École Nationale Polytechnique de Constantine

Classe préparatoire deuxième année, Informatique 3

Théorie des graphes et programmation linéaire

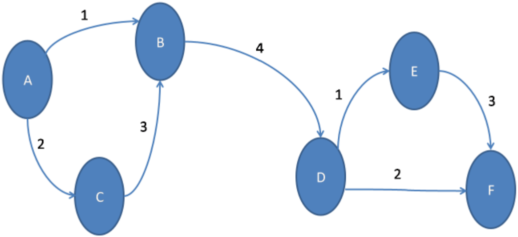
Dr. Mohamed Mahdi Benmoussa

Plan

Réseau de transport

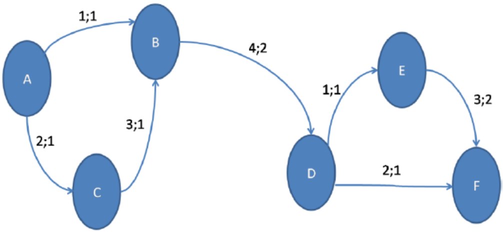
Un réseau de transport est un graphe valué avec les spécifications suivante :

- A chaque arc e est associé un nombre positif $c(e)$ appelé *capacité*.
- Il existe deux sommets particuliers un sommet *source* (un sommet sans prédécesseurs) et un sommet *puits* (un sommet sans successeurs).



Flot

Un flot F dans un réseau de transport, associe à chaque arc e une quantité $f(e)$ qui représente la quantité de flux qui passe par cet arc en provenance de la source vers le puits.



Flot conservatif

Un flot est conservatif s'il obéît à la loi de **KIRCHOFF** (la loi des nœuds) : La somme des quantités de flux sur les arcs entrant dans un sommet doit être égale à la somme des quantités de flux sur les arcs sortant de ce même sommet.

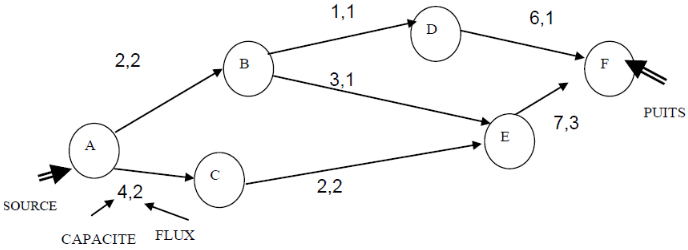
Flot compatible

Un flot est compatible dans un réseau si pour tout arc $e = (u, v), 0 \leq f(e) \leq c(e)$, c'est-à-dire pour chaque arc e, le flux qui le traverse ne dépasse pas sa capacité.

Flot complet

Un flot est complet si pour tout chemin allant de la source au puits, il y'a au moins un arc saturé (le flux qui le traverse est égal à sa capacité $f(e) = c(e)$).

Exemple



Recherche du flot maximum

Elle consiste à trouver la quantité maximum de flot à acheminer de la source s vers le puits p, en tenant compte des capacités de transport. L'algorithme de Ford-Fulkerson est le plus connu pour résoudre ce problème.

Principe de l'algorithme de Ford-Fulkerson

- Faire passer un flot compatible dans le réseau puis l'améliorer jusqu'à ce qu'on obtienne un flot complet.
- Pour améliorer le flot, il faut trouver une chaine de s à p dont les arcs dans le sens direct n'ont pas atteint leur limite ($f(e) < c(e)$) et les arcs dans le sens indirect ont un flux non nul ($f(u) > 0$). Cette chaine est dite améliorante ou augmentant.
- Le flot peut être amélioré de $\varepsilon = \min(\varepsilon^+, \varepsilon^-)$ avec $\varepsilon^+ = \min(c(e) - f(e))/e \in C^+$ et $\varepsilon^- = \min(f(e))/e \in C^-$, en rajoutant ε au flot des arcs C^+ et en retranchant ε au flot des arcs C^-.

Principe de l'algorithme de Ford-Fulkerson

- Le principe est de faire passer un flot compatible dans le réseau (le plus évident est le flot nul) puis de l'améliorer jusqu'à ce qu'on obtienne un flot complet.
- Pour améliorer le flot, il faut trouver une chaine de la source vers le puits dont les arcs dans le sens direct n'ont pas atteint leur limite ($f(e) < c(e)$) et les arcs dans le sens indirect ont un flux non nul ($f(u) > 0$).
- Cette chaine est dite **améliorante ou augmentant**. Le flot sur cette chaine peut être augmenté de la valeur ε.

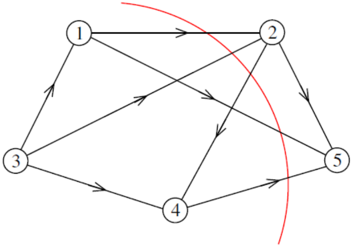
	Données: un graphe valué G= (V, E, c).
	Résultat: Un flot F complet.
0	Initialisation: s ϵ V, A = {s}; $C^+ = Ø$; $C^- = Ø$; $F^k = 0$; k = 0
1	Chercher u ϵ A
	Marquer le sommet v successeur de u telquef(u,v)<c(u,v) $C^+ := C^+ \cup \{(u,v)\}$; A := A ∪ {v}
	Marquer le sommet v prédécesseur de u telquef(v,u)>0$C^- := C^- \cup \{(v,u)\}$; A := A ∪ {v}
	Quand on ne peut plus marquer, deux cas se présentent: - p (sommet puits) est marqué, aller en (2) - p n'est pas marqué, terminé (le flot est maximum)
2	On a obtenu une chaine augmentante $C = C^+ \cup C^-$ de s à p.
	On calcul, $\varepsilon = \min(\varepsilon^+, \varepsilon^-)$ avec $\varepsilon^+ = m\,\mathrm{in}\big(c(e) - f(e)\big) / e \in C^+$ et $\varepsilon^- = m\,\mathrm{in}\big(f(e)\big) / e \in C^-$
	On définit le nouveau Flot: $F^{k+1}(e)=\{F^k(e)+\varepsilon$ pour $e\epsilon C^+$, $F^k(e)-\varepsilon$ pour $e\epsilon C^-$, $F^k(e)$ pour $e \notin C\}$.
	A:={s} et aller en (1).

Propriétés

- Différence entre un chemin et une chaine
- **flot complet** : saturer tous les chemins
- **flot maximale** : flot complet + saturer toutes les chaines
- Choix des chemins est aléatoire
- Un flot complet (avec sa valeur) n'est pas unique
- Un flot maximum n'est pas unique mais sa valeur est unique

Coupe

Étant donné un graphe $G = (V, E)$ et un sous-ensemble de S de sommets de V, on appelle **coupe** associée à S, et on la note $\delta(S)$, l'ensemble des arcs (i, j) tels que $i \in S$ et $j \in V$§.



$$\delta(\{1, 3, 4\}) = \{(1, 2), (1, 5), (3, 2), (4, 5)\}$$

Capacité d'une coupe

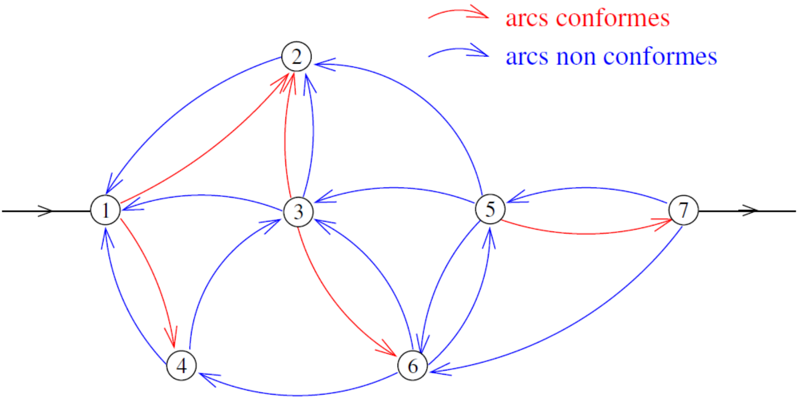
Si $c = (c(i,j),\ (i,j) \in E)$ est un système de capacités associés aux arcs du graphe $G = (V, E)$ et si $\delta(S)$ est une coupe du graphe alors la capacité de $\delta(S)$ est définie par :
$C(\delta(S)) = \Sigma_{(i,j)\in\delta(S)} c(i,j)$

Théorème

Soit un réseau $R = (G = (V, E), s, t, c)$. Si $f = (f(i,j), (i,j) \in E)$ est un flot réalisable entre s et t de valeur F et si $\delta(S)$ est une coupe qui sépare s et t alors $F \leq c(\delta(S))$.

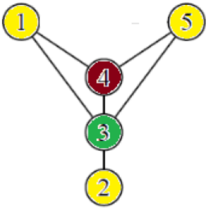
Théorème : flot max - coupe min

La valeur maximum d'un flot réalisable entre s et t est égale à la capacité minimum d'une coupe séparant s et t.


arcs conformes
arcs non conformes
1
2
3
4
5
6
7

Coloration

- Soit $G = (V, E)$ un graphe orienté ou non orienté. On dit que G est **k-chromatique** ou bien G admet une **k-coloration** s'il existe une partition de V en k ensembles stables. Ceci revient à dire qu'il est possible de colorier les sommets de G avec k couleurs différentes.
- **Le nombre chromatique** est le nombre le plus petit nécessaire pour colorier un graphe et il est compris entre 1 et le nombre de points.



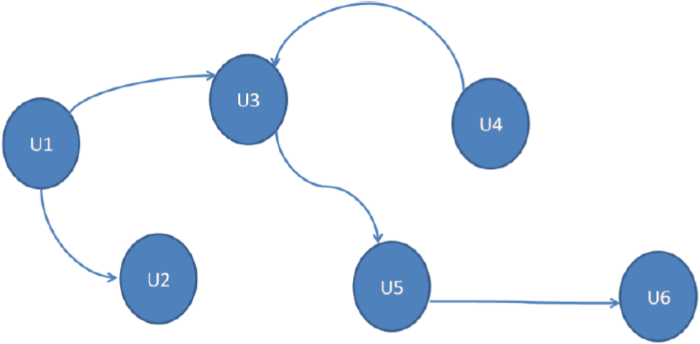
Algorithme de k-coloration d'un graphe

Établir une liste ordonnée de sommets (suivant l'ordre décroissant de leur degré). Tant qu'il reste des sommets à colorier, exécuter (Welsh et Powell, glouton) :

1	Choisir une nouvelle couleur (couleur d'usage)
2	Chercher dans la liste du sommet le premier sommet non coloré et le colorer par la couleur d'usage
3	Examiner tour à tour, dans l'ordre de la liste, tous les sommets non coloriés et colorier chaque sommet non adjacent à un sommet déjà coloré avec la couleur d'usage.

Définition d'un arbre

Un arbre est un graphe simple, connexe et sans cycles, le nombres d'arêtes est alors $\mid E \mid = n - 1 (n \geq 2)$

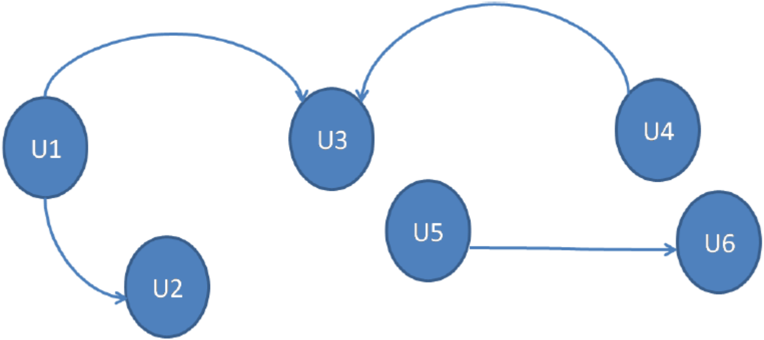


Caractéristiques

- G est connexe et sans cycles
- G et sans cycles et possède $(n-1)$ arcs
- G est sans cycles et est maximal pour cette propriété
- G est connexe et possède $(n-1)$ arcs
- G est connexe et minimal pour cette propriété
- Il existe dans G une chaîne et une seule joignant tout couple de sommet

Forêt

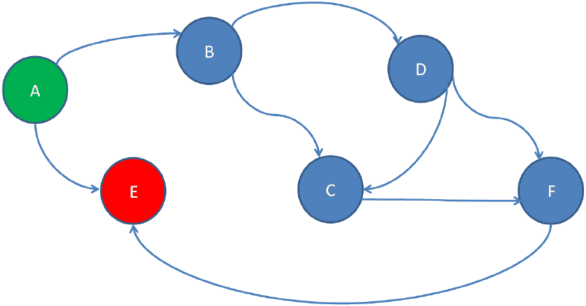
- On appelle forêt un graphe dont lequel chaque composante connexe est un arbre.
- Tout graphe partiel d'un arbre est une forêt
- Une forêt ayant p composantes connexes possède $(n - p)$ arcs



Arborescence

Racine

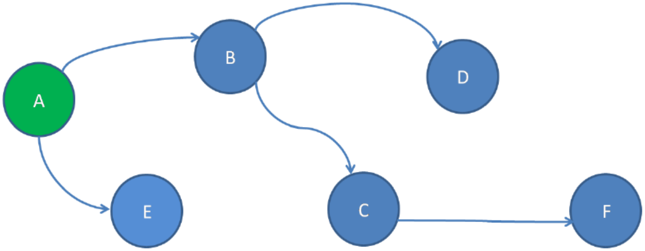
Un sommet S d'un graphe $G = (V, E)$ est une racine (respectivement antiracine) s'il existe un chemin enjoignant S à chaque sommet du graphe G (respectivement joignant chaque somme de G à S). Autrement dit $\forall u \in V \backslash \{S\}$ il existe un chemin de S à u (respectivement de u à S).



Arborescence

Un graphe $G = (V, E)$, avec $\mid V \mid= n(n \geq 2$ est une arborescence de racine S si :

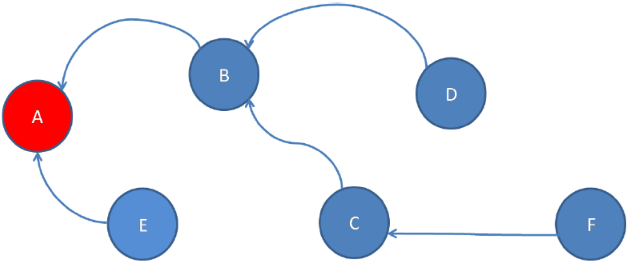
- G est un arbre
- S est une racine de G



Anti-arborescence

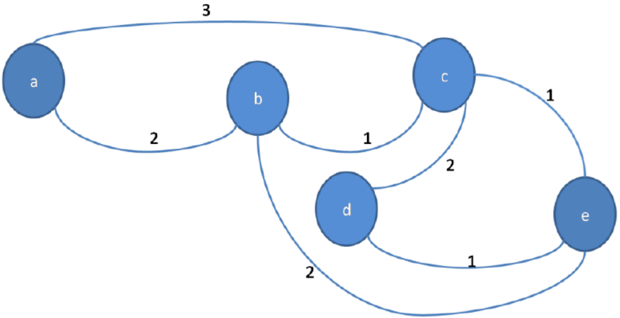
Un graphe $G = (V, E)$, avec $\mid V \mid = n(n \geq 2$ est une anti-arborescence admettant le sommet S comme anti-racine si :

- G est un arbre
- S est une anti-racine de G



Kruskal (devoir)

Cherchez et expliquer l'algorithme de Kruskal qui a pour but de chercher un arbre de pois minimal. Appliquez cet algorithme sur l'exemple suivant :



Plan

Recherche Opérationnelle (RO)

D'une manière générale

Ensemble de méthodes (algorithmiques, mathématiques, modélisation) afin de prendre des décisions optimales ou proches de l'optimum dans des problèmes complexes qui traitent de la maximisation d'un profit ou la minimisation d'un coût.

- Comment aller le plus vite d'Orléans à Berlin, en voiture ?
- Comment ordonnancer les tâches d'un projet en fonction de la main d'œuvre, tout en minimisant sa durée ?
- Comment investir ses 1000 euros d'économie de sorte à maximiser le profit obtenu après deux ans ?

Exemple de problèmes de RO

Example

- Trouver un (plus court) chemin entre deux villes
  - plus courts chemins dans les graphes
- Broadcast le coût minimum dans un réseau
  - arbres recouvrants de poids minimum
- Envoi d'un maximum d'information dans un réseau
  - problème du flot maximum

Modélisation

En recherche Opérationnelle (RO), modéliser un problème consiste à identifier :

- les **variables** intrinsèques (inconnues)
- les différentes **contraintes** auxquelles sont soumises ces variables
- l'**objectif** visé (optimisation).

Dans un problème de programmation linéaire (PL) les contraintes et l'objectif sont des fonctions **linéaires** des variables. On parle aussi de programme linéaire.

Programmation linéaire

Définition

Les problèmes de programmation linéaire (PL) sont des problèmes d'optimisation où la fonction objectif et les contraintes sont toutes linéaires. Objectif : apprendre à modéliser les problèmes réels et à résoudre les programmes linéaires.

- De nombreux problèmes réels peuvent être exprimés comme des programmes linéaires.
- Les programmes linéaires peuvent être résolus efficacement par certains algorithmes.
- Ce sont d'excellents exemples de questions pratiques dont la résolution nécessite une combinaison de méthodes algorithmiques, de mathématiques élémentaires et de bon sens.

Example

Considérons un agriculteur qui possède des terres, de superficie égale à H hectares, dans lesquelles il peut planter du blé et du maïs. l'agriculteur possède une quantité E d'engrais et I d'insecticide. Le blé nécessite une quantité E_b d'engrais par hectare et I_b d'insecticide par hectare. Les quantités correspondantes pour le maïs sont notées E_m et I_m.
Soit P_b le prix de vente du blé et P_m celui du maïs. Si l'on note par x_b et x_m le nombre d'hectares à planter en blé et en maïs, comment exprimer le fait que l'agriculteur souhaite maximiser son gain, tout en restant dans les limites de ses ressources.

Formulation

Maximiser la fonction objectif

- $P_b x_b + P_m x_m$ (maximiser le revenu net)
- $x_b + x_m \leq H$ (la surface totale)
- $x_b \geq 0$
- $x_m \geq 0$
- $E_b x_b + E_m x_m \leq E$ (quantité d'engrais)
- $I_b x_b + I_m x_m \leq I$ (quantité d'insecticide)

Définition (PL)

- Variables réelles :
  - $x_1, x_2, ..., x_n$
- Fonction objectif linéaire à optimiser (min ou max) :
  - $c_1x_1 + c_2x_2 + ... + c_nx_n$
- Contraintes linéaires (égalités ou inégalités) :
  - $a_{11}x_1 + a_{12}x_2 + ... + a_{1n}x_n \leq b_1$
  - $a_{21}x_1 + a_{22}x_2 + ... + a_{2n}x_n \geq b_2$
  - ...
  - $a_{m1}x_1 + a_{m2}x_2 + ... + a_{mn}x_n = b_m$

Solution : toute affectation des variables qui respecte les contraintes.
Solution optimale : solution qui optimise (maximise ou minimise) la fonction objectif.

Formes standard et forme canonique d'un PL

Forme standard

Un programme linéaire est sous **forme standard** lorsque toutes ses contraintes sont des égalités et toutes ses variables sont non-négatives.

- fonction objectif : $\max c^T x$
- sous contraintes : $Ax = b$
- $x \geq 0$
- n variables, m contraintes, $m < n$, $c, x \in \mathbb{R}^n, b \in \mathbb{R}^m, A \in \mathbb{R}^{m \times n}$.

Formes standard et forme canonique d'un PL

Forme canonique

Un programme linéaire est dit sous **forme canonique** lorsque toutes ses contraintes sont des **inégalités** et toutes ses variables sont non-négatives.

- fonction objectif : $\max c^T x$
- sous contraintes : $Ax \leq b$
- $x \geq 0$
- n variables, m contraintes, $m < n$, $c, x \in R^n, b \in R^m, A \in R^{m \times n}$.

Formes standard et forme canonique d'un PL

Theorem (Équivalence des formes standard et canonique)

Tout programme linéaire peut s'écrire sous forme standard et sous forme canonique.

Démonstration

- Une contrainte d'inégalité $a^T x \leq b$ peut être transformée en égalité par l'introduction d'une **variable d'écart** :

$$a^T x + s = b,$$
$$s \geq 0$$

- Une contrainte d'égalité $a^T x = b$ peut être remplacée par deux inégalités :

$$a^T x \leq b$$
$$-a^T x \leq -b$$

Formes standard et forme canonique d'un PL

Démonstration (suite)

- $a^T x \geq b \Leftrightarrow -a^T x \leq -b$
- $\min c^T x = -\max - c^T x$
- Variable x non restreinte : substitution par deux variables (partie positive et partie négative)

$$x = x^+ - x^-$$
$$x^+, x^- \geq 0.$$

Il existe toujours une solution optimale telle que $x^+ = 0$ ou $x^- = 0$.

Example

Production de peinture Une société produit de la peinture d'intérieur et d'extérieur à partir de deux produits de base $M1$ et $M2$.

- La demande maximum en penture d'intérieur est de 2 tonnes par jour.
- La production en peinture d'intérieur ne dépasser que d'une tonne celle d'extérieur.

	Quantité utilisée par tonne		Quantité disponible par jour
	Extérieure	Intérieure	
M1	6	4	24
M2	1	2	6
Profit par tonne	5	4	

Formulation

Alternatives ou variables inconnues du problème :

- x_1 = tonnes de peinture d'extérieur produites par jour
- x_2 = tonnes de peinture d'intérieur produites par jour

Fonction objectif à optimiser :

- $\max z = 5x_1 + 4x_2$

Restrictions ou contraintes :

- $6x_1 + 4x_2 \leq 24$
- $x_1 + 2x_2 \leq 6$
- $x_2 \leq 2$
- $x_2 - x_1 \leq 1$
- $x_1, x_2 \geq 0$

Formulation (forme standard)

Fonction objectif à optimiser :

- $\max z = 5x_1 + 4x_2$

Restrictions ou contraintes :

- $6x_1 + 4x_2 + s_1 = 24$
- $x_1 + 2x_2 + s_2 = 6$
- $x_2 + s_3 = 2$
- $x_2 - x_1 + s_4 = 1$
- $x_1, x_2, s_1, s_2, s_3, s_4 \geq 0$

Variables avec des valeurs négatives

Example

- Un fast-food vend des hamburgers et des cheeseburgers. Un hamburger utilise 125g de viande alors qu'un cheeseburger n'en utilise que 100g.
- Le fast-food démarre chaque journée avec 10kg de viande mais peut commander de la viande supplémentaire avec un coût additionnel de 2*euro* par kg pour la livraison.
- Le profit est de 0.02*euro* pour un hamburger et 0.0015*euro* pour un cheeseburger.
- La demande ne dépasse pas 900 sandwiches par jour, et les surplus de viande sont donnés au Restos du cœur.

Combien le fast-food doit-il produire de sandwiches de chaque type par jour ?

Variables

- x_1 = nombre de hamburgers par jour.
- x_2 = nombre de cheeseburgers par jour.

Contraintes

- Commande de viande supplémentaire :

$$125x_1 + 100x_2 + x_3 = 10000$$

- Le coût pour la viande supplémentaire apparaît seulement si $x_3 < 0$.
- Substitution de x_3 par deux variables non-négatives :

$$x_3 = x_3^+ - x_3^-, x_3^+, x_3^- \geq 0.$$
$$125x_1 + 100x_2 + x_3^+ - x_3^- = 10000.$$

- Borne supérieure sur les ventes : $x_1 + x_2 \leq 900$.

Modèle complet

$$\max z = 0.02x_1 + 0.015x_2 - 0.002x_3^-$$
$$\text{s.c}: 125x_1 + 100x_2 + x_3^+ - x_3^- = 10000$$
$$x_1 + x_2 \leq 900$$
$$x_1, x_2, x_3^+, x_3^- \geq 0$$

Remarque : Il existe une solution optimale telle que $x_3^+ = 0$ ou $x_3^- = 0$.

Résolution des problèmes de la PL

Méthodes

Il existe plusieurs solutions pour la résolution des PL :

- Résolution graphique
- La méthode du simplexe
- La méthode des deux phases
- etc...

Problème à deux variables

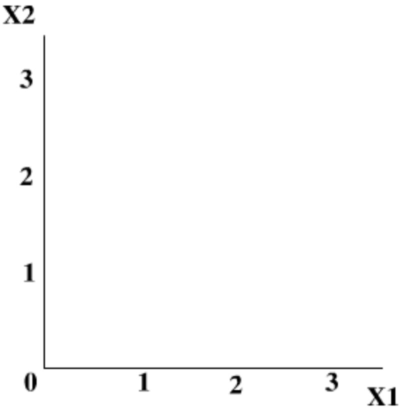
Définition

Les contraintes où apparaissent des inégalités correspondent géométriquement à des demi-plans.
Intersection de ces demi-plans = ensemble des variables satisfaisant à toutes les contraintes.
L'ensemble des contraintes (des solutions acceptables) est un **polygone convexe**.
Solution optimale (si elle existe) : sommet du polygone.

Représentation graphique

L'ensemble des solutions réalisables est l'ensemble des points de l'espace qui vérifient les contraintes:

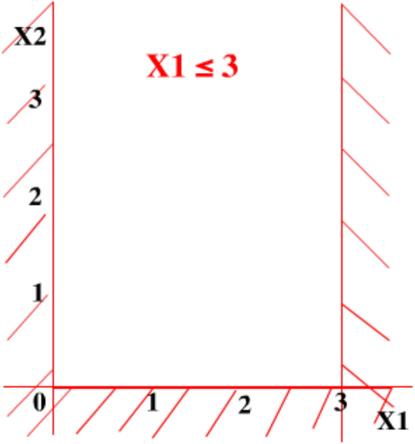
maximiser:
$Z_1 = 2.X_1 + X_2$
sous les contraintes:
$X_1 + 2.X_2 \leq 6$
$X_1 + X_2 \leq 4$
$X_1 \leq 3$
$X_i \geq 0$



Représentation graphique

L'ensemble des solutions réalisables est l'ensemble des points de l'espace qui vérifient les contraintes:

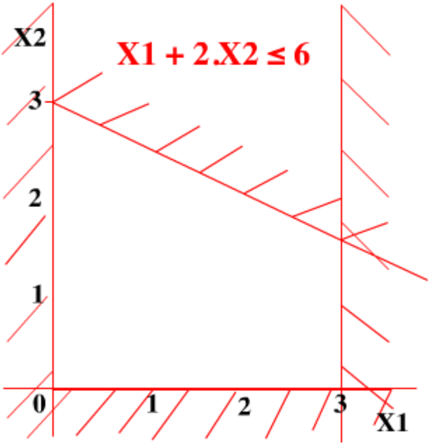
maximiser:
$Z_1 = 2.X_1 + X_2$
sous les contraintes:
$X_1 + 2.X_2 \leq 6$
$X_1 + X_2 \leq 4$
$X_1 \leq 3$
$X_i \geq 0$



Représentation graphique

L'ensemble des solutions réalisables est l'ensemble des points de l'espace qui vérifient les contraintes:

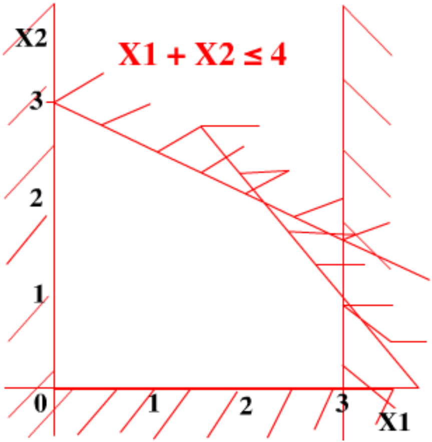
maximiser:
$Z_1 = 2.X_1 + X_2$
sous les contraintes:
$X_1 + 2.X_2 \leq 6$
$X_1 + X_2 \leq 4$
$X_1 \leq 3$
$X_i \geq 0$



Représentation graphique

L'ensemble des solutions réalisables est l'ensemble des points de l'espace qui vérifient les contraintes:

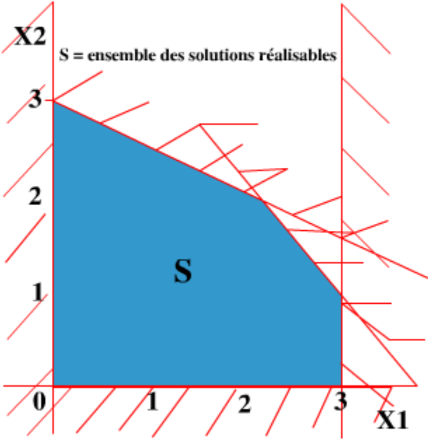
maximiser:
$Z_1 = 2.X_1 + X_2$
sous les contraintes:
$X_1 + 2.X_2 \leq 6$
$X_1 + X_2 \leq 4$
$X_1 \leq 3$
$X_i \geq 0$



Représentation graphique

L'ensemble des solutions réalisables est l'ensemble des points de l'espace qui vérifient les contraintes:

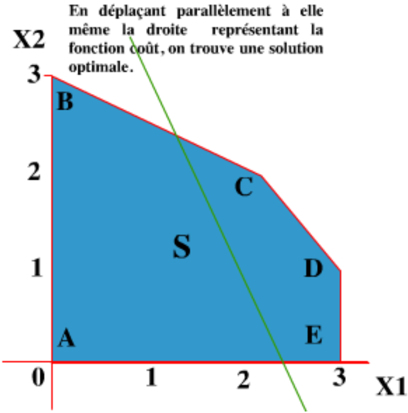
maximiser:
$Z_1 = 2.X_1 + X_2$
sous les contraintes:
$X_1 + 2.X_2 \leq 6$
$X_1 + X_2 \leq 4$
$X_1 \leq 3$
$X_i \geq 0$



Représentation graphique

L'ensemble des solutions réalisables est l'ensemble des points de l'espace qui vérifient les contraintes:

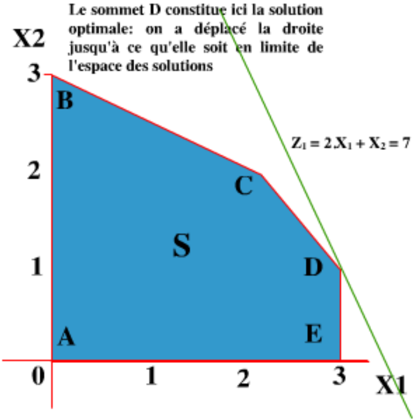
maximiser:
$Z_1 = 2X_1 + X_2$
sous les contraintes:
$X_1 + 2X_2 \leq 6$
$X_1 + X_2 \leq 4$
$X_1 \leq 3$
$X_i \geq 0$



Représentation graphique

L'ensemble des solutions réalisables est l'ensemble des points de l'espace qui vérifient les contraintes:

maximiser:
$Z_1 = 2.X_1 + X_2$
sous les contraintes:
$X_1 + 2.X_2 \leq 6$
$X_1 + X_2 \leq 4$
$X_1 \leq 3$
$X_i \geq 0$



Représentation graphique

Dans cet exemple la solution optimale correspond à

$$X_1 = 3 \text{ et } X_2 = 1$$

La valeur correspondante de la fonction Z est donc de :

$$Z = 2 * X_1 + X_2 = 7$$

Conclusion

Cette méthode graphique est bien sûr facile à mettre en œuvre lorsqu'il y a deux variables, elle devient plus difficile pour trois variables et impossible au delà.
Une méthode du simplexe a été développée par Dantzig afin de résoudre ces types de problèmes de programmation linéaire.

Méthode du simplexe

- Méthode développée par G.Dantzig
- Algorithme de résolution de programmes linéaires
- Facile à comprendre et à implémenter
- Efficace en pratique, même pour un nombre important de variables et de contraintes

Principe

- Solution optimale : sommet (point extrême).
- Idée fondamentale du simplexe : déplacement de sommet en sommet adjacent de manière à améliorer la fonction objectif.
- Transformation des inégalités en égalités : forme standard du programme linéaire - système de m équations à n inconnues ($m < n$).
- Identification algébrique des sommets : correspondance avec les bases d'un système d'équations.

Exemple 1

$$
\begin{array}{lrcrcrcr}
\max & 5x_1 & + & 4x_2 & + & 3x_3 & & \\
\text{contraintes} & 2x_1 & + & 3x_2 & + & x_3 & \leq & 5 \\
 & 4x_1 & + & x_2 & + & 2x_3 & \leq & 11 \\
 & 3x_1 & + & 4x_2 & + & 2x_3 & \leq & 8 \\
 & x_1, & & x_2, & & x_3 & \geq & 0
\end{array}
$$

Exemple 1

PL équivalent :

$$
\begin{array}{rcrcrcrcr}
x_4 & = & 5 & - & 2x_1 & - & 3x_2 & - & x_3 \\
x_5 & = & 11 & - & 4x_1 & - & x_2 & - & 2x_3 \\
x_6 & = & 8 & - & 3x_1 & - & 4x_2 & - & 2x_3 \\
\hline
z & = & & & 5x_1 & + & 4x_2 & + & 3x_3
\end{array}
$$

$$\max z, \text{sous contraintes : } x_1, x_2, x_3, x_4, x_5, x_6 \geq 0$$

Solution initiale : $x_1 = x_2 = x_3 = 0$, $x_4 = 5$, $x_5 = 11$, $x_6 = 8$; $z = 0$.

Si on augmente x_1, ça augmente z!

Exemple 1

1. De combien ?

   $x_1 \leq \frac{5}{2}$ car $x_4 = 5 - 2x_1 - 3x_2 - x_3 \geq 0.$

2. Nouvelle solution ?

   5/2 $x_1 = \frac{1}{2}, x_2 = x_3 = x_4 = 0, x_5 = 1, x_6 = \frac{1}{2}\ ;\ z = \frac{25}{2}.$

3. Reformuler le système afin d'exprimer les variables x_1, x_5, x_6 en fonction de x_2, x_3, x_4.

   $x_1 = \frac{5}{2} - \frac{3}{2}x_2 - \frac{1}{2}x_3 - \frac{1}{2}x_4$

Exemple 1

$$
\begin{array}{rcrcrcrcr}
x_1 & = & \frac{5}{2} & - & \frac{3}{2}x_2 & - & \frac{1}{2}x_3 & - & \frac{1}{2}x_4 \\
x_5 & = & 1 & + & 5x_2 & & & + & 2x_4 \\
x_6 & = & \frac{1}{2} & + & \frac{1}{2}x_2 & - & \frac{1}{2}x_3 & + & \frac{3}{2}x_4 \\
\hline
z & = & \frac{25}{2} & - & \frac{7}{2}x_2 & + & \frac{1}{2}x_3 & - & \frac{5}{4}x_4
\end{array}
$$

Augmenter qui?	De combien?	Qu'obtient-on?
x_3	$x_3 = 1$ ($x_6 = \dots$)	à droite : x_2, x_3, x_6

Exemple 1

$$
\begin{array}{rcrcrcrcr}
x_3 & = & 1 & + & x_2 & + & 3x_4 & - & 2x_6 \\
x_1 & = & 2 & - & 2x_2 & - & 2x_4 & + & x_6 \\
x_5 & = & 1 & + & 5x_2 & + & 2x_4 & & \\
\hline
z & = & 13 & - & 3x_2 & - & x_4 & - & x_6
\end{array}
$$

Solution actuelle : $x_2 = x_4 = x_6 = 0, x_1 = 2, x_3 = 1, x_5 = 1$;
objectif : $z = 13$

Peut-on faire mieux ? Non car $x_2, x_4, x_6 \geq 0$ et donc $z \leq 13$

Exemple 1

Entrée :

$$\max \quad \sum_{j=1}^{n} c_j x_j$$

$$\text{contraintes :} \quad \sum_{j=1}^{n} a_{ij} x_j \leq b_i \quad i = 1, 2, \ldots, m$$

$$x_j \geq 0 \quad j = 1, 2, \ldots, n$$

- On introduit m variables d'écart :

$$x_{n+i} = b_i - \sum_{}^{n} a_{ij} x_j \quad (i = 1 \ldots, m)$$

- La $i$ème contrainte devient $x_{n+i} \geq 0$.

Exemple 1

$$x_{n+i} = b_i - \sum_{j=1}^{n} a_{ij} x_j \qquad (i = 1, \ldots, m)$$

$$z = \sum_{j=1}^{n} c_j x_j$$

$$\max z, \text{sous contraintes} : \; x_k \geq 0 \qquad (k = 1, \ldots n + m)$$

Dictionnaires : équivalents au PL d'origine

- $n + m + 1$ variables $x_1, \ldots, x_{n+m}$ et z
- m équations linéaires de types $x_k = \ldots$
- tous les membres droits ont les mêmes n variables
- objectif : max z sous contraintes ci-dessus plus $x_k \geq 0 (k = 1, \ldots n + m)$

Exemple 1

$$
\begin{array}{rcrcrcrcr}
x_4 & = & 5 & - & 2x_1 & - & 3x_2 & - & x_3 \\
x_5 & = & 11 & - & 4x_1 & - & x_2 & - & 2x_3 \\
x_6 & = & 8 & - & 3x_1 & - & 4x_2 & - & 2x_3 \\
\hline
z & = & & & 5x_1 & + & 4x_2 & + & 3x_3
\end{array}
$$

- Dictionnaire faisable : en mettant à 0 toutes les variables des membres droits, on obtient une solution du PL
  - On suppose que notre premier dictionnaire est faisable
  - Nous verrons plus tard comment faire pour que ce soit toujours le cas
- Variables "de gauche" (sauf z) : variables de base (x_4, x_5, x_6)
- Variables "de droite" : variables hors base (x_1, x_2, x_3)

Exemple 1

$$
\begin{array}{rcrcrcrcr}
x_4 & = & 5 & - & 2x_1 & - & 3x_2 & - & x_3 \\
x_5 & = & 11 & - & 4x_1 & - & x_2 & - & 2x_3 \\
x_6 & = & 8 & - & 3x_1 & - & 4x_2 & - & 2x_3 \\
\hline
z & = & & & 5x_1 & + & 4x_2 & + & 3x_3
\end{array}
$$

1. Solution courante : variables hors base à 0
2. Trouver une variables hors base x_l dont le coefficient dans l'expression de z soit strictement positif
3. S'il n'en existe pas, la solution courante est optimale !
4. Trouver la contrainte la plus forte pour l'augmentation de x_k ; soit $x_i = \ldots$ cette contrainte
5. Augmenter x_k jusqu'à ce que l'on ai $x_i = 0$
6. Exprimer x_k en fonction de x_i et des autres variables hors base
7. Mettre à jour le dictionnaire, en faisant entrer x_k dans la

Dualité

Idée : associer au programme linéaire initial, appelé primal , un deuxième programme linéaire appelé le programme dual tel que :

- Le dual a m variables (autant que de contraintes dans le primal)
- Le dual a n contraintes (autant que de variables dans le primal)
- Le dual est un problème de minimisation
- Pour toute solution du dual et toute solution du primal, l'objectif du dual est supérieur ou égal à celui du primal

Problème primal

$$\begin{array}{rl} \max & c^T x \\ \text{s.c.} & Ax = b \\ & x \geq 0 \end{array}$$

n variables, m contraintes, $m < n$,
$c, x \in \mathbb{R}^n, \quad b \in \mathbb{R}^m, \quad A \in \mathbb{R}^{m \times n}$.

Problème dual

$$\begin{array}{rl} \min & b^T y \\ \text{s.c.} & A^T y \geq c \\ & (y \text{ non restreint}) \end{array}$$

m variables, n contraintes, $m < n$,
$c \in \mathbb{R}^n, \quad b, y \in \mathbb{R}^m, \quad A \in \mathbb{R}^{m \times n}$.

Example

Problème primal :

$$\begin{aligned} \max z &= x_1 + x_2 \\ 2x_1 + x_2 &= 5 \ (y_1) \\ 3x_1 - x_2 &= 6 \ (y_2) \\ x_1,\ x_2 &\geq 0 \end{aligned}$$

Problème dual :

$$\begin{aligned} \min W &= 5y_1 + 6y_2 \\ 2y_1 + 3y_2 &\geq 1 \ x_1 \\ y_1 - y_2 &\geq 1 \ x_2 \end{aligned}$$

Propriétés

Théorème

Le problème dual du problème dual est le problème primal.

Règles de construction

Problème max	Problème min
Contrainte	Variable
$\leq$	≥ 0
=	non restreinte
Variable	Contrainte
≥ 0	$\geq$
non restreinte	=

Example

Problème primal :

$$\max z = 5x_1 + 12x_2 + 4x_3$$
$$x_1 + 2x_2 + x_3 \leq 10 \ (y_1)$$
$$2x_1 - x_2 + 3x_3 = 8 \ (y_2)$$
$$x_1,\ x_2,\ x_3 \geq 0$$

Problème dual :

$$\min w = 10y_1 + 8y_2$$
$$y_1 + 2y_2 \geq 5 \ (x_1)$$
$$2y_1 - y_2 \geq 12 \ (x_2)$$
$$y_1 + 3y_2 \geq 4 \ (x_3)$$
$$y_1 \geq 0$$

Relation primal/dual)

Théorème (dualité faible)

Considérons la paire primale-duale :

$$\begin{array}{ll} \max c^T x & \min b^T y \\ Ax = b & A^T y \geq c \\ x \geq 0 & \end{array}$$

- Si x est une solution admissible du pirmal et y une solution admissible du dual, alors
$$c^T x \leq b^T y$$
- S'il y a égalité, alors x est une solution optimale du primal et y une solution optimale du dual.

Relation primal/dual

Théorème (dualité forte)

Considérons la paire primale-duale :

$$\begin{array}{ll} \max c^T x & \min b^T y \\ Ax = b & A^T y \geq c \\ x \geq 0 & \end{array}$$

- Si le primal et le dual admettent tous les deux deux une solution admissible, ils ont tous deux une solution optimale finie et la même valeur objectif optimale.
- Si le primal (dual) est non borné, le dual (primal) n'admet pas de solution admissible.

Relation primal/dual

Théorème (Complémentarité)

Considérons la paire primale-duale :

$$\begin{array}{ll} \max\, c^T x & \min\, b^T y \\ Ax = b & A^T y \geq c \\ x \geq 0 & \end{array}$$

Si x est une solution optimale du primal et y une solution optimale du dual alors

$$x_i(a_i^T y - c_i) = 0$$

En d'autres termes :

$$\begin{array}{l} x_i > 0 \;\Rightarrow\; a_i^T y = c_i \\ a_i^T y > c_i \;\Rightarrow\; x_i = 0 \end{array}$$

Relation primal/dual

Primal (P) :

$$\begin{array}{llllll} \max & z = 5x_1 & +12x_2 & +4x_3 & & \\ \text{s.c.} & x_1 & +2x_2 & +x_3 & \leq 10 & (y_1) \\ & 2x_1 & -x_2 & +3x_3 & = 8 & (y_2) \\ & x_1, & x_2, & x_3 & \geq 0 & \end{array}$$

Dual (D) :

$$\begin{array}{lllll} \min & w = 10y_1 & +8y_2 & & \\ \text{s.c} & y_1 & +2y_2 & \geq 5 & (x_1) \\ & 2y_1 & -y_2 & \geq 12 & (x_2) \\ & y_1 & +3y_2 & \geq 4 & (x_3) \\ & y_1 & & \geq 0 & \end{array}$$

Solution optimale de (P) :

$$(x_1, x_2, x_3) = \left(\frac{26}{5}, \frac{12}{5}, 0\right)$$

$$z = \frac{274}{5}$$

$$x_1 > 0 \quad \Rightarrow \quad y_1 + 2y_2 = 5$$
$$x_2 > 0 \quad \Rightarrow \quad 2y_1 - y_2 = 12$$

Solution optimale de (D) :

$$(y_1, y_2) = \left(\frac{29}{5}, -\frac{2}{5}\right)$$

$$w = \frac{274}{5}$$

Références

- Flots dans les réseaux (Algorithmique de graphes), Sylvie Borne
- Théorie des graphes, Bouzenada
- Recherche opérationnelle et applications, Bernard Fortz
- Programmation linéaire et recherche opérationnelle, Loan Todinca
- Graphes et RO, J.-F. Scheid