

Examen terminal d'optimisation combinatoire
Durée 2h, le 23 juin 2006
Le barème est indicatif

Exercice 1 Sac à dos(4 points)

Développez au choix, la dernière méthode arborescente vue en cours ou l'algorithme de programmation dynamique pour le problème de sac à dos ci-dessous avec $P = 8$.

i	1	2	3	4	5	6
w_i	18	5	10	8	12	18
p_i	3	1	2	2	4	6

Exercice 2 Equipe sportive (8 points)

On considère un ensemble de n joueurs. Chaque joueur j possède un indice de performance, résultant de ses activités antérieures, noté u_j et appelé l'utilité du joueur. Un entraîneur souhaite constituer une équipe de 23 joueurs d'utilité maximale (l'utilité d'une équipe étant la somme des utilités de ses joueurs).

Mais pour faire son choix il doit se soumettre à certaines obligations. Tout d'abord il dispose d'une enveloppe financière maximale F , sachant que la sélection d'un joueur j nécessite de payer une somme c_j à son club d'origine.

D'autre part l'entraîneur dispose d'un graphe d'inimitié : les sommets représentent les joueurs, et une arête entre deux sommets i et j indique que les deux joueurs ne peuvent pas jouer dans la même équipe.

1 Modéliser le problème de l'entraîneur, et notons le $P1$. Notons $Z1$ l'utilité maximale d'une équipe pour ce problème.

2 Considérons maintenant le problème obtenu en supposant que l'équipe n'a pas un nombre de joueurs donné à l'avance, mais qu'elle peut-être de taille variable. Que faut-il changer au modèle ? Notons le $P2$, et $Z2$ l'utilité maximale d'une équipe. Quelle relation y-a-t-il entre $Z1$ et $Z2$ pour les mêmes données (nombre de joueurs, graphe d'inimitié, coûts et utilités des joueurs) ?

3 Considérons maintenant le problème $P3$ obtenu à partir de $P2$ en remplaçant le graphe d'inimitié par un graphe vide (en supposant que tous les joueurs s'aiment). Notons $Z3$ l'utilité maximale d'une équipe pour le

problème $P3$. Quelle relation y-a-t-il entre $Z3$ et $Z2$ pour les mêmes données (nombre de joueurs, coûts et utilités des joueurs) ?

4 Quel algorithme pouvez-vous proposer pour majorer $Z3$?

5 Proposer un algorithme permettant de construire une équipe solution du problème $P2$ (pas nécessairement optimale).

Dans la suite on s'intéresse à une méthode arborescente pour résoudre le problème $P2$. Un noeud de l'arborescence S est caractérisé par un sous-ensemble de joueurs $C(S)$ choisis et un sous-ensemble de joueurs $R(S)$ rejetés. Il est associé à l'ensemble des équipes (qui rentrent dans l'enveloppe financière et respectent les inimitiés) contenant les joueurs de $C(S)$ et pas ceux de $R(S)$.

La séparation d'un noeud S se fait en déterminant un joueur j ni choisi ni rejeté, ni ennemi (selon le graphe) d'un joueur de $C(S)$ et en construisant deux fils S', S'' . Dans S' on ajoute j aux joueurs choisis, dans S'' on l'ajoute aux joueurs rejetés.

6 Quel algorithme pouvez-vous proposer obtenir une évaluation par excès du noeud S ?

7 Quel algorithme pouvez-vous proposer obtenir une évaluation par défaut du noeud S ?

Exercice 3 Deux machines(13 points)

Un atelier de production dispose de deux machines qui peuvent fonctionner en parallèle. En début de journée, un ensemble de tâches est proposée à l'atelier qui doivent être réalisées dans la journée. La plupart du temps, il y en a trop pour être réalisées. Le responsable d'atelier doit donc décider quelles sont les tâches qui sont acceptées par son atelier (celles qui peuvent être réalisées dans la journée), et sur quelle machine elles s'effectuent. Pour une tâche i la durée n'est pas la même selon la machine qu'on utilise (durées respectives a_i et b_i). De plus, le fait de rejeter une tâche i induit un coût c_i pour l'entreprise. On cherche donc à ordonnancer les tâches sur les deux machines de sorte que la durée des tâches sur chacune des machines soit inférieure à la durée ouvrable de la journée J en minimisant la somme des coûts des tâches rejetées.

1 Modéliser ce problème à l'aide d'un programme mathématique.

On définit $F_k(A, B)$ le coût minimal d'un ordonnancement des tâches k à n pour lequel la durée des tâches sur la machine 1 est inférieure ou égale

à A , celle sur la machine 2 est inférieure ou égale à B .

2 Que représente $F_1(J, J)$?

3 Donner la valeur de $F_n(A, B)$ en fonction des données du problème et de A, B

4 Montrer que si $A < a_k, B < b_k$ alors $F_k(A, B) = \min(c_k + F_{k+1}(A, B))$

5 Montrer que si $A \geq a_k, B < b_k$ alors $F_k(A, B) = \min(c_k + F_{k+1}(A, B), F_{k+1}(A - a_k, B))$

6 Que dire si $A < a_k, B < b_k$?

7 Montrer que si $A \geq a_k, B \geq b_k$ alors $F_k(A, B) = \min(c_k + F_{k+1}(A, B), F_{k+1}(A - a_k, B), F_{k+1}(A, B - b_k))$

8 Ecrire l'algorithme qui découle de ces égalités permettant de déterminer les vitesses à utiliser pour chaque tâche à l'optimum. On précisera sa complexité.