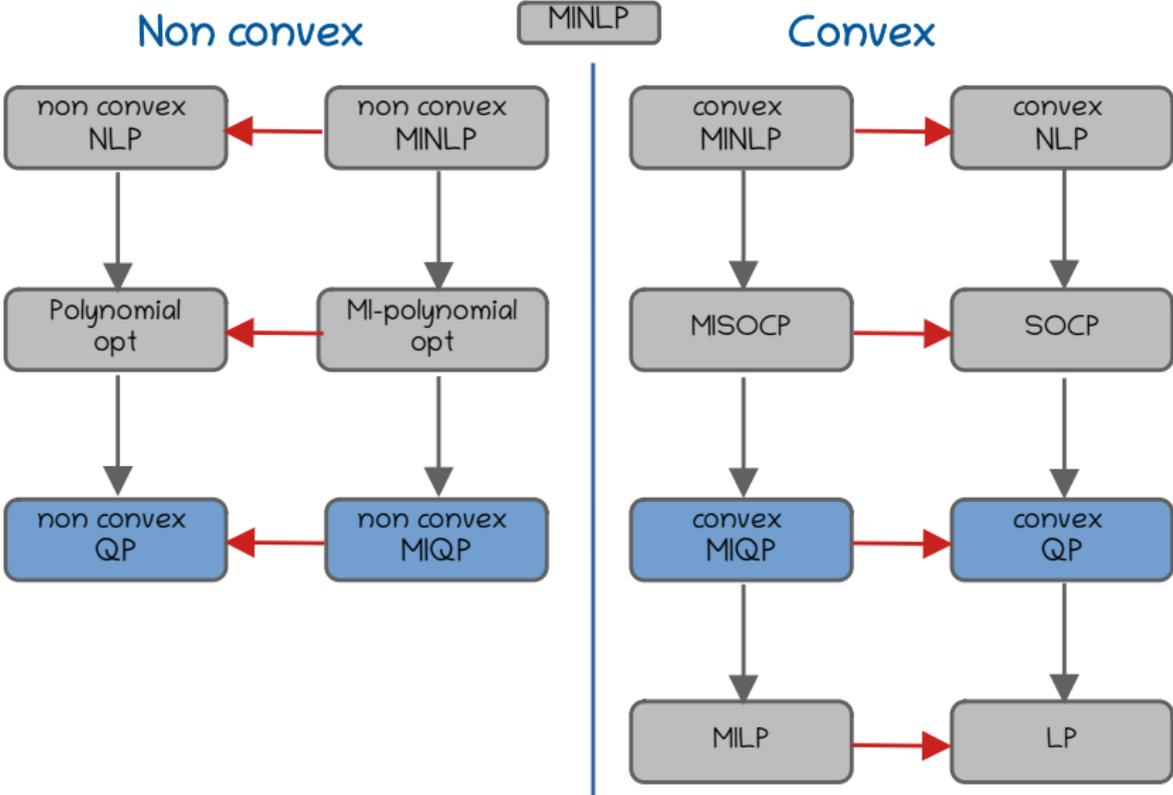
3 L'optimisation quadratique non convexe

- Reformulations linéaires de produits de variables binaires
- Reformulation linéaire de produits mixtes
- Reformulation linéaire de produits de variables entières
- Relaxation linéaire de produits de variables continues

Le graphe des MINLPs



Le graphe des MINLPs



Les formes quadratiques

Les formes quadratiques : définition

Définition : On appelle *forme quadratique* une expression de la forme

$$f(x) = \sum_{i=1}^n \sum_{j=1}^n q_{ij} x_i x_j$$

avec $x = (x_1, x_2, \dots, x_n)^T \in \mathbb{R}^n$.

Notation : Une forme quadratique peut s'écrire sous la forme matricielle :

$$f(x) = \sum_{i=1}^n \sum_{j=1}^n q_{ij} x_i x_j = x^T Q x = \langle Q, x x^T \rangle$$

On dit que *la matrice Q représente la forme quadratique $f(x)$* .

Les formes quadratiques : représentation

Pour $f(x)$, il peut exister plusieurs matrices Q qui la représentent.

Exemple : La forme quadratique $x_1^2 + 4x_1x_2 + 3x_2^2$ peut s'écrire par exemple de plusieurs façons :

- $x_1^2 + 2x_1x_2 + 2x_2x_1 + 3x_2^2$, avec $Q_1 = \begin{pmatrix} 1 & 2 \\ 2 & 3 \end{pmatrix}$
- $x_1^2 + 4x_1x_2 + 0x_2x_1 + 3x_2^2$, avec $Q_2 = \begin{pmatrix} 1 & 4 \\ 0 & 3 \end{pmatrix}$
- $x_1^2 + 6x_1x_2 - 2x_2x_1 + 3x_2^2$, avec $Q_3 = \begin{pmatrix} 1 & 6 \\ -2 & 3 \end{pmatrix}$

Propriété : Une forme quadratique peut toujours être représentée par une matrice symétrique, i.e. $f(x) = \frac{1}{2}x^T(Q + Q^T)x$.
Cette représentation par une matrice symétrique est unique.

On considère ici la représentation de $f(x)$ par une matrice symétrique.

Matrices semidéfinie positives

On note \mathcal{S}^n l'ensemble des matrices symétriques de dimension n .

Soit une matrice $X \in \mathcal{S}^n$, les assertions suivantes sont équivalentes :

- ① X est **semidéfinie positive (SDP)**, noté $X \succeq 0$, si

$$\forall x \in \mathbb{R}^n, x^T X x = \sum_{i=1}^n \sum_{j=1}^n X_{ij} x_i x_j = \langle X, x x^T \rangle \geq 0$$

- ② La plus petite valeur propre de X est positive ou nulle.
- ③ Tous les mineurs symétriques (déterminants des sous-matrices carrées) de X sont positifs ou nuls.
- ④ Décomposition de Cholesky : Pour $k \geq 1$, il existe une matrice triangulaire inférieure $L \in \mathbb{R}^{n \times k}$ tel que $X = LL^T$

Matrices semidéfinie positives : exemple

$$X = \begin{pmatrix} 3 & 1 & -1 \\ 1 & 3 & 1 \\ -1 & 1 & 1 \end{pmatrix}$$

② Les valeurs propres de X sont $(0, 4, 3)$.

③ Mineurs symétriques :

▶ Mineur symétrique d'ordre 3 :

$$\begin{vmatrix} 3 & 1 & -1 \\ 1 & 3 & 1 \\ -1 & 1 & 1 \end{vmatrix} = 3 \begin{vmatrix} 3 & 1 \\ 1 & 1 \end{vmatrix} - 1 \begin{vmatrix} 1 & 1 \\ -1 & 1 \end{vmatrix} - 1 \begin{vmatrix} 1 & 3 \\ -1 & 1 \end{vmatrix} = 0 \geq 0$$

▶ Mineurs symétriques d'ordre 2 :

$$\begin{vmatrix} 3 & 1 \\ 1 & 3 \end{vmatrix} = 8 \geq 0, \quad \begin{vmatrix} 3 & -1 \\ -1 & 1 \end{vmatrix} = 2 \geq 0, \quad \begin{vmatrix} 3 & 1 \\ 1 & 1 \end{vmatrix} = 2 \geq 0$$

▶ Mineurs symétriques d'ordre 1 : $|3| = 3 \geq 0$, $|3| = 3 \geq 0$, $|1| = 1 \geq 0$.

\implies La matrice X est semidéfinie positive.

Matrices semidéfinie positives : exemple

$$X = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 2 & 4 \\ 0 & 4 & 8 \end{pmatrix}$$

④ Décompositions de Choleski non unique :

$$\blacktriangleright \begin{pmatrix} 0 & 0 & 0 \\ 0 & 2 & 4 \\ 0 & 4 & 8 \end{pmatrix} = LL^T = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \sqrt{2} & 0 \\ 0 & \frac{4}{\sqrt{2}} & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 \\ 0 & \sqrt{2} & \frac{4}{\sqrt{2}} \\ 0 & 0 & 0 \end{pmatrix}$$

$$\blacktriangleright \begin{pmatrix} 0 & 0 & 0 \\ 0 & 2 & 4 \\ 0 & 4 & 8 \end{pmatrix} = L'L'^T = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 2 & 2 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 2 \\ 0 & 1 & 2 \\ 0 & 0 & 0 \end{pmatrix}$$

\implies La matrice X est semidéfinie positive.

L'optimisation quadratique

Formulation générale des programmes quadratiques

$$(P) \begin{cases} \min f_0(x) = \langle Q_0, xx^T \rangle + c_0^T x \\ \text{s.t. } f_r(x) = \langle Q_r, xx^T \rangle + c_r^T x - e_r \leq 0 & (r \in \mathcal{R}) \\ 0 \leq \ell_i \leq x_i \leq u_i & (i \in \mathcal{I}) \\ x_i \in \mathbb{N} (i \in \mathcal{J}), \quad x_i \in \mathbb{R} (i \in \mathcal{I} \setminus \mathcal{J}) \end{cases}$$

- $\mathcal{I} = \{1, \dots, N\}$ et $\mathcal{J} = \{1, \dots, n\}$: N variables dont n entières.
- $\mathcal{R} = \{1, \dots, m\}$: m contraintes quadratiques
Remarque : si $Q_r = 0_n \forall r$, elles sont linéaires.
(on note 0_n l'ensemble des matrice nulles de dimension n)

1 Introduction

- Les classes de problèmes non-linéaires
- Les formes quadratiques
- L'optimisation quadratique

2 L'optimisation quadratique convexe

- Les conditions d'optimalité
- Différence avec la résolution d'un PLNE
- Cas d'une fonction quadratique convexe sous des contraintes linéaires

3 L'optimisation quadratique non convexe

- Reformulations linéaires de produits de variables binaires
- Reformulation linéaire de produits mixtes
- Reformulation linéaire de produits de variables entières
- Relaxation linéaire de produits de variables continues

Les conditions d'optimalité

Cas : optimisation sans contrainte

$$(P) \begin{cases} \min f(x) = \langle Q, xx^T \rangle + c^T x \\ \text{s.t. } \ell_i \leq x_i \leq u_i \quad (i \in \mathcal{I}) \end{cases}$$

Propriété : On dit qu'une forme quadratique f est *convexe* lorsque sa matrice symétrique associée est semidéfinie positive.

Définition : f possède un *minimum local* u si

$$\exists \delta > 0 \text{ tel que } f(u) \leq f(x), \forall x, \|x - u\| \leq \delta$$

Définition : f possède un *minimum global* u si

$$\forall x, f(u) \leq f(x)$$

Conditions nécessaires suffisantes d'optimalité globale

Soient $f : \mathbb{R}^n \rightarrow \mathbb{R}$ deux fois différentiable sur \mathbb{R}^n , et u un minimum local de f , si f est *convexe*, alors u est un minimum global de f .

Cas : optimisation sous contraintes d'inégalité

$$(P) \begin{cases} \min f_0(x) = \langle Q_0, xx^T \rangle + c_0^T x \\ \text{s.t. } f_r(x) = \langle Q_r, xx^T \rangle + c_r^T x - e_r \leq 0 & (r \in \mathcal{R}) \\ 0 \leq \ell_i \leq x_i \leq u_i & (i \in \mathcal{I}) \end{cases}$$

Hypothèses : les fonctions f_r sont différentiables et convexes.

On introduit le Lagrangien associé au problème (P), en associant :

- 1 multiplicateur λ_i à chaque contrainte $\ell_i - x_i \leq 0$, $x_i - u_i \leq 0$,
- 1 multiplicateur λ_{r+2N} à chaque contrainte $f_r(x) \leq 0$.

et pour $(x, \lambda) \in \mathbb{R}^N \times \mathbb{R}^{2N+m}$ la fonction :

$$L(x, \lambda) = f(x) + \sum_{r=1}^m \lambda_{r+2N} f_r(x) + \sum_{i=1}^N \lambda_i (\ell_i - x_i) + \sum_{i=1}^N \lambda_{i+N} (x_i - u_i)$$

et si $\lambda_i \geq 0$, alors le lagrangien $L(x, \lambda)$ est aussi une fonction convexe.
(puisque qu'une somme de fonctions convexes est convexe)

Conditions de Karush-Kuhn-Tucker du premier ordre

Théorème (KKT) S'il existe un λ^* tel que :

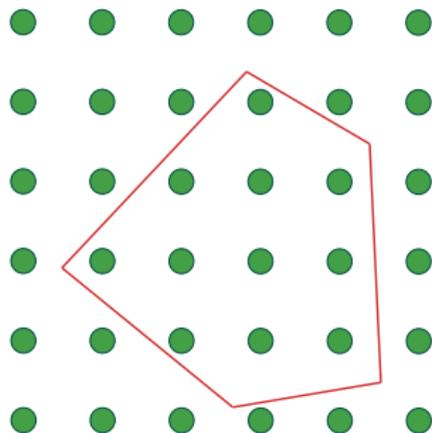
$$\left\{ \begin{array}{l} \nabla_x L(x, \lambda) = \nabla f(x) + \sum_{r=1}^m \lambda_{r+2N}^* \nabla f_r(x) - \sum_{i=1}^N (\lambda_i^* - \lambda_{i+N}^*) = 0 \\ \lambda_{2N+r}^* f_r(x^*) = 0 \quad r = 1, \dots, m \\ f_r(x^*) \leq 0 \quad r = 1, \dots, m \\ \lambda_{2N+r}^* \geq 0 \quad r = 1, \dots, m \\ \lambda_i^* (l_j - x_i) = 0 \quad i = 1, \dots, N \\ \lambda_{i+N}^* (x_i - u_i) = 0 \quad i = 1, \dots, N \\ l_i \leq x_i \leq u_i \quad i = 1, \dots, 2N \\ \lambda_i^* \geq 0 \quad i = 1, \dots, 2N \end{array} \right.$$

Alors x^* est une solution optimale globale du problème (P).

Différence avec la résolution d'un PLNE

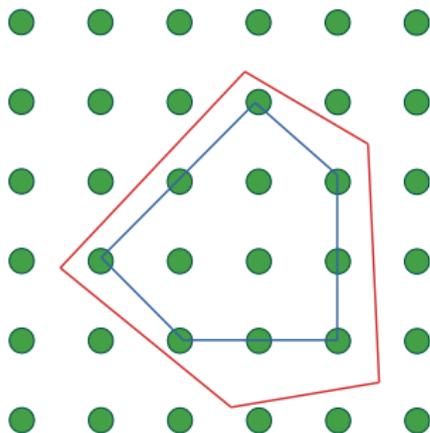
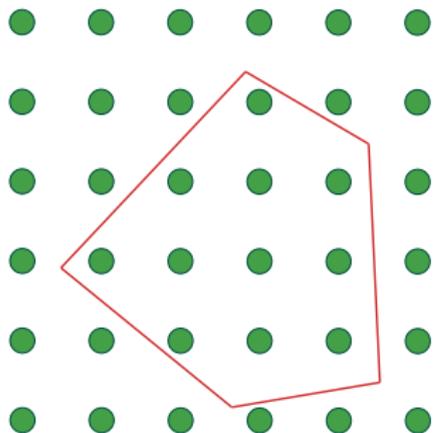
Différence avec la résolution des MILPs

- La solution optimale d'un LP est un sommet de son enveloppe convexe



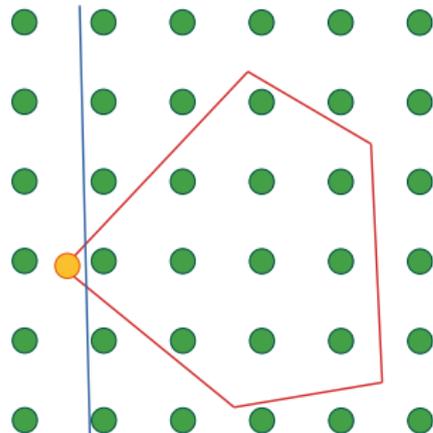
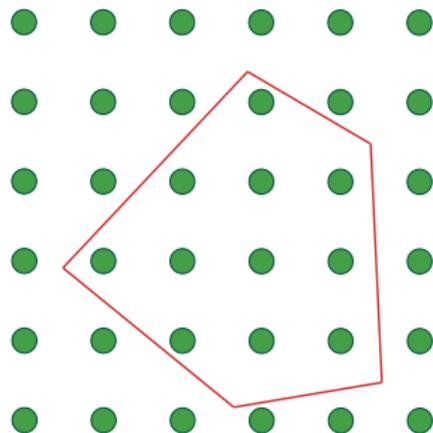
Différence avec la résolution des MILPs

- La solution optimale d'un LP est un sommet de son enveloppe convexe
- La résolution d'un MILP peut-être calculée en résolvant un LP sur l'enveloppe convexe de ses points entiers.



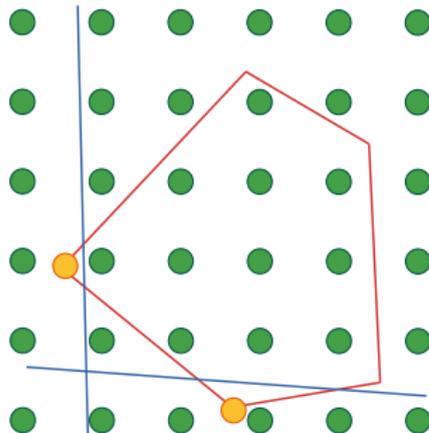
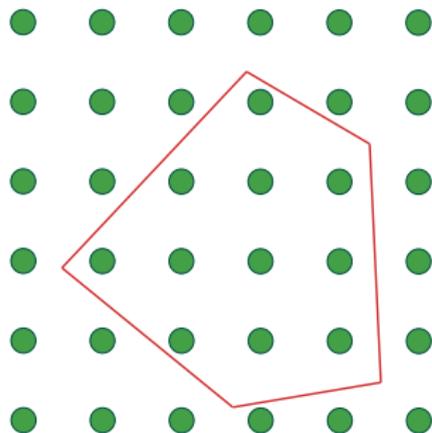
Différence avec la résolution des MILPs

- La solution optimale d'un LP est un sommet de son enveloppe convexe
- La résolution d'un MILP peut-être calculée en résolvant un LP sur l'enveloppe convexe de ses points entiers.
- Stratégie : on sépare les optimaux continus successifs



Différence avec la résolution des MILPs

- La solution optimale d'un LP est un sommet de son enveloppe convexe
- La résolution d'un MILP peut-être calculée en résolvant un LP sur l'enveloppe convexe de ses points entiers.
- Stratégie : on sépare les optimaux continus successifs

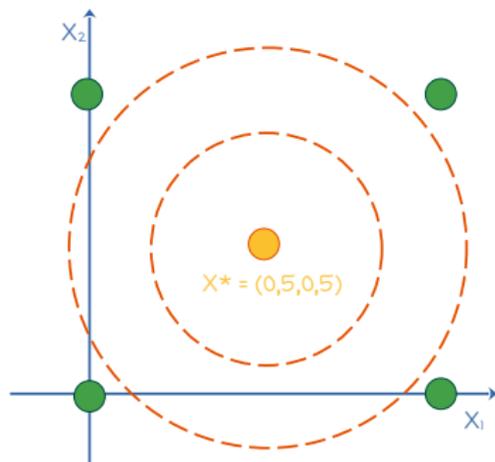


Résolution d'un MIQP convexe : exemple

$$(Ex) \begin{cases} \min & \sum_{i=1}^n \left(x_i - \frac{1}{2}\right)^2 \\ \text{s.t.} & \\ & x \in \{0, 1\}^n \end{cases}$$

Résolution d'un MIQP convexe : exemple

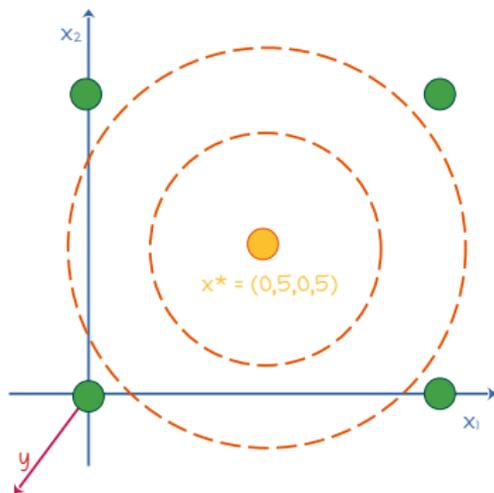
$$(Ex) \begin{cases} \min & \sum_{i=1}^n \left(x_i - \frac{1}{2}\right)^2 \\ \text{s.t.} & x \in \{0, 1\}^n \end{cases}$$



- Solution optimale continue :
 $x = (0.5, 0.5)$
- Ce n'est pas un point extrême de l'ensemble relâché (il est dans l'intérieur stricte)
- Il ne peut pas être séparé !

Résolution d'un MIQP convexe : exemple

$$(Ex) \begin{cases} \min & \sum_{i=1}^n \left(x_i - \frac{1}{2}\right)^2 \\ \text{s.t.} & x \in \{0, 1\}^n \end{cases}$$



$$(Ex') \begin{cases} \min & y \\ \text{s.t.} & y \geq \sum_{i=1}^n \left(x_i - \frac{1}{2}\right)^2 \\ & x \in \{0, 1\}^n, y \in \mathbb{R} \end{cases}$$

(Ex') équivalent à (Ex) et ses solutions optimales continues sont sur la frontière.

Minimisation d'une fonction quadratique convexe sous des contraintes linéaires

Cas particulier des contraintes linéaires

$$(QP) \left\{ \begin{array}{l} \min \quad \sum_i \sum_j q_{ij} x_i x_j + \sum_i c_i x_i \\ \text{s.t.} \quad a_r^T x \leq b_r \quad r \in \mathcal{R} \\ \quad \quad -x_i \leq 0 \quad i \in \mathcal{I} \end{array} \right.$$

Algorithme de résolution par une variante de l'algorithme du simplexe.

Cas particulier des contraintes linéaires

$$(QP) \begin{cases} \min & \sum_i \sum_j q_{ij} x_i x_j + \sum_i c_i x_i \\ \text{s.t.} & a_r^T x \leq b_r \quad r \in \mathcal{R} \quad \leftarrow \lambda_{n+r} \\ & -x_i \leq 0 \quad i \in \mathcal{I} \quad \leftarrow \lambda_i \end{cases}$$

1. Introduction du Lagrangien

$$L(x, \lambda) = \sum_{\substack{j=1 \\ j \neq i}}^n q_{ij} x_j x_j + \sum_i c_i x_i + \sum_{r=1}^m \lambda_{n+r} \left(\sum_i a_{ri} x_i - b_r \right) - \lambda_i x_i$$

Cas particulier des contraintes linéaires

$$(QP) \begin{cases} \min & \sum_i \sum_j q_{ij} x_i x_j + \sum_i c_i x_i \\ \text{s.t.} & a_r^T x \leq b_r & r \in \mathcal{R} \leftarrow \lambda_{n+r} \\ & -x_i \leq 0 & i \in \mathcal{I} \leftarrow \lambda_i \end{cases}$$

2. Écriture des conditions KKT d'optimalité :

$$\begin{cases} -\sum_{\substack{j=1 \\ j \neq i}}^n q_{ij} x_j - 2q_{ii} x_i - c_i - \sum_{r=1}^m \lambda_{n+r} a_{ri} + \lambda_i = 0 & i \in \mathcal{I} \\ a_r^T x \leq b_r & r \in \mathcal{R} \\ (a_r^T x - b_r) \lambda_{n+r} = 0 & r \in \mathcal{R} \\ \lambda_i x_i = 0 & i \in \mathcal{I} \\ x_i \geq 0 & i \in \mathcal{I} \\ \lambda_i \geq 0 & i \in \mathcal{I} \cup \mathcal{R} \end{cases}$$

Cas particulier des contraintes linéaires

$$(QP) \begin{cases} \min & \sum_i \sum_j q_{ij} x_i x_j + \sum_i c_i x_i \\ \text{s.t.} & a_r^T x \leq b_r & r \in \mathcal{R} \leftarrow \lambda_{n+r} \\ & -x_i \leq 0 & i \in \mathcal{I} \leftarrow \lambda_i \end{cases}$$

3. En introduisant les variables d'écart x_{n+r} :

$$\begin{cases} -\sum_{\substack{j=1 \\ j \neq i}}^n q_{ij} x_j - 2q_{ii} x_i - c_i - \sum_{r=1}^m \lambda_{n+r} a_{ri} + \lambda_i = 0 & i \in \mathcal{I} \\ a_r^T x + x_{n+r} = b_r & r \in \mathcal{R} \\ (a_r^T x - b_r) \lambda_{n+r} = 0 & r \in \mathcal{R} \\ \lambda_i x_i = 0 & i \in \mathcal{I} \\ x_i \geq 0 & i \in \mathcal{I} \cup \mathcal{R} \\ \lambda_i \geq 0 & i \in \mathcal{I} \cup \mathcal{R} \end{cases}$$

Cas particulier des contraintes linéaires

$$(QP) \begin{cases} \min & \sum_i \sum_j q_{ij} x_i x_j + \sum_i c_i x_i \\ \text{s.t.} & a_r^T x \leq b_r & r \in \mathcal{R} \leftarrow \lambda_{n+r} \\ & -x_i \leq 0 & i \in \mathcal{I} \leftarrow \lambda_i \end{cases}$$

3. En introduisant les variables d'écart x_{n+r} :

$$\begin{cases} -\sum_{\substack{j=1 \\ j \neq i}}^n q_{ij} x_j - 2q_{ii} x_i - c_i - \sum_{r=1}^m \lambda_{n+r} a_{ri} + \lambda_i = 0 & i \in \mathcal{I} \\ a_r^T x + x_{n+r} = b_r & r \in \mathcal{R} \\ \lambda_i x_i = 0 & i \in \mathcal{I} \cup \mathcal{R} \\ x_i \geq 0 & i \in \mathcal{I} \cup \mathcal{R} \\ \lambda_i \geq 0 & i \in \mathcal{I} \cup \mathcal{R} \end{cases}$$

Cas particulier des contraintes linéaires

$$(QP) \begin{cases} \min & \sum_i \sum_j q_{ij} x_i x_j + \sum_i c_i x_i \\ \text{s.t.} & a_r^T x \leq b_r \quad r \in \mathcal{R} \leftarrow \lambda_{n+r} \\ & -x_i \leq 0 \quad i \in \mathcal{I} \leftarrow \lambda_i \end{cases}$$

Résoudre (QP) \iff trouver une solution de (1), qui satisfait (2) .

$$(1) \begin{cases} -\sum_{\substack{j=1 \\ j \neq i}}^n q_{ij} x_j - 2q_{ii} x_i - c_i - \sum_{r=1}^m \lambda_{n+r} a_{ri} + \lambda_i = 0 & i \in \mathcal{I} \\ a_r^T x + x_{n+r} = b_r & r \in \mathcal{R} \end{cases}$$

$$(2) \begin{cases} \lambda_i x_i = 0 & i \in \mathcal{I} \cup \mathcal{R} \\ x_i \geq 0 & i \in \mathcal{I} \cup \mathcal{R} \\ \lambda_i \geq 0 & i \in \mathcal{I} \cup \mathcal{R} \end{cases}$$

\implies Appliquer le simplexe sur (1) jusqu'à satisfaire (2).

Cas particulier des contraintes linéaires

Algorithme : Application d'une variante du simplexe pour trouver la solution du système KKT :

1. Initialisation
2. Choix de la variable qui entre dans la base
3. Choix de la variable qui sort de la base
4. Critère d'arrêt
5. Convergence de l'algorithme

1. Initialisation

On considère la base triviale du système formé des variables :

- $\lambda_i, i \in \mathcal{I}$
- $x_{n+r}, r \in \mathcal{R}$

dont la solution associée est :

- $\lambda_i = c_i, i \in \mathcal{I}$
- $x_{n+r} = b_r, r \in \mathcal{R}$
- $x_i = 0, i \in \mathcal{I}$ et $\lambda_{n+r} = 0, r \in \mathcal{R}$

$$\begin{cases} -\sum_{\substack{j=1 \\ j \neq i}}^n q_{ij}x_j - 2q_{ii}x_i - c_i - \sum_{r=1}^m \lambda_{n+r}a_{ri} + \lambda_i = 0 & i \in \mathcal{I} \\ a_r^T x + x_{n+r} = b_r & r \in \mathcal{R} \end{cases}$$

Ici on a $2(n + m)$ variables et $n + m$ contraintes.

Tableau du simplexe

On appelle **tableau du simplexe** l'expression des variables de base en fonction des variables hors base.

On appelle **tableau standard** : le cas où pour tout $i \in \mathcal{I} \cup \mathcal{R}$, si x_i est dans la base, alors λ_i n'est pas dans la base

- Si le tableau est standard alors $\lambda_i x_i = 0$, $i \in \mathcal{I}$ est vérifié.
- Sinon ce n'est pas forcément le cas (sauf en cas de dégénérescence).

On va itérer en ayant toujours au plus une seule paire (x, λ) dans la base et hors de la base.

2. Choix de la variable qui entre dans la base

On considère deux cas :

- Si le tableau est **standard**, on fait entrer la variable x_e telle que la variable correspondant dans la base λ_e soit la plus petite (négative) possible.
(Les seules contraintes de (2) non satisfaites sont la positivité des solution en base, c'est elles qu'il faut rendre positives et non le gradient.)
- Sinon, on fait entrer la variable x_e telle que les deux variables x_e et λ_e sont hors base.
(On maintient toujours au plus une paire (x, λ) dans la base et hors de la base)

3. Choix de la variable qui sort de la base

On considère la variable λ_s qui vaut :

- Si le tableau est standard, λ_s est la variable qui correspond à la variable x_e qui est entrée dans la base (i.e. $\lambda_s = \lambda_e$).
- Sinon, λ_s est la variable du seul couple (x_i, λ_i) qui est dans la base.

On choisit la variable sortante parmi les x et λ_s (ensemble \mathcal{C}).

(On prend celle qui la première s'annule quand on augmente la valeur de la variables entrante)

On prend celle qui est associée à :

$$\min_{i \in \mathcal{C}} \left\{ \frac{v_i}{w_i} : w_i \neq 0 \text{ et } \frac{v_i}{w_i} > 0 \right\}$$

- v_i : valeur de la variable de base associée à la i^{eme} ligne du tableau
- w_i : l'élément de la i^{eme} ligne de la colonne e de la variable entrante x_e
- \mathcal{C} : l'ensemble des variables candidates pour la sortie.

4. Critère d'arrêt

- Le problème a toujours une solution : lorsque toutes les composantes du vecteur x sont nulles.
- Si le problème n'a pas de solution optimale finie, on le constate lorsque la valeur de la variable entrante x_e peut tendre vers l'infini. On peut montrer que ce n'est possible que sur un tableau standard.
- Si le problème admet une solution finie l'algorithme s'arrête quand :
 - ❶ La solution courante n'a que des composantes non négatives,
 - ❷ La solution courante satisfait les conditions KKTOn peut montrer que ce n'est possible que sur un tableau standard.

5. convergence de l'algorithme

- On peut montrer que le nombre de tableaux successifs (standards ou non) est inférieur ou égal à $n - m$.
- On peut montrer que la valeur de la fonction objectif dans un tableau standard diminue à chaque itération, et n'augmente pas pour un tableau non standard

⇒ On en déduit que l'algorithme converge en un nombre fini d'itération, si la solution est finie.

Exemple suivi

$$(QP) \begin{cases} \min & f(x) = -6x_1 + 2x_1^2 - 2x_1x_2 + 2x_2^2 \\ \text{s.t.} & x_1 + x_2 \leq 2 \\ & x_i \geq 0 \end{cases} \quad i = \{1, 2\}$$

Exemple suivi

$$(QP) \begin{cases} \min & f(x) = -6x_1 + 2x_1^2 - 2x_1x_2 + 2x_2^2 \\ \text{s.t.} & x_1 + x_2 \leq 2 \\ & x_i \geq 0 \end{cases} \quad i = \{1, 2\}$$

La matrice Hessienne est $\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \succeq 0$ définie positive (valeurs propres : 1 et 1)

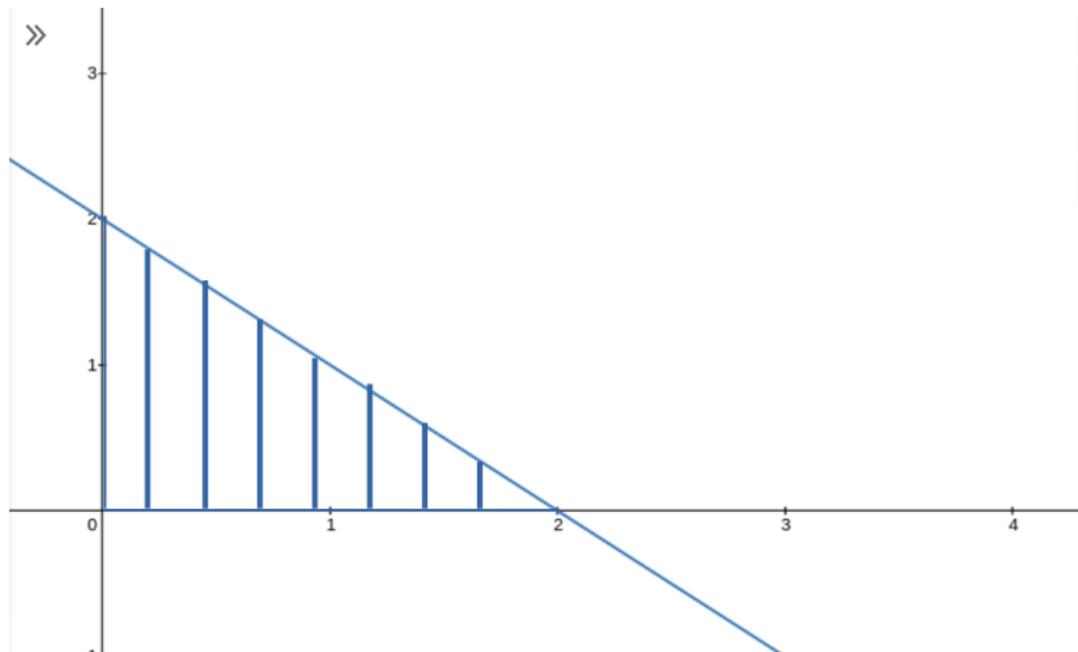
\implies la fonction objectif est convexe.

La contrainte est linéaire : $\implies (QP)$ est un problème convexe.

Exemple suivi

$$(QP) \begin{cases} \min & f(x) = -6x_1 + 2x_1^2 - 2x_1x_2 + 2x_2^2 \\ \text{s.t.} & x_1 + x_2 \leq 2 \\ & x_i \geq 0 \end{cases}$$

$$i = \{1, 2\}$$



Exemple suivi

$$(QP) \begin{cases} \min & f(x) = -6x_1 + 2x_1^2 - 2x_1x_2 + 2x_2^2 \\ \text{s.t.} & x_1 + x_2 \leq 2 \\ & -x_i \leq 0 \end{cases} \quad \begin{array}{l} \leftarrow \lambda_3 \\ i = \{1, 2\} \leftarrow \lambda_1, \lambda_2 \end{array}$$

Introduction du Lagrangien

$$L(x, \lambda) = -6x_1 + 2x_1^2 - 2x_1x_2 + 2x_2^2 + \lambda_3(x_1 + x_2 - 2) - \lambda_1x_1 - \lambda_2x_2$$

Exemple suivi

$$(QP) \begin{cases} \min & f(x) = -6x_1 + 2x_1^2 - 2x_1x_2 + 2x_2^2 \\ \text{s.t.} & x_1 + x_2 \leq 2 \\ & -x_i \leq 0 \end{cases} \quad \begin{array}{l} \leftarrow \lambda_3 \\ i = \{1, 2\} \leftarrow \lambda_1, \lambda_2 \end{array}$$

Introduction du Lagrangien

$$L(x, \lambda) = -6x_1 + 2x_1^2 - 2x_1x_2 + 2x_2^2 + \lambda_3(x_1 + x_2 - 2) - \lambda_1x_1 - \lambda_2x_2$$

Conditions KKT d'optimalité :

$$(1) \begin{cases} 6 - 4x_1 + 2x_2 - \lambda_3 + \lambda_1 = 0 \\ 2x_1 - 4x_2 - \lambda_3 + \lambda_2 = 0 \\ x_1 + x_2 + x_3 = 2 \end{cases}$$

$$(2) \begin{cases} \lambda_i x_i = 0 & i = 1, 2, 3 \\ x_i \geq 0 & i = 1, 2, 3 \\ \lambda_i \geq 0 & i = 1, 2, 3 \end{cases}$$

1. Initialisation (Exemple)

$$\left\{ \begin{array}{l} 6 - 4x_1 + 2x_2 - \lambda_3 + \lambda_1 = 0 \\ 2x_1 - 4x_2 - \lambda_3 + \lambda_2 = 0 \\ x_1 + x_2 + x_3 = 2 \\ \lambda_i x_i = 0 \quad i = 1, 2, 3 \\ x_i \geq 0 \quad i = 1, 2, 3 \\ \lambda_i \geq 0 \quad i = 1, 2, 3 \end{array} \right.$$

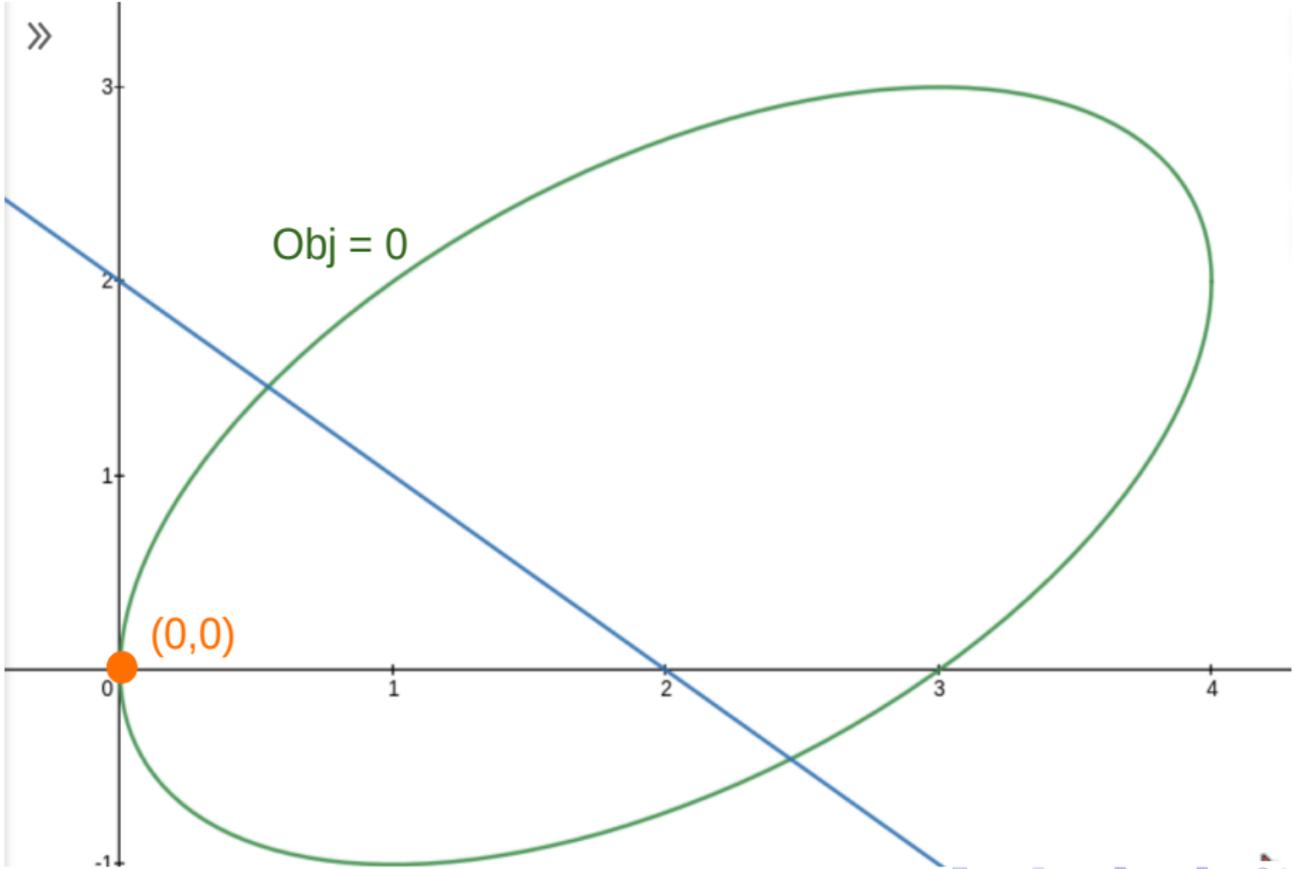
Tableau associé à la base initiale : $(\lambda_1, \lambda_2, x_3)$.

var. de base	variable hors base			variable en base			valeur
	x_1	x_2	λ_3	λ_1	λ_2	x_3	
λ_1	-4	2	-1	1	0	0	-6
λ_2	2	-4	-1	0	1	0	0
x_3	1	1	0	0	0	1	2

Solution : $\lambda_1 = -6$, $\lambda_2 = 0$, $x_3 = 2$, $x_1 = x_2 = \lambda_3 = 0$ de valeur $f(x) = -6x_1 + 2x_1^2 - 2x_1x_2 + 2x_2^2 = 0$.

Cette solution est **standard** : elle satisfait les contraintes d'égalité du système KKT, mais pas les contraintes de positivité : $\lambda_1 = -6$.

Illustration graphique



2.3. Variables entrante et sortante de la base (Exemple suite)

Variable entrant dans la base : tableau standard

- On choisit la variable qui entre dans la base parmi les x_j et x_{j+r} .
- On choisit celle pour laquelle la variable correspondante λ_j a la plus petite valeur (négative), ici $\lambda_1 = -6$, on fait donc entrer x_1 .

Variable sortant de la base

- La variable λ_s correspond ici à celle entrée dans la base, i.e. $\lambda_s = \lambda_1$
- L'ensemble des variables candidates est $\mathcal{C} = \{x_3, \lambda_1\}$.

•

$$\min_{i \in \mathcal{C}} \frac{v_i}{w_i} : w_i \neq 0 \text{ et } \frac{v_i}{w_i} > 0$$

v_i : valeur de la variable de base associée à la i^{eme} ligne du tableau

w_i : l'élément de la i^{eme} ligne de la colonne e de la variable entrante x_e

$$\min \left\{ \underbrace{\frac{-4}{-6}}_{\lambda_1}, \underbrace{\frac{1}{2}}_{x_3} \right\} = \underbrace{0.5}_{\lambda_1}$$

\implies on choisit λ_1

Mise à jour du tableau du simplexe

On effectue le pivotage comme dans l'algorithme du simplexe.

Construction du tableau associé à la base : x_1 , λ_2 et x_3 .

var. de base	x_1	x_2	λ_3	λ_1	λ_2	x_3	valeur
λ_1	-4	2	-1	1	0	0	-6
λ_2	2	-4	-1	0	1	0	0
x_3	1	1	0	0	0	1	2

Mise à jour du tableau du simplexe

On effectue le pivotage comme dans l'algorithme du simplexe.

Entrée de x_1 , sortie de λ_1 : on divise la première ligne par -4 .

var. de base	x_1	x_2	λ_3	λ_1	λ_2	x_3	valeur
λ_1	-4	2	-1	1	0	0	-6
λ_2	2	-4	-1	0	1	0	0
x_3	1	1	0	0	0	1	2

Mise à jour du tableau du simplexe

On effectue le pivotage comme dans l'algorithme du simplexe.

Entrée de x_1 , sortie de λ_1 : on divise la première ligne par -4 .

var. de base	x_1	x_2	λ_3	λ_1	λ_2	x_3	valeur
x_1	1	-0.5	0.25	-0.25	0	0	1.5
λ_2	2	-4	-1	0	1	0	0
x_3	1	1	0	0	0	1	2

Mise à jour du tableau du simplexe

On effectue le pivotage comme dans l'algorithme du simplexe.

Entrée de x_1 , sortie de λ_1 : $\forall i \in \mathcal{B} \setminus s : a'_{ij} = a_{ij} - a_{ie} \frac{a_{sj}}{a_{se}}$

var. de base	x_1	x_2	λ_3	λ_1	λ_2	x_3	valeur
x_1	1	-0.5	0.25	-0.25	0	0	1.5
λ_2	2	-4	-1	0	1	0	0
x_3	1	1	0	0	0	1	2

Exemples :

$$a'_{21} = a_{21} - a_{2e} \frac{a_{11}}{a_{1e}} = 2 - 2 \frac{1}{1} = 0$$

$$a'_{22} = a_{22} - a_{2e} \frac{a_{12}}{a_{1e}} = -4 - 2 \frac{-4}{1} = -3$$

Mise à jour du tableau du simplexe

On effectue le pivotage comme dans l'algorithme du simplexe.

Entrée de x_1 , sortie de λ_1 : $\forall i \in \mathcal{B} \setminus s : a'_{ij} = a_{ij} - a_{ie} \frac{a_{sj}}{a_{se}}$

var. de base	x_1	x_2	λ_3	λ_1	λ_2	x_3	valeur
x_1	1	-0.5	0.25	-0.25	0	0	1.5
λ_2	0	-3	-1.5	0.5	1	0	-3
x_3	1	1	0	0	0	1	2

Exemples :

$$a'_{21} = a_{21} - a_{21} \frac{a_{11}}{a_{11}} = 2 - 2 \frac{1}{1} = 0$$

$$a'_{22} = a_{22} - a_{21} \frac{a_{12}}{a_{11}} = -4 - 2 \frac{-4}{1} = -3$$

Mise à jour du tableau du simplexe

On effectue le pivotage comme dans l'algorithme du simplexe.

Entrée de x_1 , sortie de λ_1 : $\forall i \in \mathcal{B} \setminus s : a'_{ij} = a_{ij} - a_{ie} \frac{a_{sj}}{a_{se}}$

var. de base	x_1	x_2	λ_3	λ_1	λ_2	x_3	valeur
x_1	1	-0.5	0.25	-0.25	0	0	1.5
λ_2	0	-3	-1.5	0.5	1	0	-3
x_3	0	1.5	-0.25	0.25	0	1	0.5

Exemples :

$$a'_{21} = a_{21} - a_{2e} \frac{a_{11}}{a_{11}} = 2 - 2 \frac{1}{1} = 0$$

$$a'_{22} = a_{22} - a_{2e} \frac{a_{12}}{a_{11}} = -4 - 2 \frac{-4}{1} = -3$$

Mise à jour du tableau du simplexe

On effectue le pivotage comme dans l'algorithme du simplexe.

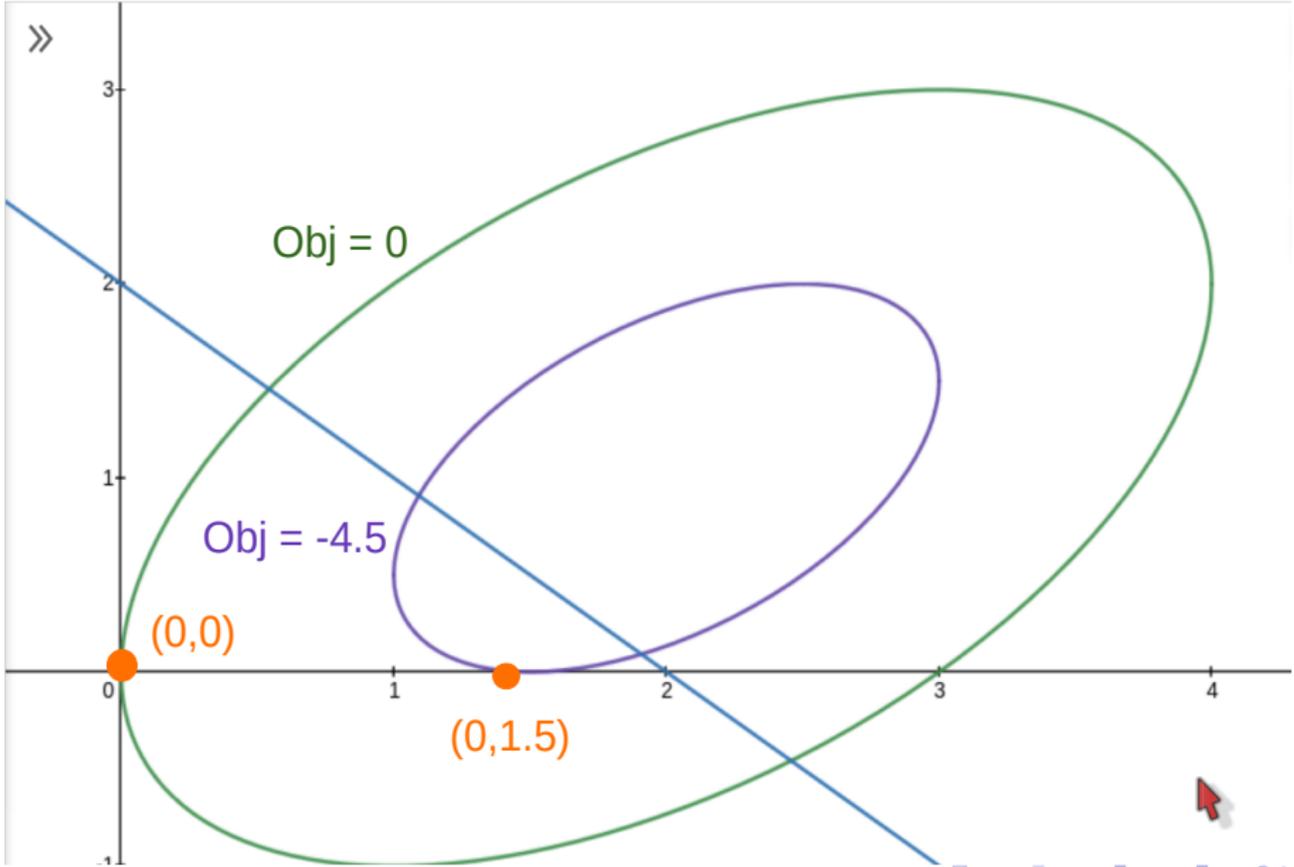
Entrée de x_1 , sortie de λ_1 : $\forall i \in \mathcal{B} \setminus s : a'_{ij} = a_{ij} - a_{ie} \frac{a_{sj}}{a_{se}}$

var. de base	x_1	x_2	λ_3	λ_1	λ_2	x_3	valeur
x_1	1	-0.5	0.25	-0.25	0	0	1.5
λ_2	0	-3	-1.5	0.5	1	0	-3
x_3	0	1.5	-0.25	0.25	0	1	0.5

Solution : $x_1 = 1.5$, $\lambda_2 = -3$, $x_3 = 0.5$, $\lambda_1 = x_2 = \lambda_3 = 0$ de valeur $f(x) = -6x_1 + 2x_1^2 - 2x_1x_2 + 2x_2^2 = -4.5$.

Cette solution est **standard** : elle satisfait les contraintes d'égalité du système KKT, mais pas les contraintes de positivités ($\lambda_2 = -3$).

Illustration graphique



2.3. Variables entrante et sortante de la base (Exemple suite)

Variable entrant dans la base : tableau standard

- On choisit la variable qui entre dans la base parmi les x_i et x_{i+r} .
- On choisit celle pour laquelle la variable correspondante λ_i a la plus petite valeur (négative), ici $\lambda_2 = -3$, on fait donc entrer x_2 .

Variable sortant de la base

- La variable λ_s correspond ici à celle entrée dans la base, i.e. $\lambda_s = \lambda_2$
- L'ensemble des variables candidates est $\mathcal{C} = \{x_1, x_3, \lambda_2\}$.

•

$$\min_{i \in \mathcal{C}} \frac{v_i}{w_i} : w_i \neq 0 \text{ et } \frac{v_i}{w_i} > 0$$

v_i : valeur de la variable de base associée à la i^{eme} ligne du tableau

w_i : l'élément de la i^{eme} ligne de la colonne e de la variable entrante x_e

$$\min \left\{ \underbrace{\frac{1.5}{-0.5}}_{x_1}, \underbrace{\frac{-3}{-3}}_{\lambda_2}, \underbrace{\frac{0.5}{1.5}}_{x_3} \right\} = \min \left\{ \underbrace{-3}_{x_1}, \underbrace{1}_{\lambda_2}, \underbrace{0.33}_{x_3} \right\} = \underbrace{0.33}_{x_3}$$

Mise à jour du tableau du simplexe

On effectue le pivotage comme dans l'algorithme du simplexe.

Construction du tableau associé à la base : x_1 , x_2 , et λ_2 .

var. de base	x_1	x_2	λ_3	λ_1	λ_2	x_3	valeur
x_1	1	-0.5	0.25	-0.25	0	0	1.5
λ_2	0	-3	-1.5	0.5	1	0	-3
x_3	0	1.5	-0.25	0.25	0	1	0.5

Mise à jour du tableau du simplexe

On effectue le pivotage comme dans l'algorithme du simplexe.

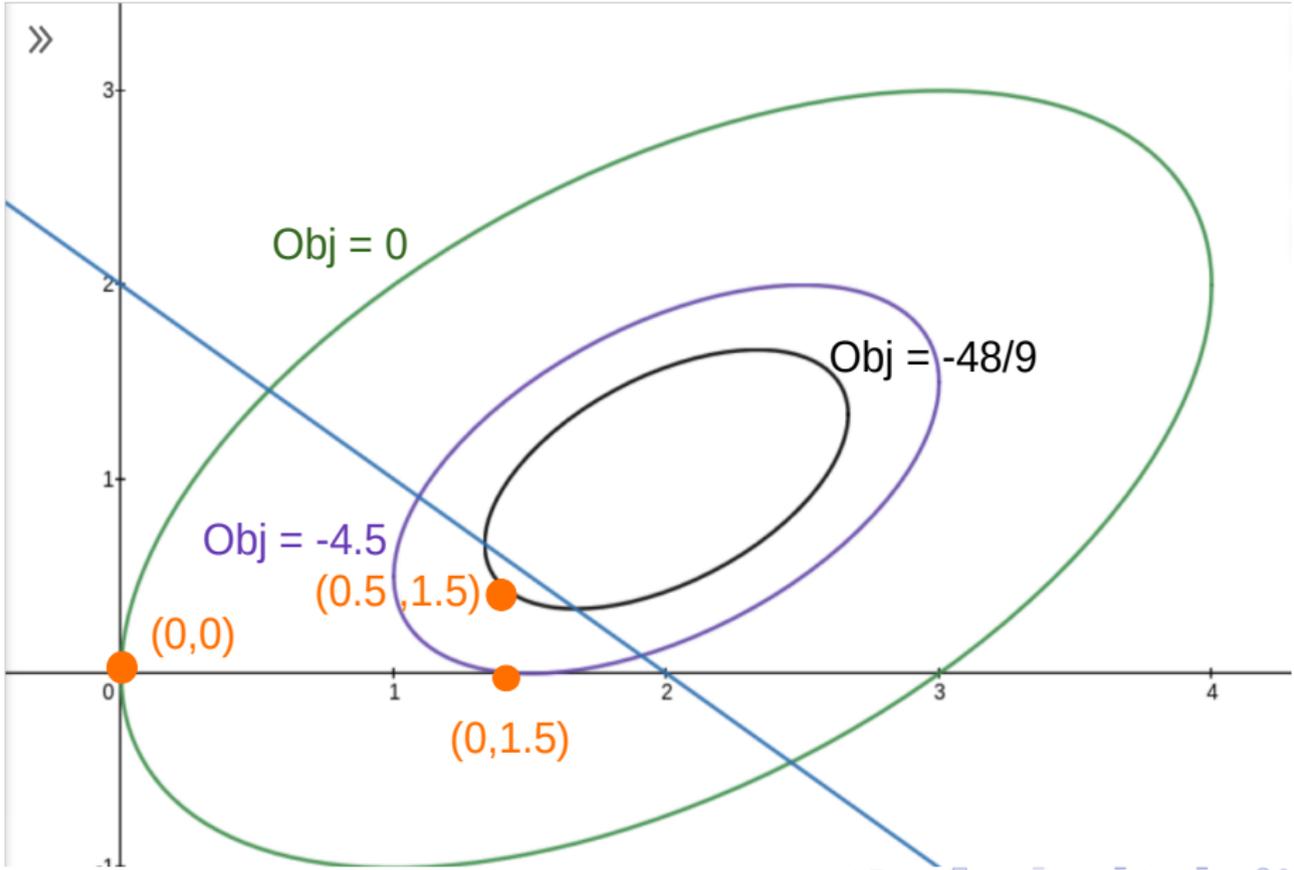
Entrée de x_2 , sortie de x_3 .

var. de base	x_1	x_2	λ_3	λ_1	λ_2	x_3	valeur
x_1	1	0	$\frac{1}{6}$	$-\frac{1}{6}$	0	$\frac{1}{3}$	$\frac{5}{3}$
λ_2	0	0	-2	1	1	2	-2
x_2	0	1	$-\frac{1}{6}$	$\frac{1}{6}$	0	$\frac{2}{3}$	$\frac{1}{3}$

Solution : $x_1 = \frac{5}{3}$, $\lambda_2 = -2$, $x_2 = \frac{1}{3}$, $\lambda_1 = x_3 = \lambda_3 = 0$, de valeur $f(x) = -6x_1 + 2x_1^2 - 2x_1x_2 + 2x_2^2 = -\frac{48}{9}$.

Cette solution est **non standard** : elle ne satisfait ni les contraintes d'égalité du système KKT ($\lambda_2 x_2 = -\frac{2}{3}$), ni les contraintes de positivités ($\lambda_2 = -2$).

Illustration graphique



2.3. Variables entrante et sortante de la base (Exemple suite)

Variable entrant dans la base : tableau non standard

- On choisit la variable λ_j dont la paire associée est hors base (ici la paire hors base est $(x_3, \lambda_3) \implies \lambda_3$).

Variable sortant de la base

- La variable λ du couple en base, ici $(x_2, \lambda_2) \implies \lambda_2$.

Mise à jour du tableau du simplexe

On effectue le pivotage comme dans l'algorithme du simplexe.

Construction du tableau associé à la base : x_1 , x_2 et λ_3 .

var. de base	x_1	x_2	λ_3	λ_1	λ_2	x_3	valeur
x_1	1	0	$\frac{1}{6}$	$-\frac{1}{6}$	0	$\frac{1}{3}$	$\frac{5}{3}$
λ_2	0	0	-2	1	1	2	-2
x_2	0	1	$-\frac{1}{6}$	$\frac{1}{6}$	0	$\frac{2}{3}$	$\frac{1}{3}$

Mise à jour du tableau du simplexe

On effectue le pivotage comme dans l'algorithme du simplexe.

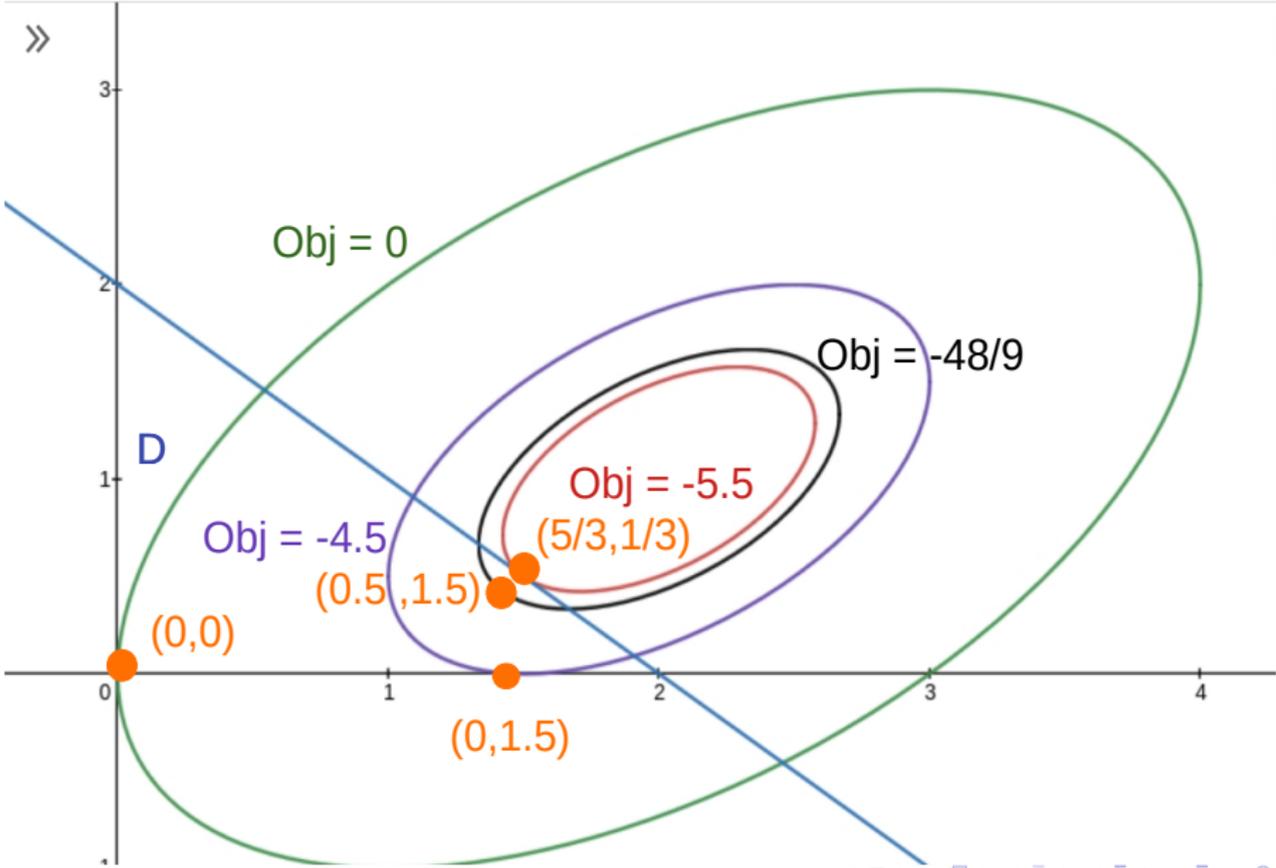
Entrée de λ_3 , sortie de λ_2 .

var. de base	x_1	x_2	λ_3	λ_1	λ_2	x_3	valeur
x_1	1	0	0	$-\frac{1}{12}$	$\frac{1}{12}$	$\frac{1}{2}$	$\frac{3}{2}$
λ_3	0	0	1	$-\frac{1}{2}$	$-\frac{1}{2}$	-1	1
x_2	0	1	0	$\frac{1}{12}$	$-\frac{1}{12}$	$\frac{1}{2}$	$\frac{1}{2}$

La solution est $x_1 = \frac{3}{2}$, $\lambda_3 = 1$, $x_2 = \frac{1}{2}$, $\lambda_1 = x_3 = \lambda_2 = 0$, de valeur $f(x) = -6x_1 + 2x_1^2 - 2x_1x_2 + 2x_2^2 = -5.5$.

Cette solution satisfait toutes les contraintes KKT, est optimale.

Illustration graphique



1 Introduction

- Les classes de problèmes non-linéaires
- Les formes quadratiques
- L'optimisation quadratique

2 L'optimisation quadratique convexe

- Les conditions d'optimalité
- Différence avec la résolution d'un PLNE
- Cas d'une fonction quadratique convexe sous des contraintes linéaires

3 L'optimisation quadratique non convexe

- Reformulations linéaires de produits de variables binaires
- Reformulation linéaire de produits mixtes
- Reformulation linéaire de produits de variables entières
- Relaxation linéaire de produits de variables continues

L'optimisation quadratique non convexe

Résolution exacte des programmes quadratiques

$$(P) \begin{cases} \min f_0(x) = \langle Q_0, xx^T \rangle + c_0^T x \\ \text{s.t. } f_r(x) = \langle Q_r, xx^T \rangle + c_r^T x \leq e_r \quad (r \in \mathcal{R}) \\ 0 \leq \ell_i \leq x_i \leq u_i \quad (i \in \mathcal{I}) \\ x_i \in \mathbb{N} \quad (i \in \mathcal{J}), \quad x_i \in \mathbb{R} \quad (i \in \mathcal{I} \setminus \mathcal{J}) \end{cases}$$

- $\mathcal{I} = \{1, \dots, N\}$ et $\mathcal{J} = \{1, \dots, n\}$: N variables dont n entières.
- $\mathcal{R} = \{1, \dots, m\}$: m contraintes quadratiques
Remarque : si $Q_r = 0_n \forall r$, elles sont linéaires.

Double difficulté pour la résolution exacte :

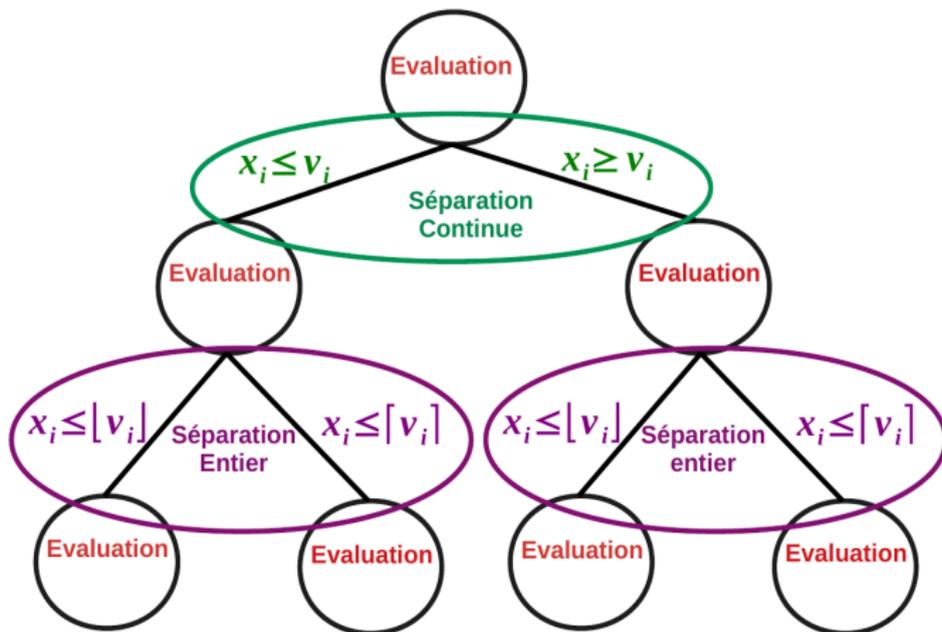
- l'intégrité d'une partie des variables,
- l'éventuelle **non-convexité** des fonctions.

Algorithme générique : spatial branch-and-bound

Résolution exacte : cas quadratique en variables mixtes

Algorithmes de **Branch-and-bound spatiaux**.

→ recherche arborescente : deux opérations **Séparation** et **Évaluation**



Résolution exacte : cas quadratique en variables mixtes

Algorithmes de **Branch-and-bound spatiaux**.

→ recherche arborescente : deux opérations **Séparation** et **Évaluation**

Résoudre une relaxation “facile” pour déterminer une **borne inférieure**

Relaxation	Qualité	Temps d'évaluation
<i>linéaire</i>	souvent faible	rapide
<i>semidéfinie</i>	très serrée	lente
<i>quadratique convexe</i>	peut être serrée	rapide

Résolution exacte : cas quadratique en variables mixtes

Algorithmes de **Branch-and-bound spatiaux**.

→ recherche arborescente : deux opérations **Séparation** et **Évaluation**

Résoudre une relaxation “facile” pour déterminer une **borne inférieure**

Relaxation	Qualité	Temps d'évaluation
<i>linéaire</i>	souvent faible	rapide
<i>semidéfinie</i>	très serrée	lente
<i>quadratique convexe</i>	peut être serrée	rapide

Relaxations linéaires : les plus étudiées et les plus rapides à calculer.

État de l'art : une centaine de variables dans le cas binaires, des problèmes plus petits pour les cas continus ou entiers généraux.

Limites : les bornes sont en général faibles \Rightarrow difficile de fermer le gap.

Résolution exacte : cas quadratique en variables mixtes

Algorithmes de **Branch-and-bound spatiaux**.

→ recherche arborescente : deux opérations **Séparation** et **Évaluation**

Résoudre une relaxation “facile” pour déterminer une **borne inférieure**

Relaxation	Qualité	Temps d'évaluation
<i>linéaire</i>	souvent faible	rapide
<i>semidéfinie</i>	très serrée	lente
<i>quadratique convexe</i>	peut être serrée	rapide

Relaxations semidéfinie : fournit les bornes les plus serrées, lent à évaluer :

État de l'art : quelques centaines de variables et de contraintes.

Limites : résoudre un SDP à chaque noeud est trop coûteux en temps.

Résolution exacte : cas quadratique en variables mixtes

Algorithmes de **Branch-and-bound spatiaux**.

→ recherche arborescente : deux opérations **Séparation** et **Évaluation**

Résoudre une relaxation “facile” pour déterminer une **borne inférieure**

Relaxation	Qualité	Temps d'évaluation
<i>linéaire</i>	souvent faible	rapide
<i>semidéfinie</i>	très serrée	lente
<i>quadratique convexe</i>	peut être serrée	rapide

Relaxations quadratiques convexes : quasiment aussi rapides à évaluer que les relaxations linéaires

Avantages elles peuvent être paramétrées pour renforcer la borne

But Déterminer une relaxation quadratique aussi serrée qu'une borne SDP.

⇒ Bénéficier de la force des bornes SDP au sein d'un B&B devient efficace

Reformulations linéaires de produits de variables binaires

Linéarisation de Fortet

On considère le problème quadratique binaire :

$$(P_{0,1}) \begin{cases} \min & f_0(x) = \langle Q_0, xx^T \rangle + c_0^T x \\ \text{s.t.} & f_r(x) = \langle Q_r, xx^T \rangle + c_r^T x - e_r \leq 0 \quad (r \in \mathcal{R}) \\ & x_i \in \{0, 1\} \quad (i \in \mathcal{J}) \end{cases}$$

La non convexité des fonctions vient des produits de variables $x_i x_j$

Linéarisation de Fortet

On considère le problème quadratique binaire :

$$(P_{0,1}) \begin{cases} \min f_0(x) = \langle Q_0, xx^T \rangle + c_0^T x \\ \text{s.t. } f_r(x) = \langle Q_r, xx^T \rangle + c_r^T x - e_r \leq 0 & (r \in \mathcal{R}) \\ x_i \in \{0, 1\} & (i \in \mathcal{J}) \end{cases}$$

La non convexité des fonctions vient des produits de variables $x_i x_j$

Idée :

- 1 Remplacer chaque produit $x_i x_j$ par une nouvelle variable y_{ij}
- 2 Ajouter la contrainte $y_{ij} = x_i x_j$, pour forcer l'équivalence avec le problème de départ.

Linéarisation de Fortet

On considère le problème quadratique binaire :

$$(P_{0,1}) \begin{cases} \min f_0(x) = \langle Q_0, xx^T \rangle + c_0^T x \\ \text{s.t. } f_r(x) = \langle Q_r, xx^T \rangle + c_r^T x - e_r \leq 0 & (r \in \mathcal{R}) \\ x_i \in \{0, 1\} & (i \in \mathcal{J}) \end{cases}$$

La non convexité des fonctions vient des produits de variables $x_i x_j$

$$(P'_{0,1}) \begin{cases} \min f_0(x) = \langle Q_0, Y \rangle + c_0^T x \\ \text{s.t. } f_r(x) = \langle Q_r, Y \rangle + c_r^T x - e_r \leq 0 & (r \in \mathcal{R}) \\ Y = xx^T \\ x_i \in \{0, 1\} & (i \in \mathcal{J}) \end{cases}$$

Linéarisation de Fortet

On considère le problème quadratique binaire :

$$(P_{0,1}) \begin{cases} \min f_0(x) = \langle Q_0, xx^T \rangle + c_0^T x \\ \text{s.t. } f_r(x) = \langle Q_r, xx^T \rangle + c_r^T x - e_r \leq 0 & (r \in \mathcal{R}) \\ x_i \in \{0, 1\} & (i \in \mathcal{J}) \end{cases}$$

La non convexité des fonctions vient des produits de variables $x_i x_j$

$$(P'_{0,1}) \begin{cases} \min f_0(x) = \langle Q_0, Y \rangle + c_0^T x \\ \text{s.t. } f_r(x) = \langle Q_r, Y \rangle + c_r^T x - e_r \leq 0 & (r \in \mathcal{R}) \\ Y = xx^T \\ x_i \in \{0, 1\} & (i \in \mathcal{J}) \end{cases}$$

- $(P_{0,1})$ et $(P'_{0,1})$ sont équivalents
- La fonction objectif et les contraintes sont linéaires \rightarrow convexes,
- **Mais la nouvelle contrainte $Y = xx^T$ est non convexe**

Réécriture convexe de $y_{ij} = x_i x_j$

Théorème (Fortet 59) :

$$y_{ij} = x_i x_j \text{ et } (x_i, x_j) \in \{0, 1\}^2 \Leftrightarrow \begin{cases} y_{ij} \leq x_i \\ y_{ij} \leq x_j \\ y_{ij} \geq x_i + x_j - 1 \\ y_{ij} \geq 0 \end{cases}$$

Réécriture convexe de $y_{ij} = x_i x_j$

Théorème (Fortet 59) :

$$y_{ij} = x_i x_j \text{ et } (x_i, x_j) \in \{0, 1\}^2 \Leftrightarrow \begin{cases} y_{ij} \leq x_i \\ y_{ij} \leq x_j \\ y_{ij} \geq x_i + x_j - 1 \\ y_{ij} \geq 0 \end{cases}$$

Preuve :

- si $x_i = 0$ et/ou $x_j = 0$, on a $y_{ij} \leq x_i = 0$ et $y_{ij} \geq 0 \Leftrightarrow y_{ij} = 0$.

Réécriture convexe de $y_{ij} = x_i x_j$

Théorème (Fortet 59) :

$$y_{ij} = x_i x_j \text{ et } (x_i, x_j) \in \{0, 1\}^2 \Leftrightarrow \begin{cases} y_{ij} \leq x_i \\ y_{ij} \leq x_j \\ y_{ij} \geq x_i + x_j - 1 \\ y_{ij} \geq 0 \end{cases}$$

Preuve :

- si $x_i = 0$ et/ou $x_j = 0$, on a $y_{ij} \leq x_i = 0$ et $y_{ij} \geq 0 \Leftrightarrow y_{ij} = 0$.
- si $x_i = x_j = 1$, on a $y_{ij} \leq x_i = 1$ et $y_{ij} \geq x_i + x_j - 1 = 1 \Leftrightarrow y_{ij} = 1$.

Réécriture convexe de $y_{ij} = x_i x_j$

Théorème (Fortet 59) :

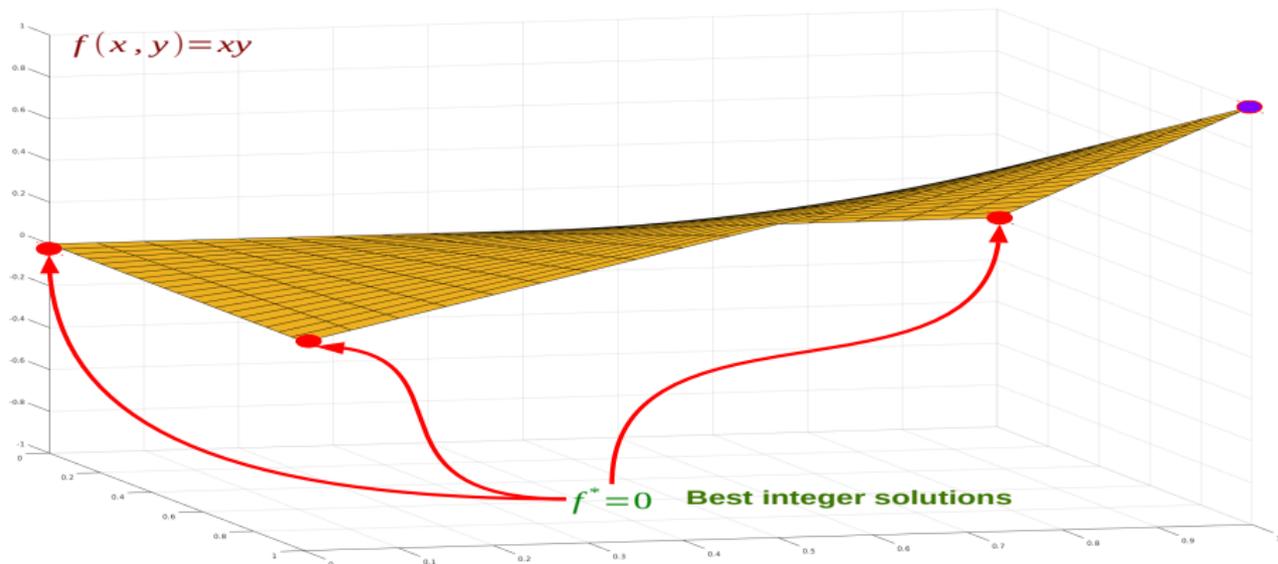
$$y_{ij} = x_i x_j \text{ et } (x_i, x_j) \in \{0, 1\}^2 \Leftrightarrow \begin{cases} y_{ij} \leq x_i \\ y_{ij} \leq x_j \\ y_{ij} \geq x_i + x_j - 1 \\ y_{ij} \geq 0 \end{cases}$$

$$(P_{0,1}) \Leftrightarrow (F_{0,1}) \begin{cases} \min f_0(x) = \langle Q_0, Y \rangle + c_0^T x \\ \text{s.t. } f_r(x) = \langle Q_r, Y \rangle + c_r^T x - e_r \leq 0 \quad (r \in \mathcal{R}) \\ y_{ij} \leq x_i \quad ((i,j) \in \mathcal{J}^2) \\ y_{ij} \leq x_j \quad ((i,j) \in \mathcal{J}^2) \\ y_{ij} \geq x_i + x_j - 1 \quad ((i,j) \in \mathcal{J}^2) \\ y_{ij} \geq 0 \quad ((i,j) \in \mathcal{J}^2) \\ x_i \in \{0, 1\} \quad (i \in \mathcal{J}) \end{cases}$$

On peut résoudre $(F_{0,1})$ avec un b&b basé sur la relaxation continue

Linéarisation de Fortet : exemple cas binaire

$$\begin{cases} \min_{(x,y) \in \{0,1\}} xy \\ \text{valeur opt} : 0 \\ \text{sol opt} : (0,0), (0,1), (1,0) \end{cases}$$



Linéarisation de Fortet : exemple cas binaire

$$\begin{cases} \min_{(x,y) \in \{0,1\}} xy \\ \text{valeur opt : } 0 \\ \text{sol opt : } (0,0), (0,1), (1,0) \end{cases}$$

On introduit

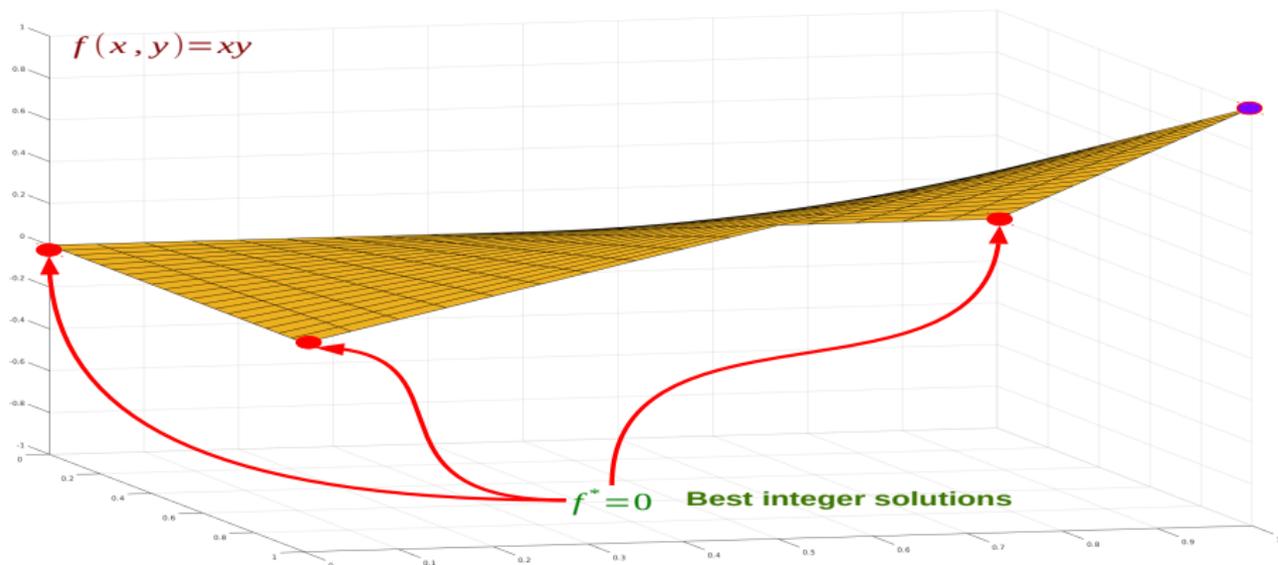
- z qui modélise le produit xy
- un ensemble d'inégalités de linéarisation.

$$\begin{cases} z = xy \\ (x, y) \in \{0, 1\} \end{cases} \iff \begin{cases} z \geq x + y - 1 \\ z \geq 0 \\ z \leq y \\ z \leq x \\ (x, y) \in \{0, 1\} \end{cases}$$

Linéarisation de Fortet : exemple cas binaire

$$\begin{cases} \min_{(x,y) \in \{0,1\}} xy \end{cases} \Leftrightarrow \begin{cases} \min_{(x,y) \in \{0,1\}} z \\ \text{s.t. } z \geq x + y - 1, z \geq 0, z \leq y, z \leq x \end{cases}$$

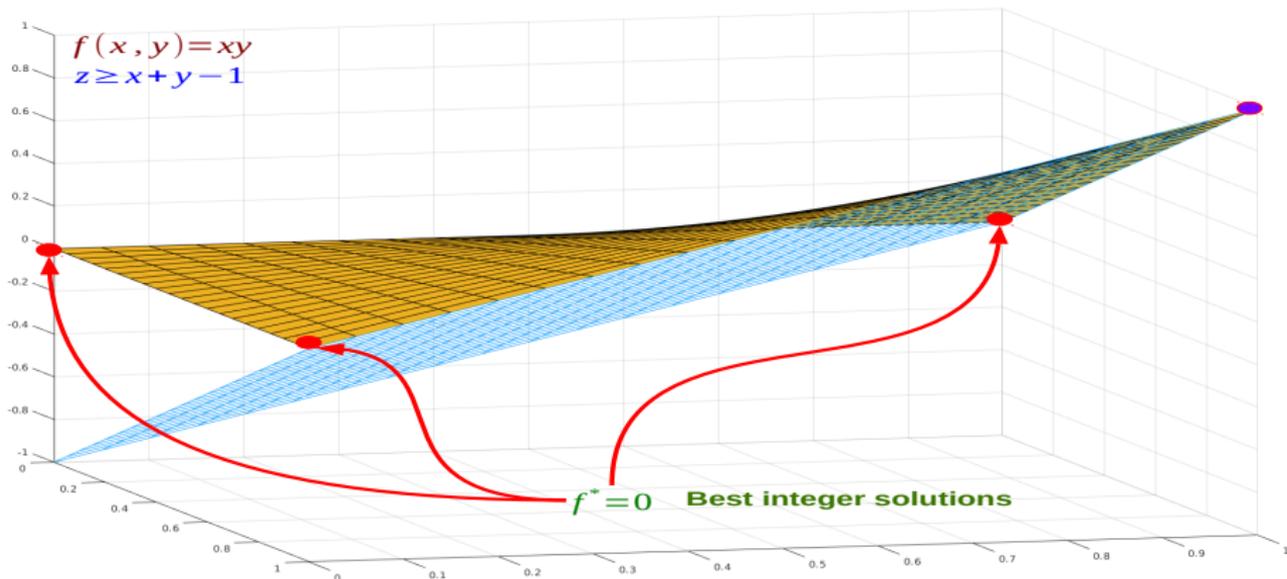
⇒ reformulation linéaire équivalente sur chaque point binaire.



Linéarisation de Fortet : exemple cas binaire

$$\left\{ \begin{array}{l} \min_{(x,y) \in \{0,1\}} xy \end{array} \right\} \Leftrightarrow \left\{ \begin{array}{l} \min_{(x,y) \in \{0,1\}} z \\ \text{s.t. } z \geq x + y - 1, z \geq 0, z \leq y, z \leq x \end{array} \right.$$

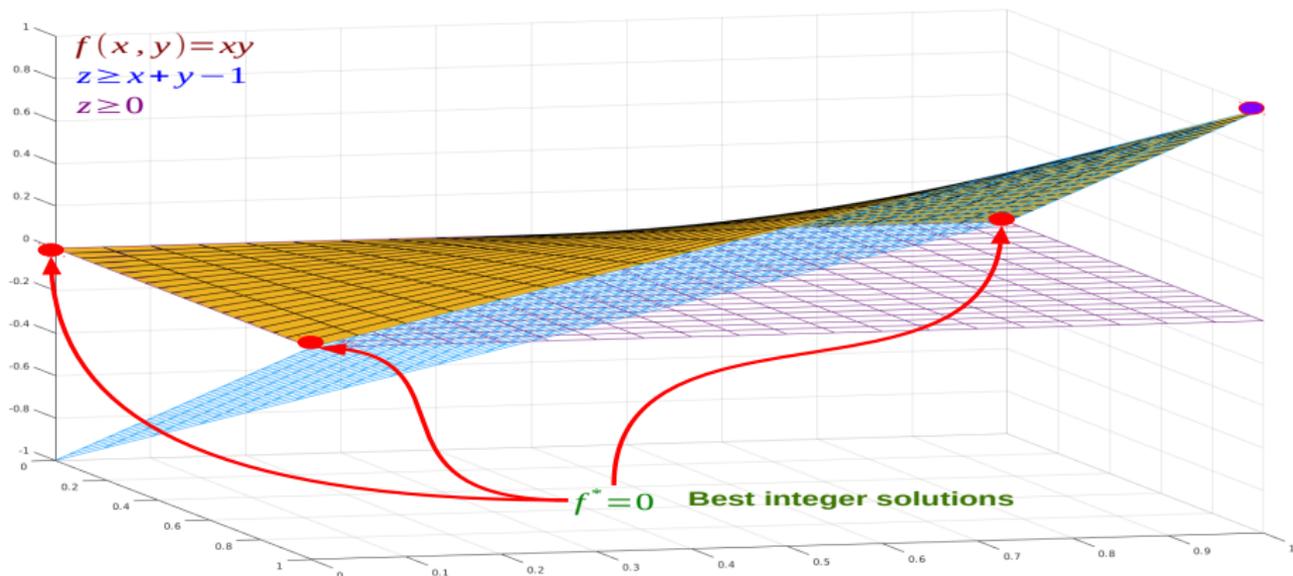
⇒ reformulation linéaire équivalente sur chaque point binaire.



Linéarisation de Fortet : exemple cas binaire

$$\left\{ \begin{array}{l} \min_{(x,y) \in \{0,1\}} xy \end{array} \right\} \Leftrightarrow \left\{ \begin{array}{l} \min_{(x,y) \in \{0,1\}} z \\ \text{s.t. } z \geq x + y - 1, z \geq 0, z \leq y, z \leq x \end{array} \right.$$

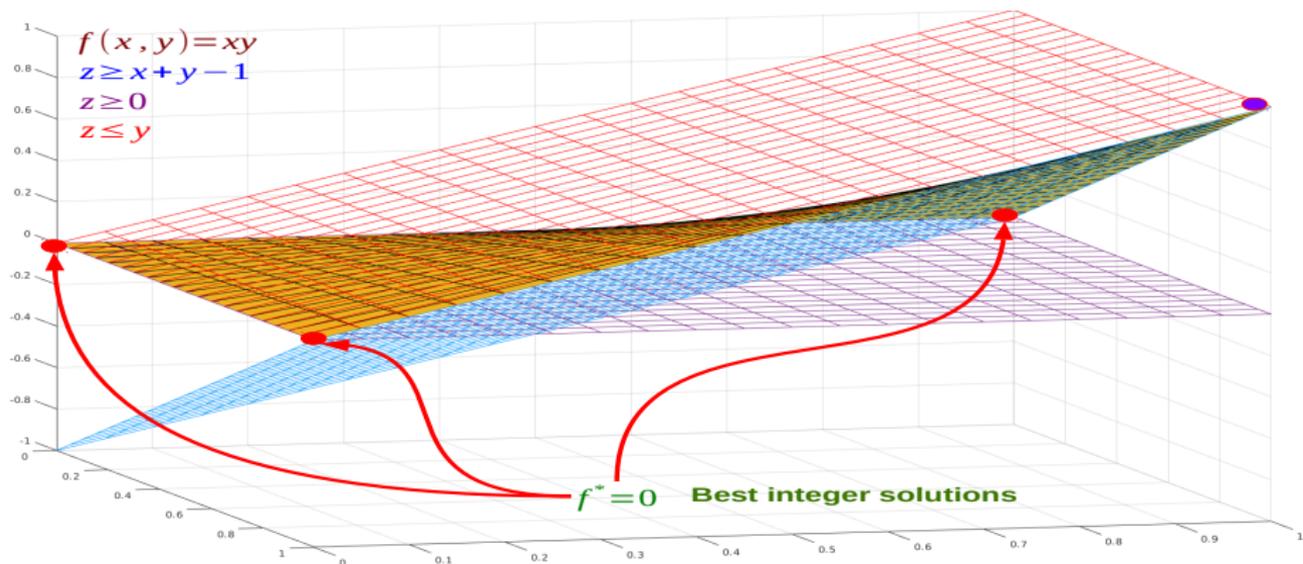
⇒ reformulation linéaire équivalente sur chaque point binaire.



Linéarisation de Fortet : exemple cas binaire

$$\left\{ \begin{array}{l} \min_{(x,y) \in \{0,1\}} xy \end{array} \right\} \Leftrightarrow \left\{ \begin{array}{l} \min_{(x,y) \in \{0,1\}} z \\ \text{s.t. } z \geq x + y - 1, z \geq 0, z \leq y, z \leq x \end{array} \right.$$

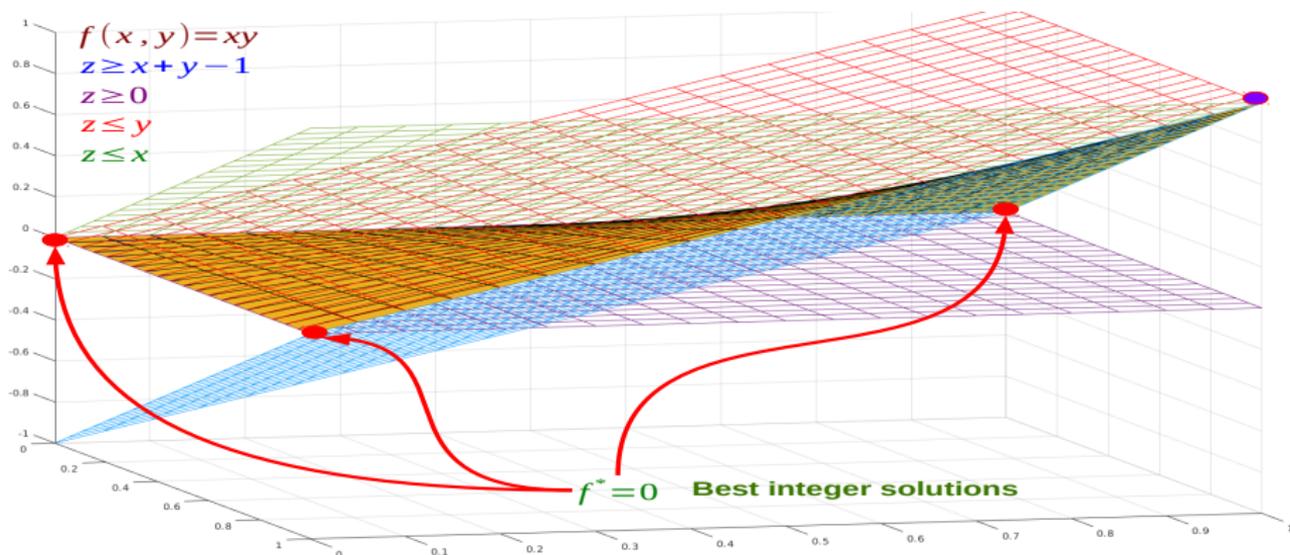
⇒ reformulation linéaire équivalente sur chaque point binaire.



Linéarisation de Fortet : exemple cas binaire

$$\left\{ \begin{array}{l} \min_{(x,y) \in \{0,1\}} xy \\ \end{array} \right. \Leftrightarrow \left\{ \begin{array}{l} \min_{(x,y) \in \{0,1\}} z \\ \text{s.t. } z \geq x + y - 1, z \geq 0, z \leq y, z \leq x \end{array} \right.$$

⇒ reformulation linéaire équivalente sur chaque point binaire.



Linéarisation de Fortet

- Ajout de n^2 variables additionnelles y , et $4n^2$ contraintes.
- Fournit une reformulation exacte sur chaque point entier
⇒ Un solveur standard peut le résoudre directement avec un branch-and-bound basé sur la relaxation continue.
- Le nombre de contraintes et de variables additionnelles est le même avec ou sans contraintes quadratiques.
⇒ Il suffit de linéariser les contraintes avec les variables y .

Linéarisation de Glover

On considère toujours le problème quadratique binaire :

$$(P_{0,1}) \begin{cases} \min & f_0(x) = \langle Q_0, xx^T \rangle + c_0^T x \\ \text{s.t.} & f_r(x) = \langle Q_r, xx^T \rangle + c_r^T x - e_r \leq 0 \quad (r \in \mathcal{R}) \\ & x_i \in \{0, 1\} \quad (i \in \mathcal{J}) \end{cases}$$

La non convexité vient des produits de variables $x_i x_j$

Linéarisation de Glover

On considère toujours le problème quadratique binaire :

$$(P_{0,1}) \begin{cases} \min f_0(x) = \langle Q_0, xx^T \rangle + c_0^T x \\ \text{s.t. } f_r(x) = \langle Q_r, xx^T \rangle + c_r^T x - e_r \leq 0 & (r \in \mathcal{R}) \\ x_i \in \{0, 1\} & (i \in \mathcal{J}) \end{cases}$$

La non convexité vient des produits de variables $x_i x_j$

Idée :

① Linéariser un sous-ensemble des expressions non-linéaires.

② Remarquons que
$$\sum_{i=1}^n \sum_{j=1}^n q_{ij} x_i x_j = \sum_{i=1}^n x_i \sum_{j=1}^n q_{ij} x_j$$

③ On pose
$$z_i = x_i \sum_{j=1}^n q_{ij} x_j$$

Linéarisation de Glover

On considère toujours le problème quadratique binaire :

$$(P_{0,1}) \begin{cases} \min & f_0(x) = \langle Q_0, xx^T \rangle + c_0^T x \\ \text{s.t.} & f_r(x) = \langle Q_r, xx^T \rangle + c_r^T x - e_r \leq 0 \quad (r \in \mathcal{R}) \\ & x_i \in \{0, 1\} \quad (i \in \mathcal{J}) \end{cases}$$

$$(P'_{0,1}) \begin{cases} \min & \sum_{i=1}^n z_i + c_0^T x \\ \text{s.t.} & \sum_{i=1}^n z_{rn+i} + c_r^T x - e_r \leq 0 \quad (r \in \mathcal{R}) \\ & z_{rn+i} = x_i \sum_{j=1}^n q_{rij} x_j \quad (r \in \mathcal{R} \cup \{0\}, i \in \mathcal{J}) \\ & x_i \in \{0, 1\} \quad (i \in \mathcal{J}) \end{cases}$$

Linéarisation de Glover

On considère toujours le problème quadratique binaire :

$$(P_{0,1}) \begin{cases} \min f_0(x) = \langle Q_0, xx^T \rangle + c_0^T x \\ \text{s.t. } f_r(x) = \langle Q_r, xx^T \rangle + c_r^T x - e_r \leq 0 & (r \in \mathcal{R}) \\ x_i \in \{0, 1\} & (i \in \mathcal{J}) \end{cases}$$

$$(P'_{0,1}) \begin{cases} \min \sum_{i=1}^n z_i + c_0^T x \\ \text{s.t. } \sum_{i=1}^n z_{rn+i} + c_r^T x - e_r \leq 0 & (r \in \mathcal{R}) \\ z_{rn+i} = x_i \sum_{j=1}^n q_{rij} x_j & (r \in \mathcal{R} \cup \{0\}, i \in \mathcal{J}) \\ x_i \in \{0, 1\} & (i \in \mathcal{J}) \end{cases}$$

- $(P_{0,1})$ et $(P'_{0,1})$ sont équivalents
- La fonction objectif et les contraintes sont linéaires \rightarrow convexes,
- Mais la nouvelle contrainte $z_{rn+i} = x_i \sum_{j=1}^n q_{rij} x_j$ est non convexe

Quelles bornes pour les fonctions considérées ?

Soient L_{rn+i} et U_{rn+i} tels que pour tout r et i fixés :

$$L_{rn+i} \leq \sum_{j=1}^n q_{rij} x_j \leq U_{rn+i}$$

On peut par exemple prendre :

$$L_{rn+i} = \min \left\{ \sum_{j=1}^n q_{rij} x_j : x \in \{0, 1\}^n \right\}$$

$$U_{rn+i} = \max \left\{ \sum_{j=1}^n q_{rij} x_j : x \in \{0, 1\}^n \right\}$$

Réécriture convexe de $z_i = x_i \sum_{j=1}^n q_{ij} x_j$

Théorème (Glover 75) :

$$\left\{ \begin{array}{l} z_i = x_i \sum_{j=1}^n q_{ij} x_j \\ x_i \in \{0, 1\} \\ L_i \leq \sum_{j=1}^n q_{ij} x_j \leq U_i \end{array} \right. \Leftrightarrow \left\{ \begin{array}{l} z_i \leq U_i x_i \\ z_i \leq \sum_{j=1}^n q_{ij} x_j - L_i (1 - x_i) \\ z_i \geq L_i x_i \\ z_i \geq \sum_{j=1}^n q_{ij} x_j - U_i (1 - x_i) \end{array} \right.$$

Réécriture convexe de $z_i = x_i \sum_{j=1}^n q_{ij} x_j$

Théorème (Glover 75) :

$$\left\{ \begin{array}{l} z_i = x_i \sum_{j=1}^n q_{ij} x_j \\ x_i \in \{0, 1\} \\ L_i \leq \sum_{j=1}^n q_{ij} x_j \leq U_i \end{array} \right. \Leftrightarrow \left\{ \begin{array}{l} z_i \leq U_i x_i \\ z_i \leq \sum_{j=1}^n q_{ij} x_j - L_i (1 - x_i) \\ z_i \geq L_i x_i \\ z_i \geq \sum_{j=1}^n q_{ij} x_j - U_i (1 - x_i) \end{array} \right.$$

Preuve :

- si $x_i = 0$, on a $z_i \leq U_i x_i = 0$ et $z_i \geq L_i x_i = 0 \iff z_i = 0$.

Réécriture convexe de $z_i = x_i \sum_{j=1}^n q_{ij} x_j$

Théorème (Glover 75) :

$$\left\{ \begin{array}{l} z_i = x_i \sum_{j=1}^n q_{ij} x_j \\ x_i \in \{0, 1\} \\ L_i \leq \sum_{j=1}^n q_{ij} x_j \leq U_i \end{array} \right. \Leftrightarrow \left\{ \begin{array}{l} z_i \leq U_i x_i \\ z_i \leq \sum_{j=1}^n q_{ij} x_j - L_i (1 - x_i) \\ z_i \geq L_i x_i \\ z_i \geq \sum_{j=1}^n q_{ij} x_j - U_i (1 - x_i) \end{array} \right.$$

Preuve :

• si $x_i = 0$, on a $z_i \leq U_i x_i = 0$ et $z_i \geq L_i x_i = 0 \iff z_i = 0$.

• si $x_i = 1$, on a $\sum_{j=1}^n q_{ij} x_j \leq z_i \leq \sum_{j=1}^n q_{ij} x_j \iff z_i = \sum_{j=1}^n q_{ij} x_j$.

Linéarisation de Glover

On obtient alors le problème linéaire $(G_{0,1})$ suivant :

$$(P'_{0,1}) \left\{ \begin{array}{l} \min \sum_{i=1}^n z_i + c_0^T x \\ \text{s.t.} \sum_{i=1}^n z_{rn+i} + c_r^T x - e_r \leq 0 \quad (r \in \mathcal{R}) \\ z_{rn+i} \leq U_{rn+i} x_i \quad (r \in \mathcal{R} \cup \{0\}, i \in \mathcal{J}) \\ z_{rn+i} \leq \sum_{j=1}^n q_{rij} x_j - L_{rn+i} (1 - x_i) \quad (r \in \mathcal{R} \cup \{0\}, i \in \mathcal{J}) \\ z_{rn+i} \geq L_{rn+i} x_i \quad (r \in \mathcal{R} \cup \{0\}, i \in \mathcal{J}) \\ z_{rn+i} \geq \sum_{j=1}^n q_{rij} x_j - U_{rn+i} (1 - x_i) \quad (r \in \mathcal{R} \cup \{0\}, i \in \mathcal{J}) \\ x_i \in \{0, 1\} \quad (i \in \mathcal{J}) \end{array} \right.$$

Linéarisation de Glover

On obtient alors le problème linéaire $(G_{0,1})$ suivant :

$$(P'_{0,1}) \left\{ \begin{array}{l} \min \sum_{i=1}^n z_i + c_0^T x \\ \text{s.t.} \sum_{i=1}^n z_{rn+i} + c_r^T x - e_r \leq 0 \quad (r \in \mathcal{R}) \\ z_{rn+i} \leq U_{rn+i} x_i \quad (r \in \mathcal{R} \cup \{0\}, i \in \mathcal{J}) \\ z_{rn+i} \leq \sum_{j=1}^n q_{rij} x_j - L_{rn+i} (1 - x_i) \quad (r \in \mathcal{R} \cup \{0\}, i \in \mathcal{J}) \\ z_{rn+i} \geq L_{rn+i} x_i \quad (r \in \mathcal{R} \cup \{0\}, i \in \mathcal{J}) \\ z_{rn+i} \geq \sum_{j=1}^n q_{rij} x_j - U_{rn+i} (1 - x_i) \quad (r \in \mathcal{R} \cup \{0\}, i \in \mathcal{J}) \\ x_i \in \{0, 1\} \quad (i \in \mathcal{J}) \end{array} \right.$$

On peut résoudre $(G_{0,1})$ avec un b&b basé sur la relaxation continue 

Linéarisation de Glover : exemple cas binaire

$$\left\{ \begin{array}{l} \min \\ x \in \{0,1\}^3 \end{array} x_1 x_2 + x_2 x_1 + 2x_1 x_3 + 2x_3 x_1 - 3x_2 x_3 - 3x_3 x_2 \right.$$

Linéarisation de Glover : exemple cas binaire

$$\left\{ \begin{array}{l} \min \\ x \in \{0,1\}^3 \end{array} \right. x_1 x_2 + x_2 x_1 + 2x_1 x_3 + 2x_3 x_1 - 3x_2 x_3 - 3x_3 x_2$$

$$\begin{aligned} & x_1 x_2 + x_2 x_1 + 2x_1 x_3 + 2x_3 x_1 - 3x_2 x_3 - 3x_3 x_2 \\ = & x_1(x_2 + 2x_3) + x_2(x_1 - 3x_3) + x_3(2x_1 - 3x_2) \end{aligned}$$

Linéarisation de Glover : exemple cas binaire

$$\left\{ \begin{array}{l} \min \\ x \in \{0,1\}^3 \end{array} \right. x_1 x_2 + x_2 x_1 + 2x_1 x_3 + 2x_3 x_1 - 3x_2 x_3 - 3x_3 x_2$$

$$\begin{aligned} & x_1 x_2 + x_2 x_1 + 2x_1 x_3 + 2x_3 x_1 - 3x_2 x_3 - 3x_3 x_2 \\ = & \underbrace{x_1(x_2 + 2x_3)}_{z_1} + \underbrace{x_2(x_1 - 3x_3)}_{z_2} + \underbrace{x_3(2x_1 - 3x_2)}_{z_3} \end{aligned}$$

Linéarisation de Glover : exemple cas binaire

$$\left\{ \begin{array}{l} \min \\ x \in \{0,1\}^3 \end{array} \right. x_1 x_2 + x_2 x_1 + 2x_1 x_3 + 2x_3 x_1 - 3x_2 x_3 - 3x_3 x_2$$

$$\begin{aligned} & x_1 x_2 + x_2 x_1 + 2x_1 x_3 + 2x_3 x_1 - 3x_2 x_3 - 3x_3 x_2 \\ = & \underbrace{x_1(x_2 + 2x_3)}_{z_1} + \underbrace{x_2(x_1 - 3x_3)}_{z_2} + \underbrace{x_3(2x_1 - 3x_2)}_{z_3} = z_1 + z_2 + z_3 \end{aligned}$$

Linéarisation de Glover : exemple cas binaire

$$\left\{ \begin{array}{l} \min \\ x \in \{0,1\}^3 \end{array} x_1 x_2 + x_2 x_1 + 2x_1 x_3 + 2x_3 x_1 - 3x_2 x_3 - 3x_3 x_2 \right.$$

$$\begin{aligned} & x_1 x_2 + x_2 x_1 + 2x_1 x_3 + 2x_3 x_1 - 3x_2 x_3 - 3x_3 x_2 \\ &= \underbrace{x_1(x_2 + 2x_3)}_{z_1} + \underbrace{x_2(x_1 - 3x_3)}_{z_2} + \underbrace{x_3(2x_1 - 3x_2)}_{z_3} = z_1 + z_2 + z_3 \end{aligned}$$

on a : $L_1 = 0 \leq x_2 + 2x_3 \leq 3 = U_1$, $L_2 = -3 \leq x_1 - 3x_3 \leq 1 = U_2$,
 $L_3 = -3 \leq 2x_1 - 3x_2 \leq 2 = U_3$

Linéarisation de Glover : exemple cas binaire

$$\left\{ \begin{array}{l} \min_{x \in \{0,1\}^3} x_1 x_2 + x_2 x_1 + 2x_1 x_3 + 2x_3 x_1 - 3x_2 x_3 - 3x_3 x_2 \end{array} \right.$$

$$\begin{aligned} & x_1 x_2 + x_2 x_1 + 2x_1 x_3 + 2x_3 x_1 - 3x_2 x_3 - 3x_3 x_2 \\ &= \underbrace{x_1(x_2 + 2x_3)}_{z_1} + \underbrace{x_2(x_1 - 3x_3)}_{z_2} + \underbrace{x_3(2x_1 - 3x_2)}_{z_3} = z_1 + z_2 + z_3 \end{aligned}$$

on a : $L_1 = 0 \leq x_2 + 2x_3 \leq 3 = U_1$, $L_2 = -3 \leq x_1 - 3x_3 \leq 1 = U_2$,
 $L_3 = -3 \leq 2x_1 - 3x_2 \leq 2 = U_3$

$$\left\{ \begin{array}{l} \min_{x \in \{0,1\}^3, z \in \mathbb{R}^3} z_1 + z_2 + z_3 \\ \text{s.t.} \\ 0 \leq z_1 \leq 3x_1, \quad x_2 + 2x_3 - 3(1 - x_1) \leq z_1 \leq x_2 + 2x_3 \\ -3x_2 \leq z_2 \leq x_2, \quad x_1 - 3x_3 - (1 - x_2) \leq z_2 \leq x_1 - 3x_3 - 3(1 - x_2) \\ -3x_3 \leq z_3 \leq 2x_3, \quad 2x_1 - 3x_2 - 2(1 - x_3) \leq z_3 \leq 2x_1 - 3x_2 - 3(1 - x_3) \end{array} \right.$$

Linéarisation de Glover

- Ajout de $(r + 1)n$ variables additionnelles z , et $4(r + 1)n$ contraintes.
- Fournit une reformulation exacte sur chaque point entier
⇒ Un solveur standard peut le résoudre directement avec un branch-and-bound basé sur la relaxation continue.
- La taille de la reformulation augmente avec le nombre de contraintes quadratiques.
⇒ Nécessité d'ajouter n variables et $4n$ contraintes supplémentaires pour chaque contrainte quadratique.

Reformulation linéaire de produits mixtes

Linéarisation d'un produit continu/binaire

On considère le problème quadratique en variables binaires et continues :

$$(P_{0,1,R}) \left\{ \begin{array}{l} \min \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} q_{ij} y_i x_j \\ \text{s.t.} \\ x_j \in \{0, 1\} \quad j \in \mathcal{J} \\ 0 \leq y_i \leq u_i \quad i \in \mathcal{I} \end{array} \right.$$

La non convexité des fonctions vient des produits de variables $y_i x_j$

Linéarisation d'un produit continu/binaire

On considère le problème quadratique en variables binaires et continues :

$$(P_{0,1,R}) \left\{ \begin{array}{l} \min \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} q_{ij} y_i x_j \\ \text{s.t.} \\ x_j \in \{0, 1\} \quad j \in \mathcal{J} \\ 0 \leq y_i \leq u_i \quad i \in \mathcal{I} \end{array} \right.$$

La non convexité des fonctions vient des produits de variables $y_i x_j$

Linéarisation d'un produit continu/binaire

- 1 Remplacer chaque produit $y_i x_j$ par une nouvelle variable z_{ij}
- 2 Ajouter la contrainte $z_{ij} = y_i x_j$, pour forcer l'équivalence avec le problème de départ.

Réécriture convexe de $z_{ij} = y_i x_j$

Théorème

$$z_{ij} = y_i x_j \text{ et } x_j \in \{0, 1\} \Leftrightarrow \begin{cases} z_{ij} \leq y_i \\ z_{ij} \leq u_i x_j \\ z_{ij} \geq y_i + u_i (x_j - 1) \\ z_{ij} \geq 0 \end{cases}$$

Réécriture convexe de $z_{ij} = y_i x_j$

Théorème

$$z_{ij} = y_i x_j \text{ et } x_j \in \{0, 1\} \Leftrightarrow \begin{cases} z_{ij} \leq y_i \\ z_{ij} \leq u_i x_j \\ z_{ij} \geq y_i + u_i (x_j - 1) \\ z_{ij} \geq 0 \end{cases}$$

Preuve :

- si $x_i = 0$ et/ou $x_j = 0$, on a $y_{ij} \leq x_i = 0$ et $y_{ij} \geq 0 \Leftrightarrow y_{ij} = 0$.

Réécriture convexe de $z_{ij} = y_i x_j$

Théorème

$$z_{ij} = y_i x_j \text{ et } x_j \in \{0, 1\} \Leftrightarrow \begin{cases} z_{ij} \leq y_i \\ z_{ij} \leq u_i x_j \\ z_{ij} \geq y_i + u_i(x_j - 1) \\ z_{ij} \geq 0 \end{cases}$$

Preuve :

- si $x_i = 0$ et/ou $x_j = 0$, on a $y_{ij} \leq x_i = 0$ et $y_{ij} \geq 0 \Leftrightarrow y_{ij} = 0$.
- si $x_i = x_j = 1$, on a $y_{ij} \leq x_i = 1$ et $y_{ij} \geq x_i + x_j - 1 = 1 \Leftrightarrow y_{ij} = 1$.

Reformulation linéaire de produits de variables entières

Linéarisation de $y_{ij} = x_i x_j$

- Décomposition binaire de $x_j = \sum_{k=0}^{\lfloor \log(u_j) \rfloor} 2^k t_{jk}$
avec $t_{jk} \in \{0, 1\}$, ($j \in \mathcal{J}, k = 0, \dots, \lfloor \log(u_j) \rfloor$).

Linéarisation de $y_{ij} = x_i x_j$

- Décomposition binaire de $x_j = \sum_{k=0}^{\lfloor \log(u_j) \rfloor} 2^k t_{jk}$
avec $t_{jk} \in \{0, 1\}$, ($j \in \mathcal{J}$, $k = 0, \dots, \lfloor \log(u_j) \rfloor$).

- $y_{ij} = x_i x_j = x_i \left(\sum_{k=0}^{\lfloor \log(u_j) \rfloor} 2^k t_{jk} \right) = \sum_{k=0}^{\lfloor \log(u_j) \rfloor} 2^k x_i t_{jk}$

Linéarisation de $y_{ij} = x_i x_j$

- Décomposition binaire de $x_j = \sum_{k=0}^{\lfloor \log(u_j) \rfloor} 2^k t_{jk}$
avec $t_{jk} \in \{0, 1\}$, ($j \in \mathcal{J}$, $k = 0, \dots, \lfloor \log(u_j) \rfloor$).

- $y_{ij} = x_i x_j = x_i \left(\sum_{k=0}^{\lfloor \log(u_j) \rfloor} 2^k t_{jk} \right) = \sum_{k=0}^{\lfloor \log(u_j) \rfloor} 2^k x_i t_{jk}$

- Ajout de variables z_{ijk} modélisant les produits $x_i t_{jk}$.

$$x_i t_{jk} = z_{ijk} \iff \begin{cases} z_{ijk} \leq t_{jk} u_i \\ z_{ijk} \leq x_i \\ z_{ijk} \geq x_i - u_i (1 - t_{jk}) \\ z_{ijk} \geq 0 \end{cases}$$

Linéarisation de $y_{ij} = x_i x_j$

- Décomposition binaire de $x_j = \sum_{k=0}^{\lfloor \log(u_j) \rfloor} 2^k t_{jk}$
avec $t_{jk} \in \{0, 1\}$, ($j \in \mathcal{J}, k = 0, \dots, \lfloor \log(u_j) \rfloor$).

- $y_{ij} = x_i x_j = x_i \left(\sum_{k=0}^{\lfloor \log(u_j) \rfloor} 2^k t_{jk} \right) = \sum_{k=0}^{\lfloor \log(u_j) \rfloor} 2^k x_i t_{jk}$

- Ajout de variables z_{ijk} modélisant les produits $x_i t_{jk}$.

$$x_i t_{jk} = z_{ijk} \iff \begin{cases} z_{ijk} \leq t_{jk} u_i \\ z_{ijk} \leq x_i \\ z_{ijk} \geq x_i - u_i (1 - t_{jk}) \\ z_{ijk} \geq 0 \end{cases}$$

- On obtient $y_{ij} = \sum_{k=0}^{\lfloor \log(u_j) \rfloor} 2^k z_{ijk}$ (Deux décompositions possibles)

Relaxation linéaire de produits de variables continues

Les enveloppes de McCormick (1976)

- Considérons l'égalité non convexe $y_{ij} = x_i x_j$ avec $l_i \leq x_i \leq u_i$ et $l_j \leq x_j \leq u_j$
- On a ici 4 inégalités valides :
 - ▶ $x_i - l_i \geq 0$,
 - ▶ $u_i - x_i \geq 0$,
 - ▶ $x_j - l_j \geq 0$,
 - ▶ $u_j - x_j \geq 0$.
- En utilisant la méthode RLT qui consiste à :
 - ▶ multiplier les contraintes entre elles
 - ▶ linéariser les inégalités obtenues

Nous obtenons les inégalités de McCormick.

Les enveloppes de McCormick (1976)

- Considérons l'égalité non convexe $y_{ij} = x_i x_j$ avec $l_i \leq x_i \leq u_i$ et $l_j \leq x_j \leq u_j$
- On obtient l'enveloppe convexe des produits de deux variables :

$$\left\{ \begin{array}{l} y_{ij} = x_i x_j \\ l_i \leq x_i \leq u_i \\ l_j \leq x_j \leq u_j \end{array} \right. \xrightarrow{\text{relax}} (\mathcal{MC}) \left\{ \begin{array}{l} y_{ij} \leq u_j x_i + l_i x_j - u_j l_i \\ y_{ij} \leq u_i x_j + l_j x_i - u_i l_j \\ y_{ij} \geq u_j x_i + u_i x_j - u_i u_j \\ y_{ij} \geq l_j x_i + l_i x_j - l_i l_j \end{array} \right.$$

- **Remarque** : on reconnaît les inégalités de Fortet lorsque $u_i = u_j = 1$ et $l_i = l_j = 0$.