Optimisation

Programmation linéaire

Louis-Claude Canon
louis-claude.canon@univ-fcomte.fr

Master 2 Informatique – Semestre 9

Plan

Plan

Problème politique

- Un politicien souhaite remporter une élection sur une circonscription.
- La circonscription comprend 3 zones : ville (100 000 électeurs), banlieues (200 000) et campagne (50 000).
- Pour être légitime, le politicien souhaite au moins la moitié des voix dans chaque zone.
- L'objectif est de dépenser le moins possible de frais en publicité.

Données du problème

Nombre de milliers d'électeurs gagnés pour 1000 € dépensés en publicité :

Stratégie	ville	banlieues	campagne
Apocalypse zombie	-2	5	3
Requins avec laser	8	2	-5
Autoroutes pour voitures volantes	0	0	10
Vote des dauphins	10	0	-2

Exemple de solution

On peut dépenser :

- 20 000 € pour la préparation à une apocalypse zombie,
- 0 € pour l'équipement de laser sur des requins,
- 4 000 € pour construire des autoroutes pour voitures volantes,
- 9 000 € pour permettre aux dauphins de voter.

Dans ce cas, on obtient :

- 50 000 électeurs en ville,
- 100 000 électeurs en banlieues,
- 82 000 électeurs en campagne.

Le coût total serait de 33 000 euros.

Exemple de solution

On peut dépenser :

- 20 000 € pour la préparation à une apocalypse zombie,
- 0 € pour l'équipement de laser sur des requins,
- 4 000 € pour construire des autoroutes pour voitures volantes,
- 9 000 € pour permettre aux dauphins de voter.

Dans ce cas, on obtient :

- 50 000 électeurs en ville,
- 100 000 électeurs en banlieues,
- 82 000 électeurs en campagne.

Le coût total serait de 33 000 euros. Est-ce optimal ?

Terminologie

- Il faut formuler mathématiquement le problème (on dit aussi le *modéliser*).
- On a 4 *variables* pour décider les sommes investies : une par stratégie. Ce sera x_1 à x_4 (e.g., x_1 étant le budget pour l'apocalypse zombie).
- On a des *contraintes* sur le nombre d'électeurs obtenu : une par zone de la circonscription. Par exemple pour la ville : $-2x_1 + 8x_2 + 0x_3 + 10x_4 \geq 50$.
- On définit l'*objectif* : minimiser la somme totale investie (ici, $x_1 + x_2 + x_3 + x_4$).

Exemple de programme linéaire

Le programme linéaire ainsi obtenu est :

$$
\begin{array}{lrcrcrcrcr}
\text{minimiser} & x_1 & + & x_2 & + & x_3 & + & x_4 & & \\
\text{sous les contraintes} & -2x_1 & + & 8x_2 & + & 0x_3 & + & 10x_4 & \geq & 50 \\
 & 5x_1 & + & 2x_2 & + & 0x_3 & + & 0x_4 & \geq & 100 \\
 & 3x_1 & - & 5x_2 & + & 10x_3 & - & 2x_4 & \geq & 25 \\
 & & & & & & & x_1, x_2, x_3, x_4 & \geq & 0
\end{array}
$$

Plan

Fonction linéaire

- Soit a_1, a_2, ..., a_n des valeurs réelles.
- Soit x_1, x_2, ..., x_n des variables réelles.
- La *fonction linéaire f* est définie par :

$$f(x_1, x_2, \ldots, x_n) = a_1x_1 + a_2x_2 + \ldots + a_nx_n = \sum_{j=1}^{n} a_jx_j$$

- Exemple : $f(x_1, x_2) = 2x_1 - x_2$.
- Une *égalité linéaire* (avec b une valeur réelle) : $f(x_1, x_2, \ldots, x_n) = b$.
- Les *inégalités linéaires* : $f(x_1, x_2, \ldots, x_n) \leq b$ et $f(x_1, x_2, \ldots, x_n) \geq b$.
- Une *contrainte linéaire* est soit une égalité, soit une inégalité linéaire.

Problème de programmation linéaire

- Un *problème de programmation linéaire* consiste à minimiser ou maximiser une fonction linéaire soumise à des contraintes linéaires.
- Soit n le nombre de variables et m le nombre de contraintes.
- On cherche les valeurs des variables x_1, x_2, ..., x_n pour :

$$
\begin{array}{ll}
\text{maximiser} & \displaystyle\sum_{j=1}^{n} c_j x_j \\
\text{sous les contraintes} & \displaystyle\sum_{j=1}^{n} a_{ij} x_j \leq b_i \text{ pour } i = 1, 2, \ldots, m \\
& \qquad x_j \geq 0 \ \text{ pour } j = 1, 2, \ldots, n
\end{array}
$$

- Les c_j sont les coefficients réels de la *fonction objectif*.
- Les a_{ij} et b_i sont les coefficients réels des *contraintes*.
- La dernière ligne représente les contraintes de non-négativité.

Forme standard

La *forme standard* utilise une forme plus compacte en utilisant une matrice A (qui contient les a_{ij}) et 3 vecteurs colonnes b (b_i), c (c_j) et x (x_j) :

$$\begin{array}{ll} \text{maximiser} & c^T x \\ \text{sous les contraintes} & Ax \leq b \\ & x \geq 0 \end{array}$$

Les bibliothèques de résolution s'appuient souvent sur cette forme.

Solutions

- Soit $\bar{x}$ un ensemble de valeurs réelles pour le vecteur x.
- Si les contraintes sont respectées, $\bar{x}$ est une *solution réalisable* et sa *valeur de l'objectif* est $c^T\bar{x}$.
- S'il n'existe pas de solution avec un meilleur objectif, on dit que c'est la *solution optimale* et qu'elle atteint la *valeur de l'objectif optimale*.

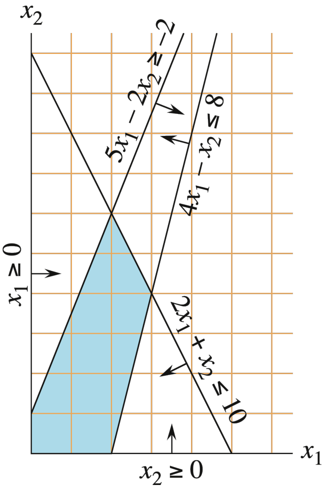
Résolution sur un exemple

On considère le problème de programmation linéaire suivant :

$$
\begin{array}{lrcrcr}
\text{maximiser} & x_1 & + & x_2 & & \\
\text{sous les contraintes} & 4x_1 & - & x_2 & \leq & 8 \\
 & 2x_1 & + & x_2 & \leq & 10 \\
 & 5x_1 & - & 2x_2 & \geq & -2 \\
 & & & x_1, x_2 & \geq & 0
\end{array}
$$

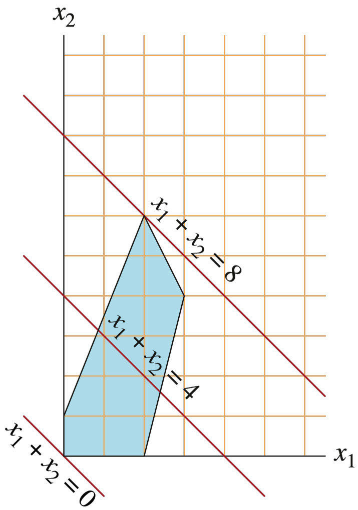
Représentation graphique (1/2)

- Les contraintes sont définies par la matrice $A = \left\{ \begin{array}{cc} 4 & -1 \\ 2 & 1 \\ -5 & 2 \end{array} \right\}$ et le vecteur $b = \{8, 10, 2\}$.
- Les 5 contraintes définissent un polygone, c'est-à-dire une région de réalisabilité (toute solution dans cette région est réalisable).



Représentation graphique (2/2)

- La fonction objectif est $f(x_1, x_2) = x_1 + x_2$ avec $c_1 = c_2 = 1$.
- La solution $\bar{x} = \{0, 0\}$ est réalisable et sa valeur de l'objectif est 0.
- La solution optimale est $\bar{x} = \{2, 6\}$ et la valeur de l'objectif optimale est 8.



Algorithme du simplexe

- L'algorithme du simplexe est très efficace en pratique pour résoudre les problèmes de programmation linéaire (bien que de complexité exponentielle en temps au pire cas).
- Il part d'une solution réalisable ($\bar{x} = 0$ par exemple) et se déplace de sommet en sommet sur le polyèdre (polygone en dimension n) formant la région de réalisabilité.
- Il s'arrête sur la solution optimale.
- D'autres méthodes, de complexité en temps polynomiale, se déplacent à l'intérieur de la région de réalisabilité (méthodes de points intérieurs).
- En pratique, on s'appuie sur un solveur comme CPLEX ou COIN-OR LP qui peuvent traiter des centaines de milliers de variables et de contraintes.

Question

Lequel de ces programmes linéaires n'admet pas de solution réalisable ?

1.
$$\begin{array}{lrcrcr}
\text{minimiser} & -2x_1 & + & 3x_2 & & \\
\text{contraintes} & x_1 & + & x_2 & = & 7 \\
 & x_1 & - & 2x_2 & \leq & 4 \\
 & & & x_1 & \geq & 0
\end{array}$$

2.
$$\begin{array}{lrcrcr}
\text{minimiser} & 2x_1 & + & x_2 & & \\
\text{contraintes} & x_1 & - & x_2 & = & 7 \\
 & 3x_1 & & & \geq & 24 \\
 & & & x_2 & \geq & 0
\end{array}$$

3.
$$\begin{array}{lrcrcr}
\text{minimiser} & 3x_1 & - & 2x_2 & & \\
\text{contraintes} & x_1 & + & x_2 & \leq & 2 \\
 & -2x_1 & - & 2x_2 & \leq & -10 \\
 & & & x_1, x_2 & \geq & 0
\end{array}$$

4.
$$\begin{array}{lrcrcr}
\text{minimiser} & x_1 & - & x_2 & & \\
\text{contraintes} & -2x_1 & + & x_2 & \leq & -1 \\
 & -x_1 & - & 2x_2 & \leq & -2 \\
 & & & x_1, x_2 & \geq & 0
\end{array}$$

Question

Lequel de ces programmes linéaires n'admet pas de solution réalisable ?

1.

$$\begin{array}{lrcrcr} \text{minimiser} & -2x_1 & + & 3x_2 & & \\ \text{contraintes} & x_1 & + & x_2 & = & 7 \\ & x_1 & - & 2x_2 & \leq & 4 \\ & & & x_1 & \geq & 0 \end{array}$$

2.

$$\begin{array}{lrcrcr} \text{minimiser} & 2x_1 & + & x_2 & & \\ \text{contraintes} & x_1 & - & x_2 & = & 7 \\ & 3x_1 & & & \geq & 24 \\ & & & x_2 & \geq & 0 \end{array}$$

3. ✓

$$\begin{array}{lrcrcr} \text{minimiser} & 3x_1 & - & 2x_2 & & \\ \text{contraintes} & x_1 & + & x_2 & \leq & 2 \\ & -2x_1 & - & 2x_2 & \leq & -10 \\ & & & x_1, x_2 & \geq & 0 \end{array}$$

4.

$$\begin{array}{lrcrcr} \text{minimiser} & x_1 & - & x_2 & & \\ \text{contraintes} & -2x_1 & + & x_2 & \leq & -1 \\ & -x_1 & - & 2x_2 & \leq & -2 \\ & & & x_1, x_2 & \geq & 0 \end{array}$$

Plan

Problèmes concernés

- La *recherche opérationnelle* est une discipline qui se base sur la formulation mathématique des problèmes d'optimisation.
- Exemples : gestion d'emploi du temps avec contraintes, organisation de chaînes logistiques, affectation de ressources, etc.
- On peut notamment formuler des problèmes de graphes et de flots comme des programmes linéaires.

Plus courts chemins : description

- Un graphe orienté $G = (V, E)$ avec une pondération sur les arcs $w : E \to \mathbb{R}$.
- Un nœud source $s \in V$ et une destination $t \in V$.
- Soit d_v le poids du plus court chemin de s vers v ($d_s = 0$).
- On cherche d_t.

Plus courts chemins : programme linéaire

$$\begin{array}{ll} \text{maximiser} & d_t \\ \text{sous les contraintes} & d_v \leq d_u + w(u,v) \text{ pour tout arc } (u,v) \in E \\ & d_s = 0 \end{array}$$

- Si c'était une minimisation, $d_t = -\infty$ serait un solution optimale.
- Astuce : c'est l'inégalité triangulaire qui garantit une valeur maximale aux variables.
- Nombre de variables : $|V|$.
- Nombre de contraintes : $|E| + 1$.

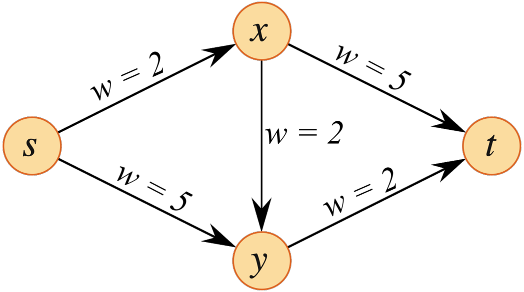
Plus courts chemins : programme linéaire

$$
\begin{array}{ll}
\text{maximiser} & d_t \\
\text{sous les contraintes} & d_v \leq d_u + w(u,v) \text{ pour tout arc } (u,v) \in E \\
& d_s = 0
\end{array}
$$

- Si c'était une minimisation, $d_t = -\infty$ serait un solution optimale.
- Astuce : c'est l'inégalité triangulaire qui garantit une valeur maximale aux variables.
- Nombre de variables : $|V|$.
- Nombre de contraintes : $|E| + 1$.

Plus courts chemins : exemple

- $d_s = 0$.
- $d_x \leq d_s + w(s, x) = 2$.
- $d_y \leq d_s + w(s, y) = 5$ et
  $d_y \leq d_x + w(x, y) \leq 4$.
- $d_t \leq d_x + w(x, t) \leq 7$ et
  $d_t \leq d_y + w(y, t) \leq 6$.



Flot maximum : description

- Un graphe orienté $G = (V, E)$ avec des capacités sur les arcs $c : V \times V \to \mathbb{R}$.
- Un nœud source $s \in V$ et un puits $t \in V$.
- Soit $f : V \times V \to \mathbb{R}$ le flot de chaque arc.
- On cherche à maximiser le flot total provenant de la source.
- Le flot doit respecter les contraintes de capacité et se conserver.

Flot maximum : programme linéaire

$$
\begin{array}{lrl}
\text{maximiser} & \displaystyle\sum_{v \in V} f_{sv} - \sum_{v \in V} f_{vs} & \\
\text{sous les contraintes} & f_{uv} \leq c(u, v) & \text{pour chaque } u, v \in V \\
& \displaystyle\sum_{v \in V} f_{uv} = \sum_{v \in V} f_{vu} & \text{pour chaque } u \in V \setminus \{s, t\} \\
& f_{uv} \geq 0 & \text{pour chaque } u, v \in V
\end{array}
$$

- La première contrainte garantit les capacités.
- La seconde contrainte concerne la conservation du flot.
- Nombre de variables : $|V|^2$.
- Nombre de contraintes : $2|V|^2 + |V| - 2$.

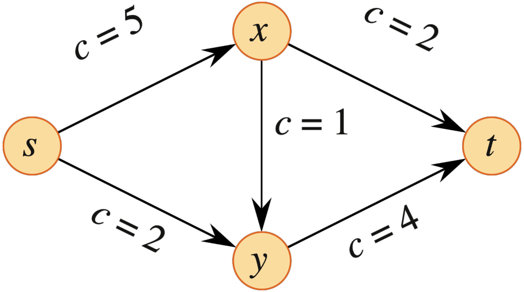
Flot maximum : programme linéaire

$$
\begin{array}{lrl}
\text{maximiser} & \displaystyle\sum_{v\in V} f_{sv} - \sum_{v\in V} f_{vs} & \\
\text{sous les contraintes} & f_{uv} \leq c(u,v) & \text{pour chaque } u, v \in V \\
& \displaystyle\sum_{v\in V} f_{uv} = \sum_{v\in V} f_{vu} & \text{pour chaque } u \in V \setminus \{s,t\} \\
& f_{uv} \geq 0 & \text{pour chaque } u, v \in V
\end{array}
$$

- La première contrainte garantit les capacités.
- La seconde contrainte concerne la conservation du flot.
- Nombre de variables : $|V|^2$.
- Nombre de contraintes : $2|V|^2 + |V| - 2$.

Flot maximum : exemple

- $\bar{f_{sx}} = 3$.
- $\bar{f_{sy}} = 2$.
- $\bar{f_{xy}} = 1$.
- $\bar{f_{xt}} = 2$.
- $\bar{f_{yt}} = 3$.



Flot maximum à coût minimal : description

- Les deux problèmes précédents peuvent déjà être résolus par des algorithmes existants.
- L'approche par la programmation linéaire permet traiter de nouveaux problèmes.
- On modifie le problème du flot maximum en rajoutant des coût sur les arcs ($a : E \to \mathbb{R}$).
- On souhaite alors minimiser le coût ($\sum_{(u,v)\in E} a(u,v) \times f_{uv}$) en garantissant un flot minimum d.

Flot maximum à coût minimal : programme linéaire

$$
\begin{aligned}
\text{minimiser} \quad & \sum_{(u,v)\in E} a(u,v) \times f_{uv} \\
\text{sous les contraintes} \quad & f_{uv} \leq c(u,v) \quad \text{pour chaque } u, v \in V \\
& \sum_{v\in V} f_{uv} - \sum_{v\in V} f_{vu} = 0 \quad \text{pour chaque } u \in V \setminus \{s,t\} \\
& \sum_{v\in V} f_{sv} - \sum_{v\in V} f_{vs} = d \\
& f_{uv} \geq 0 \quad \text{pour chaque } u, v \in V
\end{aligned}
$$

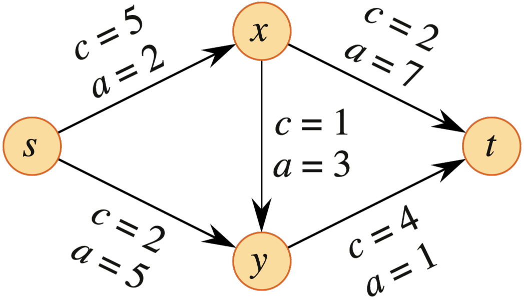
- Nombre de variables : $|V|^2$.
- Nombre de contraintes : $2|V|^2 + |V| - 1$.

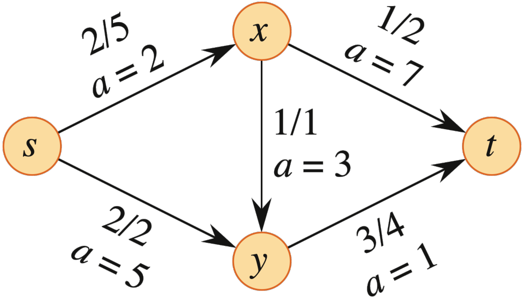
Flot maximum à coût minimal : programme linéaire

$$
\begin{aligned}
\text{minimiser} \quad & \sum_{(u,v)\in E} a(u,v) \times f_{uv} \\
\text{sous les contraintes} \quad & f_{uv} \leq c(u,v) && \text{pour chaque } u,v \in V \\
& \sum_{v\in V} f_{uv} - \sum_{v\in V} f_{vu} = 0 && \text{pour chaque } u \in V \setminus \{s,t\} \\
& \sum_{v\in V} f_{sv} - \sum_{v\in V} f_{vs} = d \\
& f_{uv} \geq 0 && \text{pour chaque } u,v \in V
\end{aligned}
$$

- Nombre de variables : $|V|^2$.
- Nombre de contraintes : $2|V|^2 + |V| - 1$.

Flot maximum : exemple





Flot multi-produits : contexte

- L'entreprise Max & Fils produit et livre des tissus, des pots et des fleurs artificielles.
- Chaque type de produits est fabriqué dans une usine spécifique puis expédié chaque jour vers son entrepôt.
- Les différents types de produit doivent se partager le réseau de transport.

Flot multi-produits : description

- Un graphe orienté $G = (V, E)$ avec des capacités sur les arcs $c : E \to \mathbb{R}$.
- Il y a k produits différents chacun spécifié par s_i (son usine), t_i (son entrepôt) et d_i (la quantité de produit, c'est-à-dire le flot souhaité).
- Soit f_{iuv} le flot pour chaque produit sur chaque arc.
- Le flot total est la somme des flots de chaque produit $f_{uv} = \sum_{i=1}^{k} f_{iuv}$.
- On cherche à déterminer si une solution existe.

Flot multi-produits : programme linéaire

$$
\begin{array}{lrl}
\text{minimiser} & 0 & \\
\text{sous les contraintes} & \displaystyle\sum_{i=1}^{k} f_{iuv} \leq c(u,v) & \text{pour chaque } u, v \in V \\
& \displaystyle\sum_{v\in V} f_{iuv} - \sum_{v\in V} f_{ivu} = 0 & \text{pour chaque } i = 1, 2, \ldots, k \text{ et} \\
& & \text{pour chaque } u \in V \setminus \{s, t\} \\
& \displaystyle\sum_{v\in V} f_{i,s_i,v} - \sum_{v\in V} f_{i,v,s_i} = d_i & \text{pour chaque } i = 1, 2, \ldots, k \\
& f_{iuv} \geq 0 & \text{pour chaque } i = 1, 2, \ldots, k \text{ et} \\
& & \text{pour chaque } u, v \in V
\end{array}
$$

Question

Lequel de ces programmes linéaires résout le problème de flot maximum ?

1.

$$
\begin{array}{lll}
\text{maximiser} & \displaystyle\sum_{v \in V} f_{sv} & \\
\text{contraintes} & f_{uv} \leq c(u,v) & \forall u, v \in V \\
& \displaystyle\sum_{v \in V} f_{uv} = \sum_{v \in V} f_{vu} & \forall u \in V \setminus \{s,t\} \\
& f_{uv} \geq 0 & \forall u, v \in V
\end{array}
$$

2.

$$
\begin{array}{lll}
\text{minimiser} & \displaystyle\sum_{(u,v) \in E} a(u,v) \times f_{uv} & \\
\text{contraintes} & f_{uv} \leq c(u,v) & \forall u, v \in V \\
& \displaystyle\sum_{v \in V} f_{uv} - \sum_{v \in V} f_{vu} = 0 & \forall u \in V \setminus \{s,t\} \\
& \displaystyle\sum_{v \in V} f_{sv} - \sum_{v \in V} f_{vs} = d & \\
& f_{uv} \geq 0 & \forall u, v \in V
\end{array}
$$

3.

$$
\begin{array}{lll}
\text{maximiser} & d_t & \\
\text{contraintes} & d_v \leq d_u + w(u,v) & \forall (u,v) \in E \\
& d_s = 0 &
\end{array}
$$

4.

$$
\begin{array}{lll}
\text{minimiser} & \displaystyle\sum_{(u,v) \in E} c(u,v) \times d_{uv} & \\
\text{contraintes} & d_{uv} - z_u + z_v \geq 0 & \forall (u,v) \in E, u \neq s, v \neq t \\
& d_{sv} + z_v \geq 1 & \forall (s,v) \in E, v \neq t \\
& d_{ut} - z_u \geq 0 & \forall (u,t) \in E, u \neq s \\
& d_{st} \geq 1 & \text{si } (s,t) \in E \\
& d_{uv} \geq 0 & \forall (u,v) \in E
\end{array}
$$

Question

Lequel de ces programmes linéaires résout le problème de flot maximum ?

1. ✓

$$\begin{array}{lll} \text{maximiser} & \displaystyle\sum_{v \in V} f_{sv} & \\ \text{contraintes} & f_{uv} \leq c(u,v) & \forall u, v \in V \\ & \displaystyle\sum_{v \in V} f_{uv} = \sum_{v \in V} f_{vu} & \forall u \in V \setminus \{s,t\} \\ & f_{uv} \geq 0 & \forall u, v \in V \end{array}$$

2.

$$\begin{array}{lll} \text{minimiser} & \displaystyle\sum_{(u,v) \in E} a(u,v) \times f_{uv} & \\ \text{contraintes} & f_{uv} \leq c(u,v) & \forall u, v \in V \\ & \displaystyle\sum_{v \in V} f_{uv} - \sum_{v \in V} f_{vu} = 0 & \forall u \in V \setminus \{s,t\} \\ & \displaystyle\sum_{v \in V} f_{sv} - \sum_{v \in V} f_{vs} = d & \\ & f_{uv} \geq 0 & \forall u, v \in V \end{array}$$

3.

$$\begin{array}{lll} \text{maximiser} & d_t & \\ \text{contraintes} & d_v \leq d_u + w(u,v) & \forall (u,v) \in E \\ & d_s = 0 & \end{array}$$

4.

$$\begin{array}{lll} \text{minimiser} & \displaystyle\sum_{(u,v) \in E} c(u,v) \times d_{uv} & \\ \text{contraintes} & d_{uv} - z_u + z_v \geq 0 & \forall (u,v) \in E, u \neq s, v \neq t \\ & d_{sv} + z_v \geq 1 & \forall (s,v) \in E, v \neq t \\ & d_{ut} - z_u \geq 0 & \forall (u,t) \in E, u \neq s \\ & d_{st} \geq 1 & \text{si } (s,t) \in E \\ & d_{uv} \geq 0 & \forall (u,v) \in E \end{array}$$

✓(problème dual de coupe minimum)

Plan

Contexte

- Pour chaque programme linéaire, on peut en dériver un second avec la même valeur de l'objectif optimale mais qui inverse le sens de l'optimisation (minimisation/maximisation).
- Le *primal* est le programme linéaire de base.
- Le *dual* est son problème associé.
- Si l'objectif du primal est de maximiser, alors le dual minimise la fonction objectif.
- Si une valeur de l'objectif est atteinte la fois pour le primal et le dual, alors elle est optimale.
- C'est le principe de *dualité en programmation linéaire* qui sert à prouver l'optimalité d'une solution.

Mécanisme

Primal :

$$\begin{aligned} \text{maximiser} \quad & \sum_{j=1}^{n} c_j x_j \\ \text{sous les contraintes} \quad & \sum_{j=1}^{n} a_{ij} x_j \leq b_i \text{ pour } i = 1, 2, \ldots, m \\ & x_j \geq 0 \quad \text{pour } j = 1, 2, \ldots, n \end{aligned}$$

Dual :

$$\begin{aligned} \text{minimiser} \quad & \sum_{i=1}^{m} b_i y_i \\ \text{sous les contraintes} \quad & \sum_{i=1}^{m} a_{ij} y_i \geq c_j \text{ pour } j = 1, 2, \ldots, n \\ & y_i \geq 0 \quad \text{pour } i = 1, 2, \ldots, m \end{aligned}$$

Mécanisme

- Le primal a n variables et m contraintes, l'inverse pour le dual.
- On inverse le sens de l'optimisation et des inégalités des contraintes.
- Chacune des m contraintes du primal devient une variable du dual (et inversement) : le coefficient de la contrainte devient le coefficient de l'objectif.

Exemple numérique

$$
\begin{array}{lrcrcrcr}
\text{maximiser} & 3x_1 & + & x_2 & + & 4x_3 & & \\
\text{sous les contraintes} & x_1 & + & x_2 & + & 3x_3 & \leq & 30 \\
 & 2x_1 & + & 2x_2 & + & 5x_3 & \leq & 24 \\
 & 4x_1 & + & x_2 & + & 2x_3 & \leq & 36 \\
 & & & & & x_1, x_2, x_3 & \geq & 0
\end{array}
$$

$$
\begin{array}{lrcrcrcr}
\text{minimiser} & 30y_1 & + & 24y_2 & + & 36y_3 & & \\
\text{sous les contraintes} & y_1 & + & 2y_2 & + & 4y_3 & \geq & 3 \\
 & y_1 & + & 2y_2 & + & y_3 & \geq & 1 \\
 & 3y_1 & + & 5y_2 & + & 2y_3 & \geq & 4 \\
 & & & & & y_1, y_2, y_3 & \geq & 0
\end{array}
$$

Borne supérieure sur l'objectif (1/2)

- On va analyser les contraintes pour déterminer quelle valeur pourrait être atteinte par le primal au maximum.
- Aucune des contraintes n'est plus grande que la fonction objectif, mais en sommant les 2 premières contraintes, on obtient : $3x_1 + 3x_2 + 8x_3 \leq 54$.
- C'est un majorant de la fonction objectif : $3x_1 + x_2 + 4x_3 \leq 3x_1 + 3x_2 + 8x_3$.
- Les variables du dual sont en fait les coefficients multiplicatifs appliqués à chaque contrainte. On les minimise pour affiner le majorant.

Borne supérieure sur l'objectif (2/2)

- En multipliant chaque contrainte par une variable y, on obtient :
  $y_1(x_1 + x_2 + 3x_3) + y_2(2x_1 + 2x_2 + 5x_3) + y_3(4x_1 + x_2 + 2x_3) \leq 30y_1 + 24y_2 + 36y_3$.
- Dans l'exemple précédent, on avait $y_1 = y_2 = 1$ et $y_3 = 0$.
- On va chercher les conditions pour que cette inégalité constitue un majorant valide de la fonction objectif. Après ré-écriture :
  $(y_1 + 2y_2 + 4y_3)x_1 + (y_1 + 2y_2 + y_3)x_2 + (3y_1 + 5y_2 + 2y_3)x_3 \leq 30y_1 + 24y_2 + 36y_3$.
- Pour que ce soit un majorant, il faut que $y_1 + 2y_2 + 4y_3 \geq 3$, $y_1 + 2y_2 + y_3 \geq 1$ et $3y_1 + 5y_2 + 2y_3 \geq 4$. Auquel cas, $30y_1 + 24y_2 + 36y_3$ est bien un majorant de la valeur de l'objectif optimale.
- Comme on veut le majorant le plus fin possible, on minimise cette quantité.

Dualité faible en programmation linéaire

Lemme

Soit $\bar{x}$ une quelconque solution réalisable du programme linéaire primal et soit $\bar{y}$ une quelconque solution réalisable du programme linéaire dual. Alors

$$\sum_{j=1}^{n} c_j \bar{x}_j \leq \sum_{i=1}^{m} b_i \bar{y}_i$$

Dualité faible en programmation linéaire (preuve)

Démonstration.

$$\begin{aligned}
\sum_{j=1}^{n} c_j \bar{x}_j &\leq \sum_{j=1}^{n} \left(\sum_{i=1}^{m} a_{ij} \bar{y}_i \right) \bar{x}_j \\
&\leq \sum_{i=1}^{m} \left(\sum_{j=1}^{n} a_{ij} \bar{x}_j \right) \bar{y}_i \\
&\leq \sum_{i=1}^{m} b_i \bar{y}_i
\end{aligned}$$

$\square$

Corollaire

Soit $\bar{x}$ une solution réalisable d'un programme linéaire primal et soit $\bar{y}$ une solution réalisable du programme dual associé. Si

$$\sum_{j=1}^{n} c_j \bar{x}_j = \sum_{i=1}^{m} b_i \bar{y}_i$$

alors $\bar{x}$ et $\bar{y}$ sont des solutions optimales pour les programmes linéaires primal et dual respectivement.

Démonstration.

- D'après le lemme de la dualité faible en programmation linéaire, les valeurs des solutions du primal ne peuvent pas dépasser celles des solutions du dual.
- Le primal est un problème de maximisation et le dual de minimisation.
- Si des solutions réalisables ont la même valeur de l'objectif, aucune ne peut être améliorée.

$\square$

Dualité en programmation linéaire

Théorème

Si un primal et son dual admettent des solutions réalisables et finies, alors les solutions optimales x^* *et* y^* *sont telles que* $c^T x^* = b^T y^*$.

Question

Quel est le dual du problème introductif sur l'élection ?

1.

$$\begin{array}{lrcrcrcl}
\text{maximiser} & 100y_1 & + & 50y_2 & + & 25y_3 & & \\
\text{contraintes} & -4y_1 & + & 5y_2 & + & 6y_3 & \leq & 1 \\
 & 8y_1 & + & 2y_2 & - & 7y_3 & \leq & 1 \\
 & & & 10y_2 & & & \leq & 1 \\
 & 10y_1 & & & - & 2y_3 & \leq & 1 \\
 & & & & & y_1, y_2, y_3 & \geq & 0
\end{array}$$

2.

$$\begin{array}{lrcrcrcl}
\text{maximiser} & 50y_1 & + & 25y_2 & + & 100y_3 & & \\
\text{contraintes} & -3y_1 & + & 5y_2 & + & 4y_3 & \leq & 1 \\
 & 9y_1 & + & 2y_2 & - & 5y_3 & \leq & 1 \\
 & & & & & 10y_3 & \leq & 1 \\
 & 10y_1 & & & - & 2y_3 & \leq & 1 \\
 & & & & & y_1, y_2, y_3 & \geq & 0
\end{array}$$

3.

$$\begin{array}{lrcrcrcl}
\text{maximiser} & 100y_1 & + & 50y_2 & + & 25y_3 & & \\
\text{contraintes} & 5y_1 & - & 2y_2 & + & 3y_3 & \leq & 1 \\
 & 2y_1 & + & 8y_2 & - & 5y_3 & \leq & 1 \\
 & & & & & 10y_3 & \leq & 1 \\
 & & & 10y_2 & - & 2y_3 & \leq & 1 \\
 & & & & & y_1, y_2, y_3 & \geq & 0
\end{array}$$

4.

$$\begin{array}{lrcrcrcl}
\text{maximiser} & 25y_1 & + & 100y_2 & + & 50y_3 & & \\
\text{contraintes} & -2y_1 & + & 6y_2 & + & 4y_3 & \leq & 1 \\
 & 4y_1 & + & 2y_2 & + & 5y_3 & \leq & 1 \\
 & & & 10y_2 & & & \leq & 1 \\
 & 10y_1 & & & - & 2y_3 & \leq & 1 \\
 & & & & & y_1, y_2, y_3 & \geq & 0
\end{array}$$

Question

Quel est le dual du problème introductif sur l'élection ?

1. ✗

$$\begin{array}{lrcrcrcl}
\text{maximiser} & 100y_1 & + & 50y_2 & + & 25y_3 & & \\
\text{contraintes} & -4y_1 & + & 5y_2 & + & 6y_3 & \leq & 1 \\
 & 8y_1 & + & 2y_2 & - & 7y_3 & \leq & 1 \\
 & & & 10y_2 & & & \leq & 1 \\
 & 10y_1 & & & - & 2y_3 & \leq & 1 \\
 & & & & & y_1, y_2, y_3 & \geq & 0
\end{array}$$

2. ✗

$$\begin{array}{lrcrcrcl}
\text{maximiser} & 50y_1 & + & 25y_2 & + & 100y_3 & & \\
\text{contraintes} & -3y_1 & + & 5y_2 & + & 4y_3 & \leq & 1 \\
 & 9y_1 & + & 2y_2 & - & 5y_3 & \leq & 1 \\
 & & & & & 10y_3 & \leq & 1 \\
 & 10y_1 & & & - & 2y_3 & \leq & 1 \\
 & & & & & y_1, y_2, y_3 & \geq & 0
\end{array}$$

3. ✓(inversion des colonnes 1 et 2)

$$\begin{array}{lrcrcrcl}
\text{maximiser} & 100y_1 & + & 50y_2 & + & 25y_3 & & \\
\text{contraintes} & 5y_1 & - & 2y_2 & + & 3y_3 & \leq & 1 \\
 & 2y_1 & + & 8y_2 & - & 5y_3 & \leq & 1 \\
 & & & & & 10y_3 & \leq & 1 \\
 & & & 10y_2 & - & 2y_3 & \leq & 1 \\
 & & & & & y_1, y_2, y_3 & \geq & 0
\end{array}$$

4. ✗

$$\begin{array}{lrcrcrcl}
\text{maximiser} & 25y_1 & + & 100y_2 & + & 50y_3 & & \\
\text{contraintes} & -2y_1 & + & 6y_2 & + & 4y_3 & \leq & 1 \\
 & 4y_1 & + & 2y_2 & + & 5y_3 & \leq & 1 \\
 & & & 10y_2 & & & \leq & 1 \\
 & 10y_1 & & & - & 2y_3 & \leq & 1 \\
 & & & & & y_1, y_2, y_3 & \geq & 0
\end{array}$$

Plan

Résumé

- La formulation d'un problème comme programme linéaire consiste à le modéliser avec des contraintes et une fonction objectif qui sont des sommes pondérées des variables que l'on souhaite déterminer.
- La programmation linéaire permet de traiter de nombreux problèmes d'optimisation combinatoire.
- Des outils efficaces permettent de résoudre des programmes avec de nombreuses variables et contraintes.
- Un problème de programmation linéaire peut s'adapter facilement en modifiant sa formulation.
- À chaque problème de programmation linéaire correspond un dual dont chaque solution réalisable atteint un majorant de la valeur de l'objectif pour le primal et dont la valeur optimale est la même que pour le primal.

Prochaines échéances

- Début du tournoi le 19/11.
- Premier rendu intermédiaire du tournoi le 2/12.
- Second rendu intermédiaire du tournoi le 9/12.
- Épreuve de TP le 17/12.
- Rendu final pour le tournoi le 8/1.
- Épreuve sur table et restitution du tournoi le 9/1.