Elements d'Optimisation Mathématique

Olivier Brun

brun@laas.fr

LAAS-CNRS
7 Av. Colonel Roche, 31077 Toulouse, France.

INSA, 2024.

Outline

Introduction

- **Objectifs**
  - Rappeler quelques éléments de base de la théorie de l'optimisation.
  - La présentation évite de rentrer dans les détails mathématiques et se contente d'introduire quelques notions nécessaires à la bonne compréhension du cours.
  - Optimisation convexe, programmation linéaire, programmation linéaire mixte, dualité en programmation linéaire.

Introduction

- **Problème d'optimisation :**

$$\begin{aligned} \text{Minimiser} \quad & f(\mathbf{x}) \\ \text{s.t} \quad & \mathbf{x} \in \mathcal{K} \end{aligned}$$

- Maximiser $f(\mathbf{x})$ est équivalent à minimiser $-f(\mathbf{x})$
- $f(x)$: fonction objectif,
- $\mathcal{K}$: ensemble des solutions admissibles
- **Optimalité :** $\mathbf{x}^* \in \mathcal{K}$ est optimal si $f(\mathbf{x}^*) \leq f(\mathbf{x}), \forall \mathbf{x} \in \mathcal{K}$.
  - Optimum local si l'inégalité n'est vraie que dans un voisinage de $\mathbf{x}^*$.
- Besoin d'hypothèses : différentiabilité et convexité.

Optimisation Convexe

Ensembles convexes

- Un ensemble $\mathcal{K} \subset \mathbb{R}^n$ est dit convexe si le segment joignant deux points $\mathbf{x}, \mathbf{y} \in \mathcal{K}$ appartient à $\mathcal{K}$

$$\alpha\,\mathbf{x} + (1-\alpha)\,\mathbf{y} \in \mathcal{K} \quad \forall \alpha \in [0,1],\ \forall \mathbf{x}, \mathbf{y} \in \mathcal{K}$$

  - Exemples : boule unité $\mathcal{K} = \{\mathbf{x} \in \mathbb{R}^n \ : \ \|\mathbf{x}\| \leq 1\}$, ellipsoides, polyhèdres,...
  - Les ensembles convexes de $\mathbb{R}$ sont les intervalles.
  - L'intersection d'ensembles convexes est un ensemble convexe.

Fonctions convexes

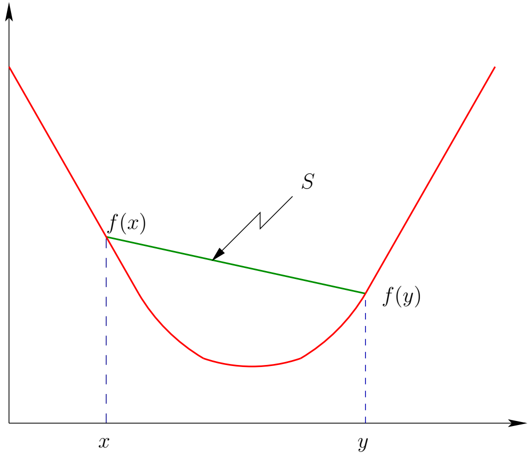
- Etant donné un ensemble convexe $\mathcal{K} \subset \mathbb{R}^n$, une fonction $f(\mathbf{x}) : \mathcal{K} \to \mathbb{R}$ est dite convexe sur $\mathcal{K}$ si

$$f(\alpha\,\mathbf{x} + (1-\alpha)\,\mathbf{y}) \leq \alpha\, f(\mathbf{x}) + (1-\alpha)\, f(\mathbf{y}) \quad \forall \alpha \in (0,1),\ \forall \mathbf{x}, \mathbf{y} \in \mathcal{K}$$

  - On dit que f est strictement convexe si l'inégalité est stricte.
  - Exemple : les fonctions linéaires $f(\mathbf{x}) = \sum_i c_i x_i$ sont convexes, $e^{\mathbf{x}}$ est strictement convexe...
  - L'addition, la multiplication par un scalaire et l'opérateur max préservent la convexité.

Fonctions convexes

- **Fonctions convexes** : tout segment S joignant les points $(\mathbf{x}, f(\mathbf{x}))$ et $(\mathbf{y}, f(\mathbf{y}))$ se trouve “au dessus” de la courbe de f.



- **Problème d'optimisation :**

$$\begin{aligned} \text{Minimiser} \quad & f(\mathbf{x}) \\ \text{s.t} \quad & \mathbf{x} \in \mathcal{K} \end{aligned}$$

- **Optimalité :** un point admissible $\mathbf{x}^* \in \mathcal{K}$ est dit optimal si $f(\mathbf{x}^*) \leq f(\mathbf{x}), \forall \mathbf{x} \in \mathcal{K}$.
  - Unicité du minimum global si f strictement convexe.

- **Optimisation convexe :**
  - $\mathcal{K}$ est un ensemble fermé et convexe,
  - f est convexe et continuellement différentiable sur $\mathcal{K}$.
  - Tout minimum local est un minimum global.

Gradient et matrice hessienne

- Gradient de f au point $\mathbf{x}$: $\nabla f(\mathbf{x}) = \left(\frac{\partial f}{\partial x_i}(\mathbf{x})\right)^T_{i=1,\ldots,n}$

- Produit scalaire des vecteurs $\mathbf{d}$ et $\nabla f(\mathbf{x})$:

$$\mathbf{d}^T \nabla f(\mathbf{x}) = \sum_{i=1}^{n} \frac{\partial f}{\partial x_i}(\mathbf{x})\ d_i$$

- Matrice hessienne de f en $\mathbf{x}$: $\mathbf{H}(\mathbf{x}) = \left(\frac{\partial^2 f}{\partial x_i \partial x_j}(\mathbf{x})\right)_{i=1,j}$

- f est convexe ssi H est semi-définie positive ($\mathbf{x}^T\mathbf{H}\mathbf{x} \geq 0$)
- Exemple : $f(x_1, x_2) = -\log(1 - x_1 - x_2) - \log(x_1) - \log(x_2)$

Conditions d'optimalité

- Développement en série de Taylor (premier ordre)

$$\mathbf{y} = \mathbf{x} + \alpha\mathbf{d} \implies f(\mathbf{y}) = f(\mathbf{x}) + \alpha\mathbf{d}^T\nabla f(\mathbf{x}) + o(\alpha\|\mathbf{d}\|)$$

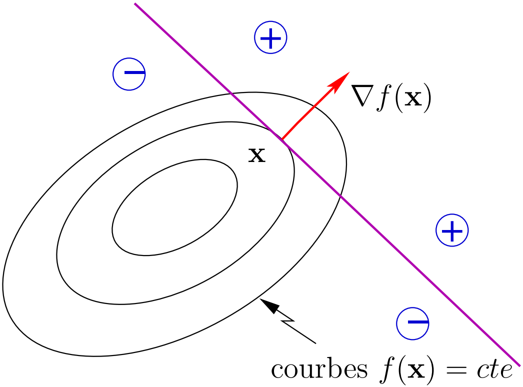
pour toute direction $d \in \mathbb{R}^n$.

- Conséquence (pour $\alpha > 0$ suffisamment petit) :

$$f(\mathbf{y}) < f(\mathbf{x}) \iff \mathbf{d}^T\nabla f(\mathbf{x}) < 0$$

Conditions d'optimalité (2)

- La direction **d** est une direction de
  - maximisation si **d** forme un angle aigu avec $\nabla f(\mathbf{x})$: $\mathbf{d}^T \nabla f(\mathbf{x}) > 0$
  - minimisation si **d** forme un angle obtu avec $\nabla f(\mathbf{x})$: $\mathbf{d}^T \nabla f(\mathbf{x}) < 0$
- Le gradient est normal aux courbes iso-coûts $f(\mathbf{x}) = cte$.



Conditions d'optimalité (3)

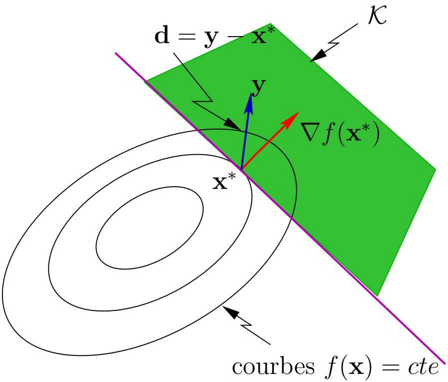
- **Direction admissible** : la direction $\mathbf{d}$ est admissible en $\mathbf{x} \in \mathcal{K}$ ssi il existe $\alpha > 0$ tel que $\mathbf{y} = \mathbf{x} + \alpha \mathbf{d} \in \mathcal{K}$.
- **Principe du minimum** : $\mathbf{x}^* \in \mathcal{K}$ est une solution optimale ssi

$$(\mathbf{y} - \mathbf{x}^*)^T \nabla f(\mathbf{x}^*) \geq 0 \quad \forall y \in \mathcal{K}$$

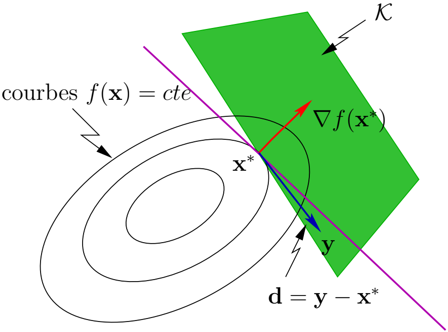
  - Toute direction admissible $\mathbf{d} = \mathbf{y} - \mathbf{x}$ est une direction de maximisation, i.e. elle forme un angle aigu avec le gradient.
  - Cette condition est suffisante grâce à la convexité.
  - Si $\mathcal{K} = \mathbb{R}^n$, on retrouve $\nabla f(\mathbf{x}^*) = 0$.

- Idée de base des méthodes de direction admissible (cf. cours optimisation non-linéaire du routage).

Conditions d'optimalité (4)



$\mathbf{x}^* =$ minimum global.



$\mathbf{x}^* \neq$ minimum global.

Conditions d'optimalité de Karush-Kuhn-Tucker

- L'ensemble $\mathcal{K}$ des solutions admissibles est défini par :

$$\mathbf{x} \in \mathcal{K} \iff \begin{cases} h_i(\mathbf{x}) = 0 & i = 1, \ldots, k \\ g_j(\mathbf{x}) \leq 0 & j = 1, \ldots, m \end{cases}$$

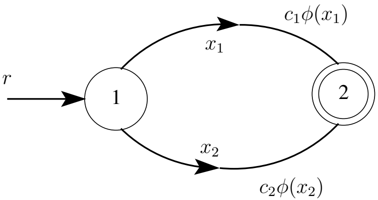
Theorem (Conditions KKT)

Si $\mathbf{x}^*$ est un minimum local, il existe des multiplicateurs $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_k) \in \mathbb{R}^k$ et $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m) \in \mathbb{R}_+^m$ tels que

$$\begin{aligned} \nabla f(\mathbf{x}^*) &= -\sum_i \mu_i \nabla h_i(\mathbf{x}^*) - \sum_j \lambda_j \nabla g_j(\mathbf{x}^*) \\ h_i(\mathbf{x}^*) &= 0 \quad i = 1, \ldots, k \\ g_j(\mathbf{x}) &\leq 0 \quad j = 1, \ldots, m \\ \lambda_j g_j(\mathbf{x}^*) &= 0 \quad j = 1, \ldots, m \end{aligned}$$

Exemple

- **Problème de routage sur deux liens**



$$\begin{aligned} \text{Min}_x \quad & f(x_1, x_2) = c_1\phi(x_1) + c_2\phi(x_2) \\ \text{s.t.} \quad & \\ & x_1 + x_2 = r \\ & x_1 \geq 0, x_2 \geq 0 \end{aligned}$$

- $h(x_1, x_2) = x_1 + x_2 - r = 0$ et $g_i(x_1, x_2) = -x_i \leq 0$, $i = 1, 2$
- Il existe μ, $\lambda_1 \geq 0$ et $\lambda_2 \geq 0$

$$\frac{\partial f}{\partial x_i}(\mathbf{x}^*) = -\mu \frac{\partial h}{\partial x_i}(\mathbf{x}^*) - \lambda_1 \frac{\partial g_1}{\partial x_i}(\mathbf{x}^*) - \lambda_2 \frac{\partial g_2}{\partial x_i}(\mathbf{x}^*), \quad i = 1, 2$$

ce qui donne, en posant $\mu' = -\mu$, $c_i\phi'(x_i^*) \geq \mu' + \lambda_i$.

Exemple

- On sait aussi que $\lambda_i g_i(x^*) = -\lambda_i x_i^* = 0$ et donc

$$x_i^* > 0 \text{ si et seulement si } \lambda_i = 0$$

- On conclu que $c_i \phi'(x_i^*) \geq \mu'$ avec égalité ssi $x_i^* > 0$.
- En supposant $c_1 < c_2$:
  - La solution optimale est $x_1^* = r, x_2^* = 0$ si

$$c_2 \phi'(0) > c_1 \phi'(r)$$

  - Sinon, la solution optimale est obtenue en résolvant

$$c_1 \phi'(x_1^*) = c_2 \phi'(r - x_1^*)$$

Programmation Linéaire

Programmation Linéaire

- Un programme linéaire est un problème d'optimisation convexe de la forme

$$\text{Minimiser } z = \sum_{i=1}^{n} c_i\, x_i$$

$$\text{s.t}$$

$$\sum_{i=1}^{n} a_{ji} x_i \geq b_j \quad j = 1, \ldots, m,$$

$$x_i \geq 0 \quad i = 1, \ldots, n.$$

ou sous forme matricielle

$$\text{Minimiser } z = \mathbf{c}^T \mathbf{x}$$

$$\text{s.t } \mathbf{A}\,\mathbf{x} \geq \mathbf{b}, \mathbf{x} \geq 0.$$

Forme standard

- Maximiser $\sum_{i=1}^{n} c_i\, x_i$ est équivalent à minimiser $-\sum_{i=1}^{n} c_i\, x_i$.
- Une contrainte d'égalité $\sum_i a_{ji} x_i = b_j$ est équivalente à

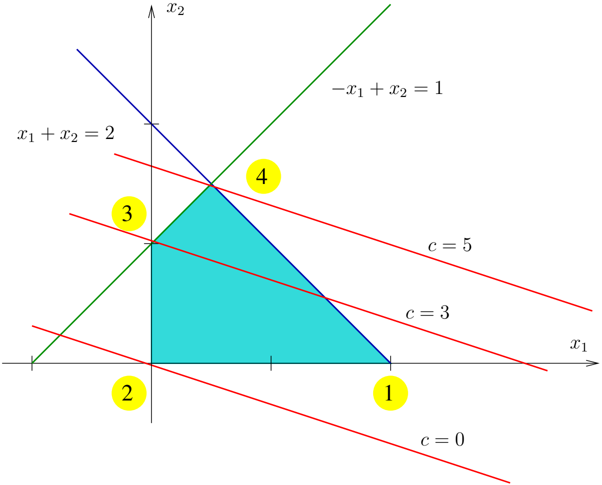
$$\sum_i a_{ji} x_i \leq b_j,$$
$$-\sum_i a_{ji} x_i \leq -b_j.$$

- Si la variable x_i peut être négative, on peut écrire

$$x_i = x_i^+ - x_i^-$$

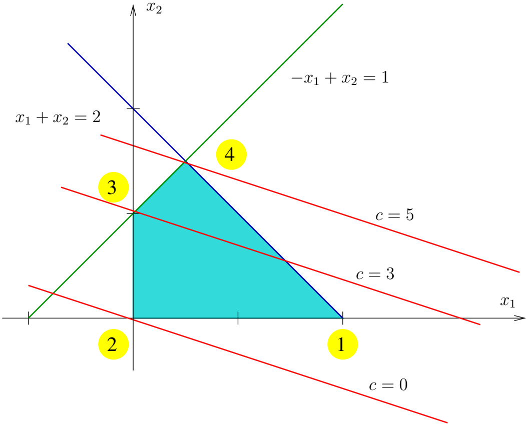
avec $x_i^+ \geq 0$ et $x_i^- \geq 0$.

$$\text{Max} \quad z = x_1 + 3x_2$$
$$\text{s.t}$$
$$-x_1 + x_2 \leq 1$$
$$x_1 + x_2 \leq 2$$
$$x_1 \geq 0, x_2 \geq 0$$



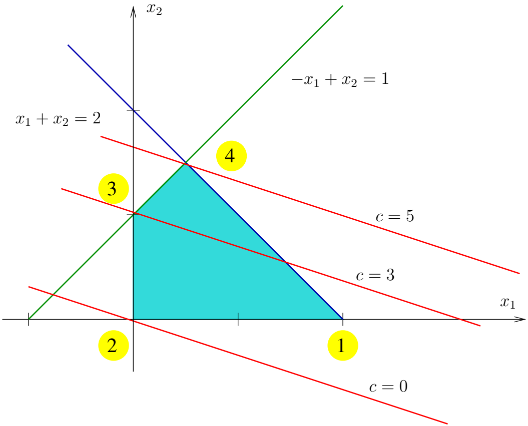
Propriétés fondamentales (2)

- Ensemble des solutions : intersection des demi-plans satisfaisant l'inégalité linéaire.
- Point numérotés : points extrémaux.
- Les courbes iso-coûts $c = x_1 + 3x_2$ sont // les unes aux autres.



Propriétés fondamentales (3)

- La valeur max de z correspond à la plus grande valeur de c pour laquelle la ligne iso-coût a au moins un point commun avec l'ensemble des solutions : $c = 5$, $x_1 = \frac{1}{2}$, $x_2 = \frac{3}{2}$.



Propriété fondamentale
L'optimum d'un PL est toujours atteint en un point extrémal.

Les différents cas de figure

- **Quatre possibilités**
  - **Pas de solution** : si on remplace $x_1 + x_2 \leq 2$ par $x_1 + x_2 \leq -1$ l'ensemble solution est vide.
  - **Solution non bornée** : si on enlève $x_1 + x_2 \leq 2$, alors $\max z = +\infty$
  - **Solution unique** : elle correspond à un point extrémal.
  - **Infinité de solution** : si on remplace $z = x_1 + 3x_2$ par $z = 2(x_1 + x_2)$ alors les lignes iso-coûts sont parallèles à la contrainte $x_1 + x_2 = 2$ et le maximum est atteint en tout point de ce segment.

La méthode du Simplex

- **Généralisation en dimension n de ce qu'on a vu dans $\mathbb{R}^2$**
  - Phase 1 : obtention d'une solution admissible de base (si existence)
  - Phase 2 : détermination de l'optimum global.
- **Méthode itérative :**
  - Visite consécutivement des solutions de base (points extrêmaux du polyhédre),
  - Diminue le coût à chaque itération,
  - Identifie le minimum global quand il est atteint.
- **Algorithme implémentés dans de nombreux solveurs linéaires** : CPLEX, Gurobi, lp solve, matlab, etc.
- **Extrêmement efficace** : on peut résoudre des problèmes avec des milliers de variables et de contraintes en des temps calcul raisonnables.

Programmation Linéaire Mixte

- Exemple des PL mixtes avec variables 0-1 :

$$\begin{array}{rl} \text{Minimiser} & z = \sum_{i=1}^{n} c_i \, x_i \\ \text{s.t} & \\ & \sum_i a_{ji} x_i \geq b_j \quad j = 1, \ldots, m \\ & x_i \in \{0, 1\} \quad i = 1, \ldots, k \\ & x_i \geq 0 \quad i = k+1, \ldots, n \end{array}$$

- Solution partielle :
  - $N_0 \subset \{1, \ldots, k\}$: variables binaires valant 0,
  - $N_1 \subset \{1, \ldots, k\}$: variables binaires valant 1,
  - $N_U \subset \{1, \ldots, k\}$: variables binaires non fixées.

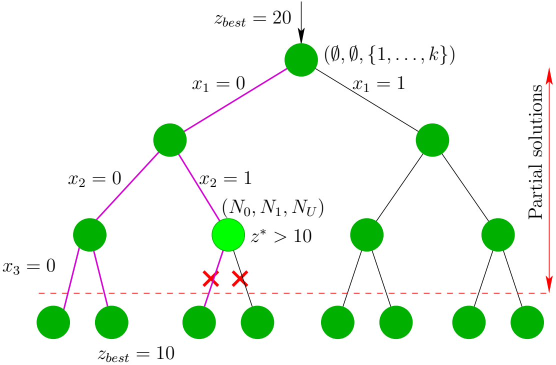
Borne inférieure

- Borne inférieure z^* sur le coût d'une solution partielle obtenue en résolvant le problème relaxé :

$$
\begin{array}{rc}
\text{Minimiser} & z = \sum_{i=1}^{n} c_i \, x_i \\
\text{s.t} & \\
& \sum_i a_{ji} x_i \geq b_j \quad j = 1, \ldots, m \\
& x_i \geq 0 \quad i = k+1, \ldots, n \\
& 0 \leq x_i \leq 1 \quad i \in N_U \\
& x_i = 0 \quad i \in N_0 \\
& x_i = 1 \quad i \in N_1
\end{array}
$$

- Toute solution complète **x** obtenue à partir de la solution partielle (N_0, N_1, N_U) en fixant les valeurs des variables $j \in N_U$ à 0 ou à 1 aura un coût supérieur ou égal à z^*.

- **Principe :**
  - Explorer l'arbre représentant l'ensemble des solutions,
  - Mettre à jour la borne sup quand une solution complète améliorante est découverte,
  - Calculer une borne inférieure en chaque noeud et élaguer l'arbre si $z^* \geq z^{best}$.



- **Algorithme du Branch-and-Bound :**

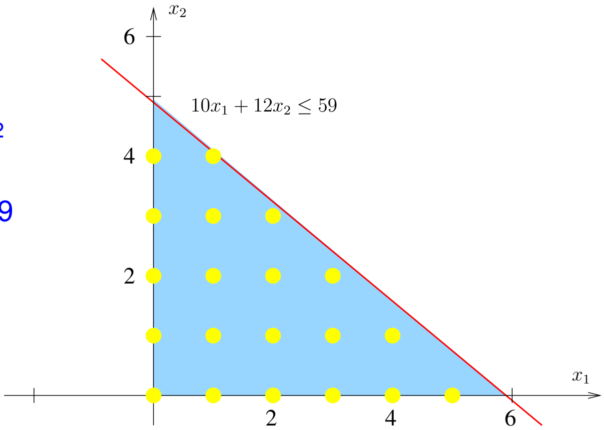
procedure BRANCH-AND-BOUND((N_U, N_0, N_1))
    LinearProgramming($(N_U, N_0, N_1, \mathbf{x}, z^*)$)
    if $N_U = \emptyset$ or $x_i \in \{0,1\}\ \forall i \in N_U$ **then**
        if $z^* < z^{best}$ **then**
            $z^{best} = z^*$ et $\mathbf{x}^{best} = \mathbf{x}$
        end if
    else $\triangleright$ au moins un x_i non binaire
        if $z^* \geq z^{best}$ **then**
            return $\triangleright$ bounding
        else $\triangleright$ branching
            Choisir $i \in N_U$ tel que $x_i \neq 0, 1$
            Branch-and-Bound($N_U - i, N_0 \cup \{i\}, N_1$)
            Branch-and-Bound($N_U - i, N_0, N_1 \cup \{i\}$)
        end if
    end if
end procedure

Branch and Bound (3)

- Généralisation à des variables entières.
- Implémenté par de nombreux solveurs : CPLEX, Gurobi, lp solve, etc.
- Performances limitées à des problèmes de "petites" tailles :
  - Beaucoup de temps passé pour prouver l' optimalité,
  - Incontournable pour évaluer des approximations.
- La borne inférieure obtenue par relaxation linéaire est souvent grossière :
  - L'idée de base du Branch-and-Cut est d'améliorer la borne inférieure en ajoutant des contraintes vérifiées par toute solution entière mais excluant une partie des solutions continues (enveloppe convexe).

Branch and Cut (1)

$$\begin{aligned} \text{Max} \quad & z = 10x_1 + 11x_2 \\ \text{s.t} \quad & \\ & 10x_1 + 12x_2 \leq 59 \\ & x_1, x_2 \in \mathbb{N} \end{aligned}$$



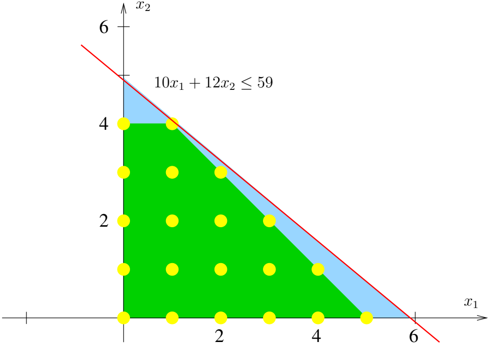
- Problème relaxé (PL) : $x^* = (5.9, 0)$ et $z^* = 59$.
  - En arrondissant : $x^* = (5, 0)$ et $z^* = 50$.
- Problème discret (PLNE) : $x^* = (1, 4)$ et $z^* = 54$.

Branch and Cut (2)

- Polyhédre convexe contenant toutes les solutions entières :

$$x_1 + x_2 \leq 5 \quad 0 \leq x_2 \leq 4 \quad x_1 \geq 0$$

$$\begin{aligned} \text{Max} \quad & z = 10x_1 + 11x_2 \\ \text{s.t} \quad & \\ & x_1 + x_2 \leq 5 \\ & 0 \leq x_2 \leq 4 \\ & x_1 \geq 0 \end{aligned}$$



- Borne inférieure : $x^* = (1, 4)$ et $z^* = 54$.

Dualité

Dualité (1)

- Problème **primal** :

$$
\begin{aligned}
\text{Minimiser } z = \quad & 6x_1 + 4x_2 + 2x_3 \\
\text{s.t} \quad & \\
& 4x_1 + 2x_2 + x_3 \geq 5 & (1)\\
& x_1 + x_2 \geq 3 & (2)\\
& x_2 + x_3 \geq 4 & (3)\\
& x_i \geq 0 \qquad i = 1, 2, 3. &
\end{aligned}
$$

- **Bornes inférieures sur la valeur optimale z^* :**
  - Avec (1) : $z = 6x_1 + 4x_2 + 2x_3 \geq 4x_1 + 2x_2 + x_3 \implies z^* \geq 5$
  - Avec (1) et (2) :
    $z \geq (4x_1 + 2x_2 + x_3) + 2(x_1 + x_2) \implies z^* \geq 5 + 2 \times 3 = 11$
  - Avec (1), (2) et (3) :
    $z \geq (4x_1 + 2x_2 + x_3) + (x_1 + x_2) + (x_2 + x_3) \implies z^* \geq 5 + 3 + 4 = 12$

Dualité (2)

- **Généralisation** : en multipliant (1) par $y_1 \geq 0$, (2) par $y_2 \geq 0$ et (3) par $y_3 \geq 0$, on obtient $z^* \geq 5y_1 + 3y_2 + 4y_3$ si la condition suivante est vérifiée :

$$\begin{aligned} 6x_1 + 4x_2 + 2x_3 &\geq y_1(4x_1 + 2x_2 + x_3) + y_2(x_1 + x_2) + y_3(x_2 + x_3) \\ &= x_1(4y_1 + y_2) + x_2(2y_1 + y_2 + y_3) + x_3(y_1 + y_3) \end{aligned}$$

- Problème **dual** :

$$\begin{aligned} \text{Maximiser } w = \quad & 5y_1 + 3y_2 + 4y_3 \\ \text{s.t} \quad & \\ & 4y_1 + y_2 \leq 6 \\ & 2y_1 + y_2 + y_3 \leq 4 \\ & y_1 + y_3 \leq 2 \\ & y_i \geq 0 \qquad i = 1, 2, 3. \end{aligned}$$

- Borne inférieure : $z^* \geq w^*$

Dualité (3)

PRIMAL

$$\text{Min } z = \sum_{i=1}^{n} c_i\, x_i$$

s.t

$$\sum_{i=1}^{n} a_{ji} x_i \geq b_j \quad j = 1, \ldots, m,$$

$$x_i \geq 0 \quad i = 1, \ldots, n.$$

DUAL

$$\text{Max } w = \sum_{j=1}^{m} b_j\, y_j$$

s.t

$$\sum_{j=1}^{m} a_{ji} y_j \leq c_i \quad i = 1, \ldots, n,$$

$$y_j \geq 0 \quad j = 1, \ldots, m.$$

Dualité (4)

- **Dualité faible** : Si $\mathbf{x}$ est une solution admissible du primal et $\mathbf{y}$ une solution admissible du dual, alors $\sum_{i=1}^{n} c_i\, x_i \geq \sum_{j=1}^{m} b_j\, y_j$.
- **Dualité forte** : Si le primal et le dual sont faisables, alors pour toute solution optimale $\mathbf{x}^*$ du primal et pour toute solution optimale $\mathbf{y}^*$ du dual, on a $\sum_{i=1}^{n} c_i\, x_i^* = \sum_{j=1}^{m} b_j\, y_j^*$.
- Exemple : la solution optimale du primal est $x_1^* = 0, x_2^* = 3$ et $x_3^* = 1$. La solution optimale du dual est $y_1^* = 0, y_2^* = 2$ et $y_3^* = 2$. On a bien $6x_1^* + 4x_2^* + 2x_3^* = 14 = 5y_1^* + 3y_2^* + 4y_3^*$
- Conditions de complémentarité :

$$x_i^* > 0 \implies \sum_j a_{j,i} y_j^* = c_i,$$

$$y_j^* > 0 \implies \sum_i a_{j,i} x_i^* = b_j.$$