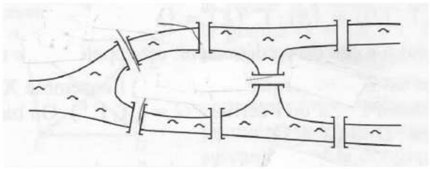
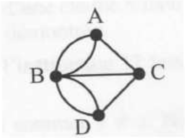
Les graphes pour la recherche opérationnelle

Introduction

Les ponts de Königsberg

- Déterminer un itinéraire à travers la ville de Königsberg passant une fois et une seule par chaque pont (7 ponts).
- C'est l'un des plus anciens problèmes combinatoire.





Introduction

Les ponts de Königsberg

En 1735, Euler montra l'impossibilité de ce problème, c'est le **premier témoignage de l'emploi des graphes** :

- Il faut déterminer une chaîne dans le graphe associé au problème, passant par toutes les arêtes.
- Pour toute chaîne, il existe au plus deux sommets de degré impair (nœud source, nœud destination).
- Tous les sommets du graphe considéré sont de degré impair.
- Il est donc impossible de résoudre le problème des ponts de Königsberg.

Introduction

Historique

- **1891** : "Graphes irréguliers", J. Petersen.
- **1936** : "Theorie der endlichen und unendlichen Graphen", Dénes König.
- **1958** : "Théorie des graphes et applications", Claude Berge.
- **1971** : "Graphes et hypergraphes", Claude Berge.

Objectif du cours

$\Rightarrow$ Application des graphes à la recherche opérationnelle.

Plan

Plan

Introduction

Historique

- **1601-1665** : Implicitement contenue dans l'œuvre de Pierre de Fermat.
- **1944** : Utilisée en recherche opérationnelle par P. Massé, France.
- **1952** : Utilisée en recherche opérationnelle par R. Bellman, USA.

Introduction

Grand principe

Toute partie d'un chemin optimal est optimale.

Prérequis

- Le critère d'optimisation doit être **décomposable**.
- Des **contraintes sur la valuation** du graphe existent en fonction de ce critère :
  - Critère multiplicatif : valuations positives du graphe.
  - . . .

Equation de récurrence

Définitions

- Etape : k.
- Nombre d'étapes : N.
- X^k l'ensemble des sommets x_k appartenant à l'étape k.
- $v_{k+1}(x_k, x_{k+1})$ la valeur de l'arc (x_k, x_{k+1}).

Equation de récurrence

Récurrence (optimisation séquentielle)

$$f^*_{k+1}(x_{k+1}) = OPT_{x_k \in X_k}[f^*_k(x_k) + v_{k+1}(x_k, x_{k+1})]$$

Algorithme

- Poser $k = 0$.
- $\forall x_{k+1} \in X_{k+1}$, déterminer l'ensemble $S_{k+1}(x_{k+1})$ où $s^{x_k}_{k+1}(x_{k+1}) = f^*_k(x_k) + v_{k+1}(x_k, x_{k+1})$, x_k étant un prédécesseur de x_{k+1}.
- Déterminer la valeur optimale de l'ensemble $s = S_{k+1}$.
- $f^*_{k+1}(x_{k+1}) = s$.
- Itérer jusqu'à $k = N - 1$.

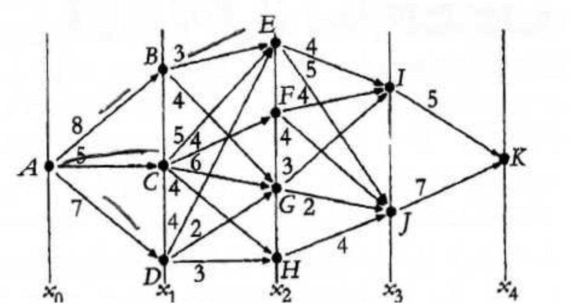
Exemple

Politique de construction de tronçons d'autoroute

- L'objectif est de construire une autoroute entre les villes A et K à moindre coût.
- Les points intermédiaires peuvent être classés en groupes, représentant des "étapes" dans la construction.
- Les différentes étapes de la construction sont notées x_i.
- Les coûts de construction des tronçons ont été évalués et sont reportés sur les différentes arêtes.

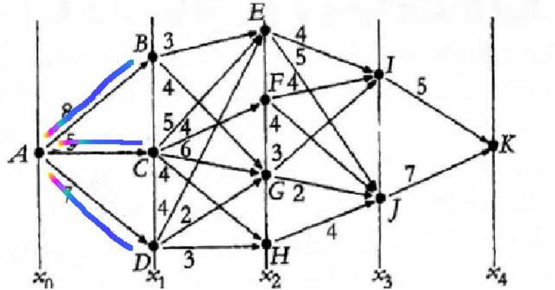
Exemple

Politique de construction de tronçons d'autoroute



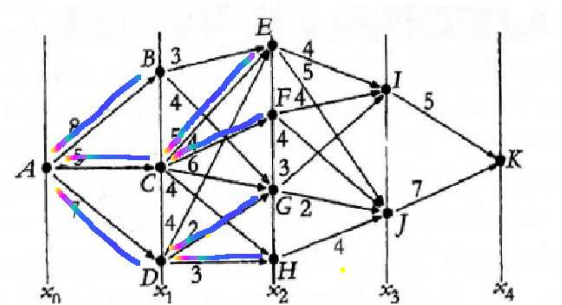
Exemple

- $X_0 = \{A\}$
- $X_1 = \{B, C, D\}$
- $f_1^*(B) = 8, f_1^*(C) = 5, f_1^*(D) = 7$



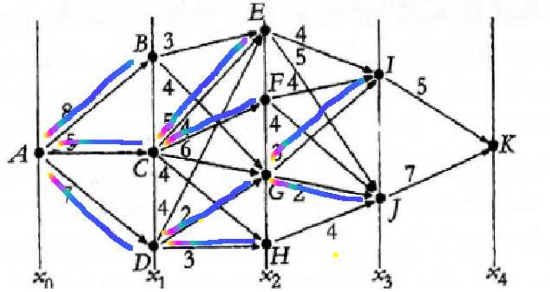
Exemple

- $X_1 = \{B, C, D\}$
- $X_2 = \{E, F, G, H\}$
- $f_2^*(E) = \min\{11, 10\} = 10, f_2^*(F) = 9, f_2^*(G) = 9, f_2^*(H) = 9$



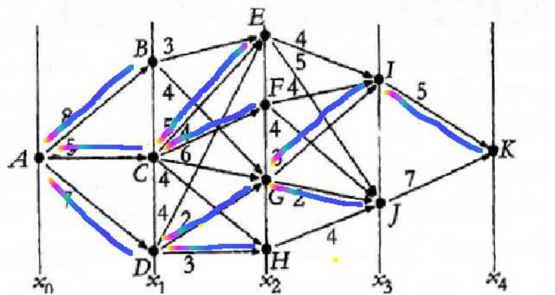
Exemple

- $X_2 = \{E, F, G, H\}$
- $X_3 = \{I, J\}$
- $f_3^*(I) = 12, f_3^*(J) = 11$



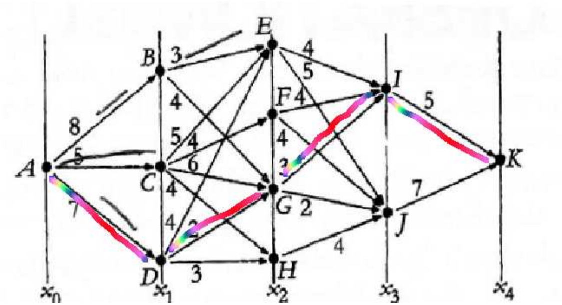
Exemple

- $X_3 = \{I, J\}$
- $X_4 = \{K\}$
- $f_4^*(K) = 17$



Exemple

- Chemin optimal de A à K : $ADGIK$, de valeur 17.



Conclusion

- La programmation dynamique permet de résoudre des problèmes dont la fonction objectif est "décomposable".
- Cela équivaut à la recherche d'un chemin optimal dans des graphes particuliers (séquentiels).
- L'affaiblissement du caractère combinatoire du problème permet une économie de calcul.

Plan

Introduction

Historique

- Avant 1958 : graphique de Gantt (planning à barres), algorithmes de Johnson (cas particuliers de problèmes d'ordonnancement).
- Apparition de la méthode CPM (critical path method), qui a évolué afin d'inclure des données aléatoires en "méthode PERT" (Program Evaluation and Review/Research Technique), USA.
- Apparition de méthode des "potentiels" (MPM), France.

Introduction

Grand principe

- Application directe des méthodes de recherche de chemins optimaux dans un graphe.
- Problème d'ordonnancement : en vue d'un objectif quelconque, il faut accomplir un ensemble de **tâches** soumises à un ensemble de contraintes.

Méthodes PERT et MPM

Objectifs des méthodes

Ces méthodes permettent :

- D'établir un ordonnancement si cela est possible (pas de contraintes contradictoires) ;
- De minimiser le temps total nécessaire à la réalisation de l'objectif ;
- De déterminer les tâches critiques ;
- D'évaluer les marges des tâches critiques.

Méthodes PERT et MPM

Notations

Soient n tâches.

- Date de début : t_i.
- Durée : d_i.
- Date attendue : θ_i.

Méthode PERT

Construction du graphe

- Orienté.
- Sans boucles.
- Les sommets sont des événements (étapes de la réalisation, objectifs intermédiaires inconnus au départ).
- Les arcs représentent les tâches élémentaires (décomposition du processus initial).
- Les arcs sont valués par les durées d'exécution.
- Des tâches fictives sont introduites afin de traduire fidèlement les contraintes.

Méthode PERT : Exemple

Ennoncé

Commande d'un ouvrage technique à un auteur scientifique :

	Opérations	Durée (en quinzaines)	Opérations antérieures
a	approbation du plan de l'ouvrage	1	néant
b	signature du contrat	1	*a*
c	remises du manuscrit	12	*b*
d	approbation du comité de lecture	2	*c*
e	composition du texte	3	*d*
f	correction par les correcteurs de l'imprimerie	1	*e*
g	clichage et tirage des hors-texte	3	*d*
h	exécution des dessins, des figures	4	*d*
i	révision des dessins par l'auteur	1	*h*
j	correction de dessins ; clichage des figures	2	*i*
k	première correction des épreuves par l'auteur	2	*f*
l	exécution des premières corrections à l'imprimerie	1	*k*
m	seconde correction des épreuves par l'auteur ; indication de l'emplacement des clichés ; approbation des hors-texte	2	*g, j, l*
n	exécution des secondes corrections à l'imprimerie ; mise en page	1	*m*
o	tirage du livre	2	*n*
p	établissement de la prière d'insérer, des listes d'exemplaires presse et d'hommage	1	*m*
q	pliage	1	*o*
r	brochage	1	*q*
s	reliure de certains exemplaires	2	*q*
t	impression de la prière d'insérer	1/2	*p*
u	envoi des exemplaires de presse	1/4	*r* et *t*
v	envoi des hommages	1/8	*s* et *t*
w	envoi des contingents aux libraires	1/2	*r* et *s*

Méthode PERT : Exemple

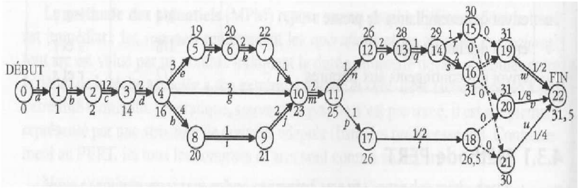
Evénements

Liste des événements :

0. début des opérations ;
1. plan approuvé ;
2. contrat signé ;
3. manuscrit remis ;
4. manuscrit approuvé ;
5. texte composé ;
6. texte corrigé par l'imprimerie ;
7. premières épreuves corrigées par l'auteur ;
8. dessins exécutés ;
9. dessins revus par l'auteur ;
10. premières corrections exécutées par l'imprimerie, secondes épreuves tirées ; clichage terminé pour les figures ; impression achevée pour les hors-texte ; envoi à l'auteur des secondes épreuves, après exécution des premières corrections, de spécimens des hors-texte et des épreuves des clichés des dessins ;
11. retour des secondes épreuves ;
12. secondes corrections exécutées ;
13. tirage exécuté ;
14. pliage exécuté ;
15. brochage exécuté ;
16. reliure exécutée ;
17. prière d'insérer, liste des services de presse et des hommages remises ;
18. prière d'insérer imprimée ;
19. début de l'envoi des contingents aux libraires ;
20. début de l'envoi des hommages ;
21. début de l'envoi des exemplaires de presse ;
22. achèvement des opérations.

Méthode PERT : Exemple

Graphe



Méthode PERT

Chemin critique

- Graphe sans circuit.
- Déterminer le chemin de valeur maximale dans le graphe, afin d'obtenir l'ordonnancement de durée minimale.
- Algorithme de Bellman transposé pour la maximisation.

Méthode PERT

Algorithme de Bellman - $O(m)$

Soit G un graphe orienté valué sans circuit de longueur strictement négative, et s le sommet source de G. L est le tableau des valuations des sommets.

- Initialiser $L(s) = 0$, $\forall i,\ L(x_i) = \infty$.
- Tant que L change faire :
  - Pour chaque arc (x_i, x_j) de G faire :
  - Si $L(x_j) > L(x_i) + v(x_i, x_j)$ alors $L(x_j) \leftarrow L(x_i) + v(x_i, x_j)$ finsi.
  - fin pour
- fin tant que

Méthode PERT

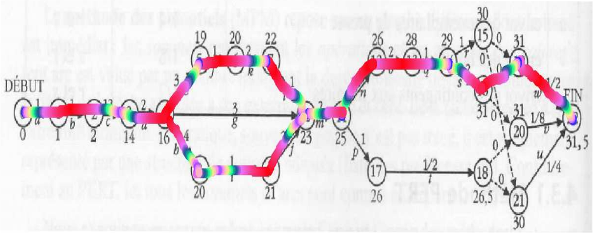
Algorithme de Bellman - $O(m)$ - *Maximisation*

Soit G un graphe orienté valué sans circuit de longueur strictement négative, et s le sommet source de G. L est le tableau des valuations des sommets.

- Initialiser $L(s) = 0$, $\forall i,\ L(x_i) = -\infty$.
- Tant que L change faire :
  - Pour chaque arc (x_i, x_j) de G faire :
  - Si $L(x_j) < L(x_i) + v(x_i, x_j)$ alors $L(x_j) \leftarrow L(x_i) + v(x_i, x_j)$ finsi.
  - fin pour
- fin tant que

Méthode PERT : Exemple

Chemin critique



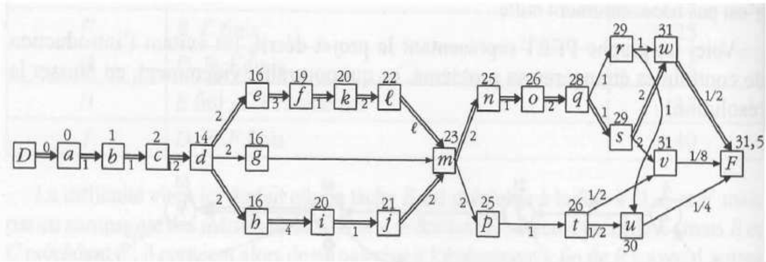
Méthode MPM

Construction du graphe

- Pas de définition préalable d'événements et de tâches fictives.
- Graphe $G = (X, U)$ où les sommets sont les tâches et les arcs les contraintes.
- Introduction d'une tâche de début D et d'une tâche de fin F.

Méthode MPM

Graphe



Méthode MPM

Chemin critique

- Construction d'un tableau de précédences entre tâches.
- Chaque colonne du tableau correspond à une tâche.
- Dans chaque colonne sont inscrites les tâches qui précèdent la tâche de la colonne considérée.
- En face (à droite) de chaque tâche est inscrite sa durée.
- Puis, les dates au plus tôt des tâches sont calculées :
    - On calcule en entête de colonne la date au plus tôt de la tâche.
    - On reporte cette durée en face à gauche de cette tâche dans les autres colonnes.

Méthode MPM

Chemin critique

Opérations	0	a	1	b	2	c	14	d	16	e	19	f	16	g	16	h
Préalables	0	$D:0$	0	$a:1$	1	$b:1$	2	$c:2$	14	$d:2$	16	$e:3$	14	$d:2$	14	$d:2$

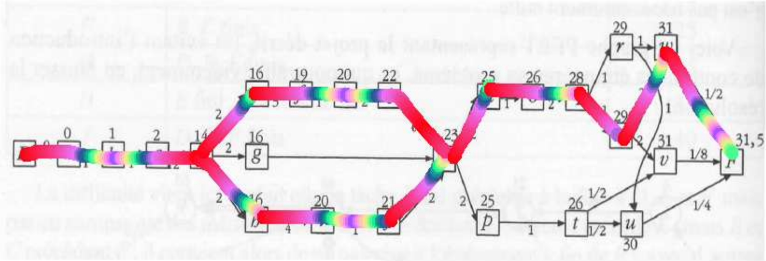
Opérations	20	i	21	j	20	k	22	ℓ	23	m	25	n	26	o	25	p
Préalables	16	$h:4$	20	$i:1$	19	$f:1$	20	$k:2$	16 21 22	$g:3$ $j:2$ $\ell:1$	23	$m:2$	25	$n:1$	23	$m:2$

Opérations	28	q	29	r	29	s	26	t	30	u	31	v	31	w	$31\frac{1}{2}$	F
Préalables	26	$o:2$	28	$q:1$	28	$q:1$	25	$p:1$	29 26	$r:1$ $t:\frac{1}{2}$	26 29	$t:\frac{1}{2}$ $s:2$	29 29	$r:1$ $s:2$	30 31 31	$u:\frac{1}{4}$ $v:\frac{1}{8}$ $w:\frac{1}{2}$

Méthode MPM

Chemin critique

- Valeur du chemin critique : $31,5$ jours.



Conclusion

- L'ordonnancement est un domaine particulièrement actif de la RO (problèmes d'ateliers). L'étude des problèmes d'ordonnancement nécessiterait un cours dédié.
- La méthode PERT est très répandue en gestion de projets.
- La méthode MPM est plus facile à mettre en œuvre que la méthode PERT.

Plan

Introduction

Algorithme de Ford-Fulkerson

- **1954** : "Max-flot min-cut theorem", Lester R. Ford et Delbert Ray Fulkerson.
- Permet de résoudre des problèmes de transport.
- Peut être appliqué aux problèmes d'affectation.

Réseau de transport

Définition

Un réseau de transport est :

- Un graphe fini, sans boucle, comportant n sommets.
- Il comprend un nœud source et un nœud puits.
- Le flot doit circuler dans le réseau du nœud source au nœud puits.
- Les arcs du réseau sont valués par des capacités (tonnages, débits . . .).
- La loi de conservation s'applique en chaque sommet du graphe : le flot qui entre dans un sommet doit en ressortir (sauf pour les sommets source et puits).

Problème de flot

Définition

Etant donné un réseau de transport, il s'agit :

- D'acheminer la quantité maximale de flux dans le réseau.
- La quantité $\varphi(x_i, x_j)$ est la quantité de flux transportée sur l'arc (x_i, x_j).
- La valeur du flot est la somme des flux partant de la source s.
- Elle est également égale à la somme des flux arrivant au puits p.

Algorithme de Ford-Fulkerson

Etant donné un flot dans un réseau de transport, l'algorithme de Ford-Fulkerson permet d'obtenir un flot maximal.

Algorithme

- Initialement la source est marquée $+$. Les autres sommets sont non marqués.
- Tant que cela est possible, choisir un sommet non marqué vérifiant l'une des deux définitions suivantes :
  - Si x_i est l'extrémité terminale d'un arc (x_j, x_i) tel que x_j est marqué et $\varphi(x_j, x_i) < c(x_j, x_i)$, marquer $+$ le sommet x_i.
  - Si x_i est l'extrémité initiale d'un arc (x_i, x_j) tel que x_j est marqué et $\varphi(x_i, x_j) > 0$, marquer $-$ le sommet x_i.
- $p(x_i) \leftarrow x_j$.

Algorithme de Ford-Fulkerson

Flot optimal

Si à la fin de l'algorithme le puits n'est pas marqué, le flot est optimal.

Amélioration du flot

Afin d'améliorer un flot non optimal, il faut :

- Augmenter les flux transportés par les arcs ayant donné lieu à un marquage $+$.
- Diminuer les flux transportés par les arcs ayant donné lieu à un marquage $-$.

Problèmes d'affectation

Grand principe

- Il s'agit d'affecter des objets, des personnes, etc. à des machines, des cours, etc.
- Différents critères d'optimisation sont possibles : satisfaction, coûts...

Une application de Ford-Fulkerson

- Le problème d'affectation est transformé en un problème de transport.
- Utilisation de l'algorithme Hongrois.

Un exemple

Données

- 5 personnes A,B,C,D,E.
- A affecter à 5 emplois a,b,c,d,e.
- De manière à maximiser la satisfaction de ces personnes.
- Préférences concernant les emplois (1 = emploi préféré) :

	a	b	c	d	e
A	1	2	3	4	5
B	1	4	2	5	3
C	3	2	1	5	4
D	1	2	3	5	4
E	2	1	4	3	5

Un exemple

Problème équivalent

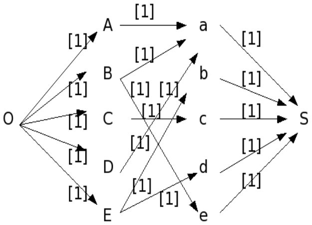
- Soustraire à chaque ligne et à chaque colonne le plus petit élément de la ligne ou de la colonne.
- Cela ne change pas le problème.
- Il faudrait choisir un 0 et un seul par ligne et par colonne.

	a	*b*	*c*	*d*	*e*
A	0	1	2	1	2
B	0	3	1	2	0
C	2	1	0	2	1
D	0	1	2	2	1
E	1	0	3	0	2

Un exemple

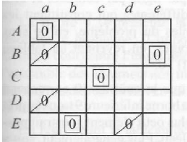
Construction du réseau de transport associé

- Source fictive O, puits fictif S.
- Arcs de capacité 1.
- Les arcs entre A,B,C,D,E et a,b,c,d,e correspondent aux 0.

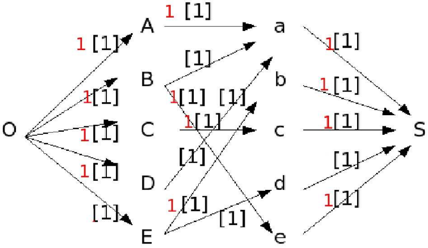


Un exemple

Calcul d’une affectation partielle

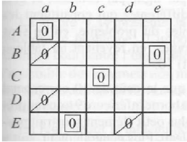


	a	b	c	d	e
A	$\boxed{0}$				
B	$\cancel{0}$				$\boxed{0}$
C			$\boxed{0}$		
D	$\cancel{0}$				
E		$\boxed{0}$		$\cancel{0}$	

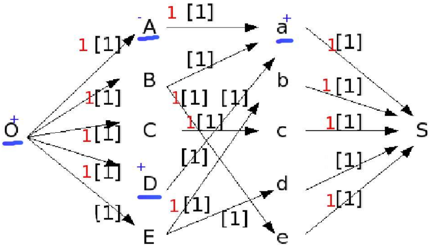


Un exemple

Application de l'algorithme de Ford-Fulkerson



	a	b	c	d	e
A	$\boxed{0}$				
B	$\cancel{0}$				$\boxed{0}$
C			$\boxed{0}$		
D	$\cancel{0}$				
E		$\boxed{0}$		$\cancel{0}$	



Un exemple

Application de l'algorithme de Ford-Fulkerson

- La procédure de marquage n'atteint pas le puits : pas de flot améliorant.
- Cela signifie qu'il est impossible de trouver une meilleure affectation : quelque soient les choix de zéros du tableau, il est impossible d'affecter plus de quatre zéros.
- **Cependant, il faut affecter chaque personne à un poste de manière à maximiser la satisfaction générale. Il est donc nécessaire de recourir à l'algorithme Hongrois**.

Méthode Hongroise

Lignes / colonnes marquées

- Marquer "+" toute une ligne n'ayant pas de 0 affecté.
- Marquer "+" toute colonne ayant un 0 non affecté sur une ligne marquée.
- Marquer "-" toute ligne ayant un 0 affecté sur une colonne marquée.
- Itérer jusqu'à ce que le marquage ne soit plus possible.
- Tracer un trait sur les lignes non marquées et les colonnes marquées.

Méthode Hongroise

Ajout des arcs de plus faible coût

- Retrancher le plus petit nombre du tableau restant à tous les éléments non rayés.
- Ajouter le plus petit nombre du tableau restant à tous les éléments rayés deux fois.

Les opérations impliquant le plus petit nombre du tableau restant reviennent à ajouter au graphe les arcs de plus faible coût.

Méthode Hongroise

Itération

- Calcul d’une affectation (partielle ou non).
- Dans le cas d’une affectation partiellle, existe-t-il un flot améliorant ? Application de l’algorithme de Ford-Fulkerson.
  - Calcul du flox maximal.
  - Est-ce toujours une affectation partielle ? Si non : fin de l’algorithme.
- Itérer.

Un exemple

Méthode Hongroise

- D et a sont marquées +
- A est marquée -
- a, B, C et E sont rayées

	a	b	c	d	e	
A	0	1	2	1	2	-
B	0	3	1	2	0	
C	2	1	0	2	1	
D	0	1	2	2	1	+
E	1	0	3	0	2	
	+					

Un exemple

Méthode Hongroise

- Le plus petit nombre est 1.
- Il est ajouté aux éléments rayés deux fois et retranché aux éléments non rayés.

	a	b	c	d	e
A	0	0	1	0	1
B	1	3	1	2	0
C	3	1	0	2	1
D	0	0	1	1	0
E	2	0	3	0	2

Un exemple

Itération

Une affectation totale est obtenue. L'algorithme est terminé.

	a	b	c	d	e
A	0	0	1	0	1
B	1	3	1	2	0
C	3	1	0	2	1
D	0	0	1	1	0
E	2	0	3	0	2

Conclusion

- Application de la recherche de flot maximal à des problèmes d'affectation.
- Utilisation de l'algorithme Hongrois (en hommage à Ergevary et König), impliquant l'algorithme de Ford-Fulkerson.
- Il existe d'autres moyens de résoudre les problèmes d'affectation...

Plan

Introduction

Historique

- **1962 :** Méthode booléenne de Faure et Malgrange, pour l'optimisation d'une fonction booléenne. Parcours en profondeur d'abord.
- **1963 :** Recherche arborescente pour le voyageur de commerce, Little et al.
- **1964-1965 :** Méthodes arborescentes SEP (séparation et évaluation progressive), B. Roy.

Introduction

Grands principes

- Problèmes NP-difficiles : ne peuvent être résolus que par des méthodes énumératives.
- Quelques exemples : voyageur de commerce, bin-packing, PLNE...
- Enumération "intelligente" des solutions, de manière à ne pas devoir toutes les énumérer.

Méthode SEP

Arborescence

- Une arborescence correspondant au problème est développée au cours de l'algorithme.
- Chaque sommet de l'arborescence correspond à un sous-ensemble de solutions admissibles.
- La racine de l'arborescence correspond à toutes les solutions réalisables.

Méthode SEP

Evaluation

- Pour chaque sommet S_i, une valeur $E(S_i)$ appelée évaluation du sommet est calculée.
- Pour un problème de maximisation, cette valeur doit être un majorant de la meilleure solution contenue dans l'ensemble des solutions correspondant au sommet de l'arborescence considérée.
- Si l'évaluation du sommet est inférieure (pour une maximisation), alors le sommet n'est pas exploré.
- Lorsque, pour un sommet S_i, la meilleure solution de l'ensemble est obtenue, ou que l'ensemble ne contient pas de solution, l'exploration de ce sommet est terminée.

Méthode SEP

Séparation

- Lorsque la meilleure solution de l'ensemble S_i n'est pas obtenue, et qu'il contient des solutions, S_i est séparé en plusieurs sous-ensembles.
- Les successeurs de S_i dans l'arborescence sont les sommets correspondant à ces ensembles.

Stratégie de parcours de l'arborescence

- Un parcours en profondeur d'abord.
- Un parcours dans lequel le sommet à explorer est celui possédant la meilleure évaluation parmi ceux non encore explorés (S.E.P.).

Exemple

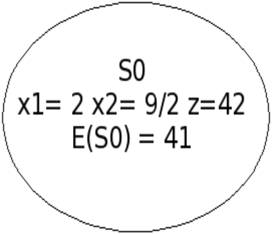
PLNE

$$\max z = 3x_1 + 8x_2$$

$$\begin{aligned} x_1 + 4x_2 &\leq 20 \\ x_1 + 2x_2 &\leq 11 \\ 3x_1 + 2x_2 &\leq 22 \\ x_1, x_2 &\in \mathbb{N} \end{aligned}$$

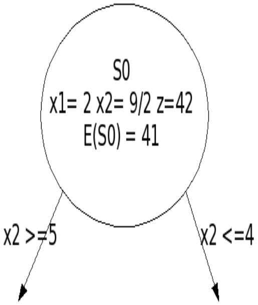
Exemple

- Résolution de la relaxation linéaire du problème



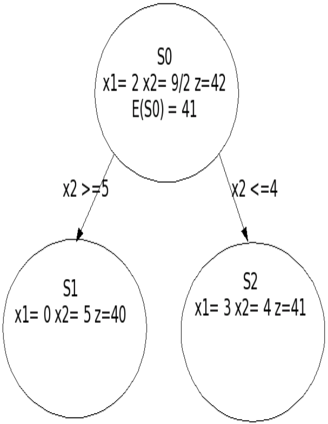
Exemple

- Séparation :



Exemple

- Deux problèmes différents à résoudre, ajout de contraintes :



Exemple

- Solutions entières "naturellement".
- Solution optimale :

$$x_1 = 3$$
$$x_2 = 4$$
$$z = 41$$

Conclusion

- Problèmes NP-difficiles résolus par méthode SEP.
- Stratégie de calcul de l'évaluation (borne) primordiale (heuristique, problème relaxé...)

Plan

Conclusion

- Les graphes sont très utilisés en optimisation combinatoire.
- D'autres exemples seront étudiés dans les cours suivants (graphes bipartis...).