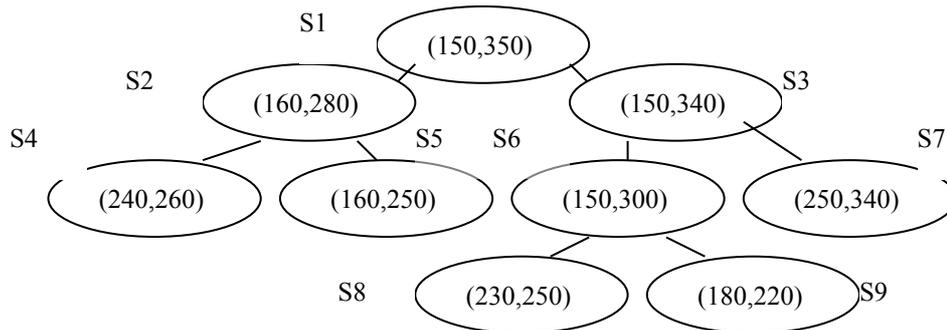


## Interrogation

Durée 1h30 Documents non autorisés

### Une arborescence (3 points)

On se donne un problème de maximisation, pour lequel une méthode arborescente a été partiellement exécutée. Chaque nœud possède une évaluation par défaut et une évaluation par excès. L'évaluation par défaut correspond à une solution réalisable du nœud déterminée grâce à une heuristique. Le schéma ci-dessous résume l'état de l'exploration de l'arbre. Dans chaque nœud figurent le couple d'évaluation (défaut, excès). Que peut-on en déduire pour la poursuite de la méthode ? Quel est le prochain nœud à séparer selon la stratégie de parcours adoptée ?



Meilleure solution rencontrée : valeur 250. Cela permet de tronquer les nodes S9, S8, S5 dont l'évaluation par excès est inférieure à 250. Seuls restent ouverts les nœuds S4 et S7. Dans le cas d'un parcours en largeur d'abord S4 serait d'abord séparé, S7 dans les deux autres cas (meilleure évaluation par excès et nœud le plus récent parmi les nœuds ouverts).

### Un problème d'ordonnancement.(7 points)

Les étudiants de M1 classique ont eu à définir une méthode arborescente pour le problème suivant : On se donne  $n$  tâches caractérisées par des durées  $p_i$ , des dates d'échéance  $d_i$ , des profits  $w_i$ . Une tâche rapporte  $w_i$  si elle se termine avant ou à sa date d'échéance, et ne rapporte rien sinon. Comment déterminer un ordonnancement des tâches sur une machine qui maximise la somme des profits des tâches à l'heure ? Les étudiants de classique ont proposé plusieurs solutions. Discuter de la validité des solutions proposées, et appliquer la méthode qui vous semble la plus appropriée au problème suivant :

$i$	1	2	3	4	5
$p_i$	3	5	4	2	3
$d_i$	5	6	7	7	10
$w_i$	8	5	7	10	12

### Solution proposée n°1 :

#### Description du sous-ensemble associé aux nœuds :

Noeud  $S$  : une suite de tâches  $J_s$ ,

$D(S)$  : l'ensemble des ordonnancements qui commencent par  $J_s$ .

C'est une définition valide, puisqu'à chaque nœud correspond bien un ensemble de solutions.

#### Opération de séparation :

Pour chaque tâche  $i$  n' appartenant pas à  $J_s$ , il y aura un fils de  $S$ , donc sa séquence sera  $J_s.i$

C'est une définition valide, puisque les ensembles des solutions associés aux fils d'un nœud  $S$  recouvrent bien l'ensemble des ordonnancements associés au nœuds  $S$ . En effet, soient  $i_1, \dots, i_k$  les tâches qui ne sont pas dans  $J_s$ . Les ordonnancement qui commencent par  $J_s$  peuvent se décomposer en

- les ordos qui commencent par  $J_s$  suivi de  $i_1$
- ceux qui commencent par  $J_s$  suivi de  $i_2$
- ...
- ceux qui commencent par  $J_s$  suivi de  $i_k$ .

la réunion, pour tous les tâches  $i$  qui ne sont pas dans  $J_s$ , des ordonnancements qui commencent par  $J_s$ .  $i$  est donc égale à l'ensemble des ordonnancements qui commence par  $J_s$

**Fonction d'évaluation par excès :**

Pour calculer l'évaluation par excès, on va placer les tâches  $i$  n'appartenant pas à  $J_s$  dans un certain ordre sur la machine à la suite de  $J_s$  par exemple par  $(d_i/p_i) \cdot w_i$  décroissant puis on calcule le profit associé :  $h(s)$

Une évaluation par excès doit vérifier que  $\text{Max} \{f(x) / x \text{ appartient à } D(S)\} \leq h(S)$   
 Donc que pour tout ordonnancement de  $D(S)$  le profit de cet ordonnancement doit être inférieur ou égal à  $h(S)$ . Or ici, on propose de définir  $h(S)$  comme un ordonnancement particulier de  $D(S)$  (c'est bien un ordonnancement qui commence par  $S$ ). Par conséquent, il n'a pas forcément un profit meilleur que tous les autres. Un exemple simple : on prend deux tâches de durée 1, d'échéances 1 et 2 et de profit respectifs 1 et 2. Considérons le nœud racine (où  $J_s$  est vide). De toute évidence, la deuxième tâche est placée avant la première dans la solution construite, donc  $h(s)$  vaut 2. Or en mettant la tâche 1 puis la tâche 2, on obtient un ordo qui respecte les échéances et qui rapporte 3.

On peut remarquer qu'en fait le  $h(S)$  décrit ici est une évaluation par défaut, puisqu'une solution particulière de  $D(S)$  est toujours majorée par la meilleure solution de  $D(S)$ .

**Fonction d'évaluation par défaut :**

Pour chaque noeud on connaît le profit des tâches dans  $J_s$  donc  $g(s) = \text{Somme } w_i x_i$  avec  $i$  appartient à  $J_s$ . ( $x_i = 0$  en retard  $x_i = 1$  à l'heure)

Il s'agit bien d'une évaluation par défaut, mais elle est particulièrement mauvaise. En effet, on doit vérifier que  $\text{Max} \{f(x) / x \text{ appartient à } D(S)\} \geq g(S)$ . Or tous les ordonnancements commençant par  $J_s$  rapporteront au moins le profit des tâches à l'heure de la séquence  $J_s$  (plus éventuellement les quelques autres tâches à l'heure placées après). Donc ce  $g(S)$  est inférieur au profit de n'importe quel ordonnancement de  $D(S)$ . Alors qu'on veut seulement qu'il soit inférieur au meilleur d'entre eux.

**Parcours qui semble le mieux adapté à cette arborescence :**

→ Pour le parcours on choisit par le meilleur d'abord.  
 Là on ne peut rien dire de particulier.

**Solution proposée n°2 :**

**Description du sous-ensemble associé aux nœuds :**

Un nœud  $S$  sera associé à une sous-suite de tâche  $J_s = \{i_1, \dots, i_r\}$ , et comprend l'ensemble des ordonnancements  $\sigma$  qui commencent par la séquence  $J_s$ .

C'est la même chose que dans la solution 1

**Opération de séparation :**

$S$  aura autant de fils que de tâches  $i$  n'appartenant pas à  $J_s$ .

Pour chaque tâche  $i$  n'appartenant pas à  $J_s$  on crée un fils à  $S$  dont la sous-suite associée est égale à  $\{J_s, i\}$ .

C'est la même chose que dans la solution 1. C'est donc correct.

**Fonction d'évaluation par excès :**

$h(S) = \text{somme des profits des tâches à l'heure de } J_s + \text{somme des profits des tâches n'appartenant pas à } J_s \text{ dont l'échéance est supérieure à la date de fin de } J_s$ .

Soit un ordonnancement de  $D(S)$ , c'est-à-dire un qui commence par  $J_s$ . Prenons une tâche qui n'appartient pas à  $J_s$ . Si son échéance est inférieure ou égale à la fin de  $J_s$ , comme elle est placée après  $J_s$ , elle sera forcément en retard. Donc dans l'ordonnancement cette tâche ne rapporte rien. Par contre si son échéance est après la fin de  $J_s$ , alors dans l'ordonnancement cette tâche rapporte au maximum  $w_i$ . Par conséquent, le profit de l'ordonnancement est bien majoré par le  $h(S)$  proposé : d'où  $\text{Max} \{f(x) / x \text{ appartient à } D(S)\} \leq h(S)$ , c'est une évaluation par excès.

**Fonction d'évaluation par défaut :**

$g(S)$  = somme des profits des tâches à l'heure de  $J_s$  + somme des profits des tâches n'appartenant pas à  $J_s$  ordonnées dans l'ordre croissant de leur  $d_i$  et qui sont à l'heure (celles qui sont en retard dans cet ordre sont replacées à la fin).

Hypothèse pour le calcul de  $g$  : les tâches sont rangées dans l'ordre croissant de leur  $d_i$

Si l'on prend n'importe quel ordonnancement de  $D(S)$  (c'est-à-dire qui commence par  $J_s$ ), le profit qu'il rapporte sera toujours inférieur au meilleur profit possible d'une telle solution. Donc tout algorithme de construction d'une solution va bien calculer une évaluation par défaut valide, c'est-à-dire inférieure à  $\text{Max} \{f(x) / x \text{ appartient à } D(S)\}$

**Parcours qui semble le mieux adapté à cette arborescence :**

Parcours en profondeur d'abord.

### **Solution proposée n°3 :**

**Description du sous-ensemble associé aux nœuds :**

A chaque nœud  $S$ , on associe une suite de tâches  $J_s$

Soit  $D(S)$  l'ensemble des ordonnancements se finissant par la suite  $J_s$ .

C'est tout à fait valide (pour les mêmes raisons que précédemment)

**Opération de séparation :**

Pour chaque tâche  $i$  n'appartenant pas à  $J_s$ , il y aura un fils de  $S$ , dont la séquence sera  $i.J_s$

C'est tout à fait valide pour les mêmes raisons que précédemment à ceci près qu'ici on décompose l'ensemble des ordonnancements qui se terminent par  $J_s$  en ceux qui se terminent par une autre tâche suivie de  $J_s$ , selon cette autre tâche.

**Fonction d'évaluation par excès :**

Pour calculer l'évaluation par excès, on va placer les tâches  $i$  n'appartenant pas à  $J_s$  dans un certain ordre, par exemple par  $d_i$  décroissant

puis on calcule le profit associé :  $h(s) = g(S) + \text{profit des tâches n'appartenant pas à } J_s$ .

Ce n'est pas une évaluation par excès pour les mêmes raisons que la solution 1.

Construire une solution particulière conduit en fait à obtenir une évaluation par défaut !!

**Fonction d'évaluation par défaut :**

La fonction d'évaluation choisie :

$g(S)$  = somme des profits des tâches dites à l'heure et appartenant à  $J_s$ .

C'est une mauvaise évaluation par défaut pour les mêmes raisons que pour la solution 1.

**Parcours qui semble le mieux adapté à cette arborescence :**

Parcours en largeur d'abord.

En ce qui concerne l'application de la méthode n° 2 à l'exemple, il convient de préciser un peu plus la fonction d'évaluation par défaut qui présente encore certaines ambiguïtés.

On considèrera ici que l'algorithme employé est le suivant:

on place les tâches de la séquence  $J_s$  dans l'ordre, pour former la séquence de début  $D$ , et on définit une séquence de fin  $F$  initialement vide. Ensuite on parcourt les tâches qui ne sont pas dans  $J_s$  dans l'ordre croissant des échéances. Si une tâche  $i$ , lorsqu'elle est placée à la fin de  $D$ , est à l'heure, on pose  $D=D.i$ , sinon on pose  $F=F.i$ . Lorsque toutes les tâches sont parcourues, l'ordonnancement est constitué de la séquence  $D$  suivie de la séquence  $F$ , et  $g(S)$  est le profit de cet ordonnancement.

Ainsi pour l'exemple, au nœud racine  $R$   $J_R$  est vide. On place donc dans  $D$  la tâche 1, puis dans  $F$  la tâche 2, puis dans  $D$  la tâche 3, dans  $F$  la tâche 4 et dans  $D$  la tâche 5, ce qui donne l'ordonnancement 1,3,5,2,4 qui est de profit 27 (somme des profits des tâches à l'heure 1, 3 et 5).

En appliquant la méthode 2 telle qu'elle est définie, on obtient une arborescence plutôt importante résumée dans le schéma suivant. On précise également par la suite quelques améliorations qui peuvent être apportées à la méthode.