ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Section d'Informatique et de Systèmes de Communication

Corrigé de la série 7 9 Novembre 2007

1. Sac à dos 0/1

a) La valeur maximale est 45. Il y a deux solutions correctes : les ensembles qui atteignent ce maximum sont :

$$\{A, C, E\} \qquad \text{et} \qquad \{B, D, E\}$$

Pour les détails voir la figure 1.

Les flèches diagonales corréspondent à la branche passant par le pas 9 de l'algorithme du polycopié et les flèches verticales aux branches passant par 11 ou 14. Les flèches en gris sont les décisions rejétés parce que l'autre chemin est mieux.

Les flèches en gras donnent des chemins optimaux (flèche diagonale à la $i$ème étape : on prend l'objet i ; flèche verticale : on ne le prend pas).

b) La solution optimale n'est pas unique, c.f. le corrigé du point précédent. On peut adapter l'algorithme pour trouver toutes les solutions optimales. La méthode la plus simple est d'utiliser une récursion pour le faire.

Call : GetOptimalChoice2(W, n, v_i, C)
Input: L'input à Knapsack et le C calculé. $x[]$ est statique.
Output: Un choix optimal.

```
if n ≥ 1 then
    if c_{i-1,W} = c_{i,W} then
        x[n] ← 0
        GetOptimalChoice2(W, n − 1, v_i, C)
    if c_{i-1,W-w_i} + v_i = c_{i,W} then
        x[n] ← 1
        GetOptimalChoice2(W − w_i, n − 1, v_i, C)
else
    Print "Solution :" x[]
```

c) Chaque solution (optimale ou non, admissible ou non) est un sous-ensemble de

$$\{1, 2, \ldots, n\}.$$

Donc, l'affirmation est vraie parce que $|\mathrm{Pot}(\{1, 2, \ldots, n\})| = 2^n$.

d) On peut choisir $v_i = 1$ et $w_i = 1$ pour $i = 1, \ldots, n$ et $W = n/2$. Les solutions optimales sont alors des sous-ensembles de $\{1, \ldots, n\}$ de taille $n/2$, et il y en a $\binom{n}{n/2}$. Par l'exercice 3b de la série 4, on a

$$\binom{n}{n/2} > 2^{n/2},$$

d'où le résultat.

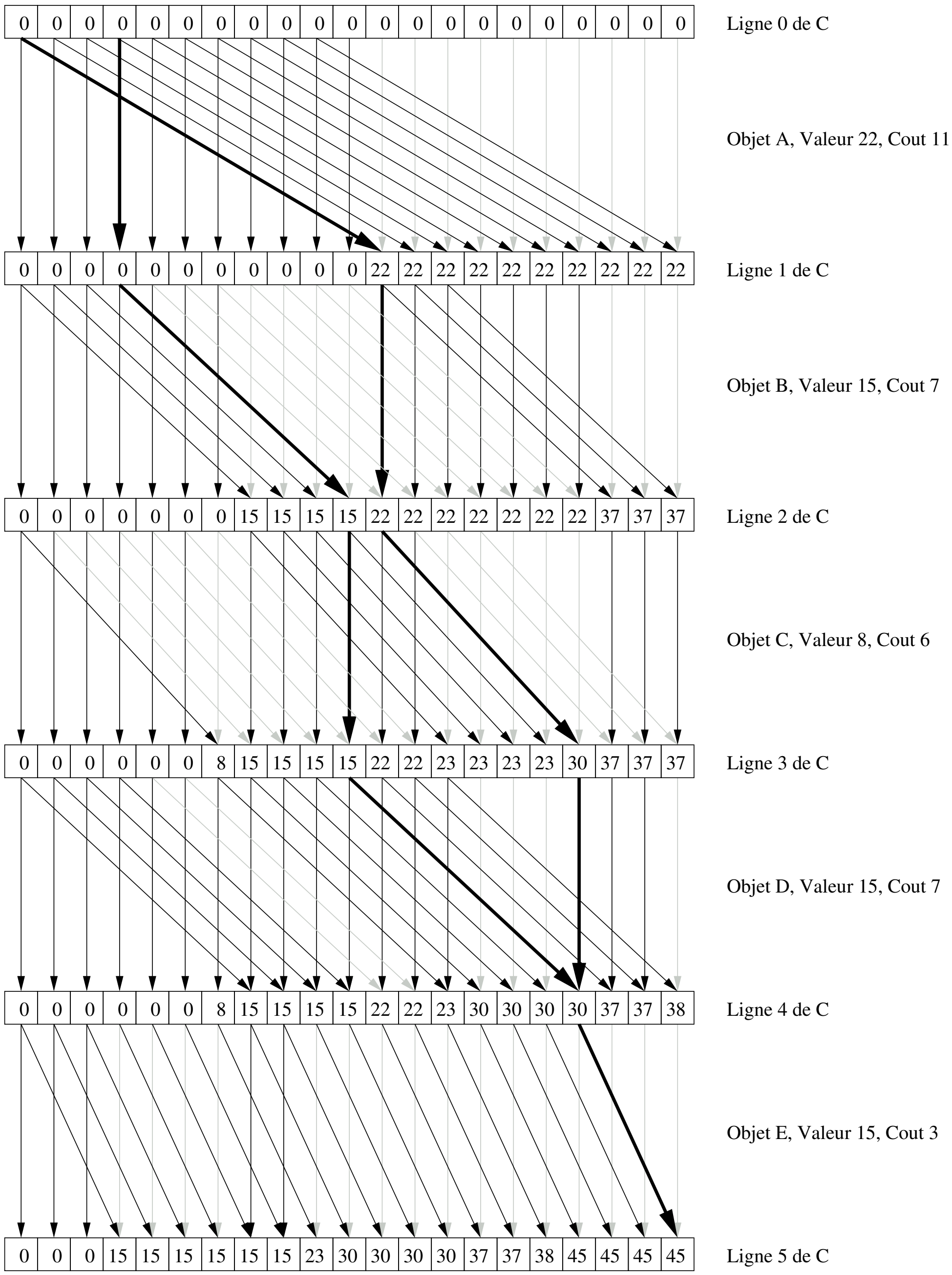
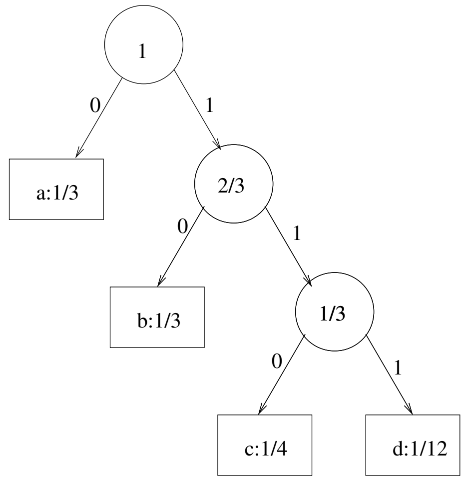


Fig. 1 – Knapsack 0/1

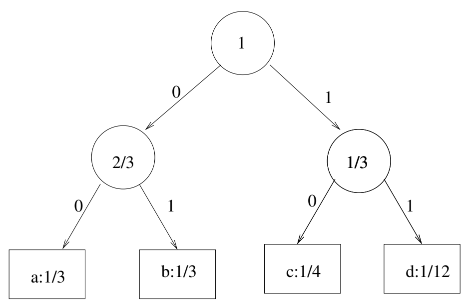
2. Le code optimal n'est pas unique

a) i) On peut construire un code de Huffman C_1 comme suit :



Les longueurs des mots de ce code sont $(1, 2, 3, 3)$.

ii) Un autre arbre de Huffman C_2 pour le même alphabet et les mêmes fréquences est :



Les longueurs de ce code sont $(2, 2, 2, 2)$.

iii) La longueur moyenne de C_1 est

$$\bar{L}_1 = 1 \cdot \frac{1}{3} + 2 \cdot \frac{1}{3} + 3 \cdot \frac{1}{4} + 3 \cdot \frac{1}{12} = 2.$$

Celle de C_2 est

$$\bar{L}_2 = 2 \cdot \frac{1}{3} + 2 \cdot \frac{1}{3} + 2 \cdot \frac{1}{4} + 2 \cdot \frac{1}{12} = 2.$$

Les deux ont la même longueur moyenne et les deux sont optimaux.

b) Oui. Prenons l'alphabet $\{a, b, c, d\}$ avec les fréquences

$$\{1/4 - \varepsilon, 1/4 - \varepsilon, 1/4 + \varepsilon, 1/4 + \varepsilon\},$$

où ε est assez petit, p.e. $\varepsilon < 1/12$ suffit. Le code

$$a \mapsto 00, b \mapsto 10, c \mapsto 01, d \mapsto 11$$

n'est pas un code de Huffman : dans un code de Huffman, a et b auront forcément le même parent.

D'autre part, c'est un code optimal parce que la longueur moyenne est deux, et il est facile de vérifier qu'elle l'est aussi pour le code de Huffman correspondant à ces poids.

3. Multiplication de plusieurs matrices

Rappelons que $m_{i,j}$ est le nombre minimal de multiplications scalaires pour le calcul de $A_{i...j}$ et on a :

$$m_{ij} = \begin{cases} 0 & \text{si } i = j = 0, \\ \min_{i \leq k < j}\{m_{ik} + m_{k+1,j} + p_{i-1} \cdot p_k \cdot p_j\} & \text{si } 1 \leq i \leq j \leq n. \end{cases}$$

Rappelons aussi que s_{ij} est la valeur de k pour laquelle le minimum de la formule est atteint. Considerons que $M[i,j] = m_{ij}$ et $S[i,j] = s_{i,j}$.

$$\begin{array}{rcl} m_{1,2} & = & 5 \cdot 10 \cdot 3 = 150 \\ m_{2,3} & = & 10 \cdot 3 \cdot 12 = 360 \\ m_{3,4} & = & 3 \cdot 12 \cdot 5 = 180 \\ m_{1,3} & = & \min \begin{cases} m_{1,2} + 5 \cdot 3 \cdot 12 = 330 \\ m_{2,3} + 5 \cdot 10 \cdot 12 = 960 \end{cases} \\ m_{2,4} & = & \min \begin{cases} m_{2,3} + 10 \cdot 12 \cdot 5 = 960 \\ m_{3,4} + 10 \cdot 3 \cdot 5 = 330 \end{cases} \\ m_{1,4} & = & \min \begin{cases} m_{2,4} + 5 \cdot 10 \cdot 5 = 580 \\ m_{1,2} + m_{3,4} + 5 \cdot 3 \cdot 5 = 405 \\ m_{1,3} + 5 \cdot 12 \cdot 5 = 630 \end{cases} \end{array}$$

Les matrices M et S sont alors :

$$M = \begin{pmatrix} 0 & 150 & 330 & 405 \\ & 0 & 360 & 330 \\ & & 0 & 180 \\ & & & 0 \end{pmatrix} \qquad S = \begin{pmatrix} 0 & 1 & 2 & 2 \\ & 0 & 2 & 2 \\ & & 0 & 3 \\ & & & 0 \end{pmatrix}$$

$S[1,4] = 2$ signifie que $A_1 \cdot A_2 \cdot A_3 \cdot A_4$ est construit de manière optimale à partir de $A_1 \cdot A_2$ et $A_3 \cdot A_4$. Le nombre de multiplications nécessaires est donc 405.

4. Le nombre de façons dont on peut placer des parenthèses

a) Pour cette partie il fallait en fait utiliser le fait que C_n est égal au nombre de façons dont peuvent être disposées les parenthèses dans le produit $A_1 \cdots A_n$.

Avec cette supposition, il est clair que $C_1 = 1$ et $C_2 = 1$. Pour le produit $A_1 \cdot A_2 \cdot A_3$, le nombre de façons dont on peut placer les parenthèses est 2 :

$$\begin{array}{c} (A_1 \cdot A_2) \cdot A_3 \\ A_1 \cdot (A_2 \cdot A_3) \end{array}$$

Ainsi $C_3 = 2$. Et on verifie que $C_3 = C_1 \cdot C_2 + C_2 \cdot C_1$.

Pour $n = 4$ on peut diviser le produit $A_1 \cdots A_4$ en deux sous-produits de trois façons :

i) $A_1 \cdot (A_2 \cdot A_3 \cdot A_4)$

ii) $(A_1 \cdot A_2) \cdot (A_3 \cdot A_4)$

iii) $(A_1 \cdot A_2 \cdot A_3) \cdot A_4$

Comment on peut placer les autres parenthèses ? Dans le premier cas, il y a $C_1 \cdot C_3$ façons, dans le cas (ii), il y a $C_2 \cdot C_2$ façons, et dans le dernier cas il y a $C_1 \cdot C_3$ façons. En ajoutant on peut calculer C_4 :

$$C_4 = C_1 \cdot C_3 + C_2 \cdot C_2 + C_3 \cdot C_1$$

En général, un produit $A_1 \cdots A_n$ complètement parenthésé est le produit de deux sous-produits complètement parenthésés. Le premier facteur est le produit $A_{1 \ldots k}$, le deuxième le produit $A_{k+1 \ldots k}$. Tous les cas $k = 1, 2, \ldots, n-1$ sont possibles. Donc on a

$$C_n = C_1 \cdot C_{n-1} + C_2 \cdot C_{n-2} + \ldots + C_{n-1} \cdot C_1 = \sum_{i=1}^{n-1} C_i \cdot C_{n-i}$$

b) – Commençons par montrer que $\binom{2n}{n} > 2^n$:

$$\begin{aligned}
\binom{2n}{n} &= \tfrac{2n \cdot (2n-1) \cdot (2n-2) \cdots (2n-(n-1))}{n \cdot (n-1) \cdot (n-2) \cdots 1} \\
&= (\tfrac{2n}{n}) \cdot (\tfrac{2n-1}{n-1}) \cdot (\tfrac{2n-2}{n-2}) \cdots (\tfrac{2n-(n-1)}{1}) \\
&> \underbrace{2 \cdot 2 \cdot 2 \cdots 2}_{n \text{ fois}} \\
&> 2^n
\end{aligned}$$

– Il reste à voir que $4^n > \binom{2n}{n}$. Rappelons que $(x+y)^{2n} = \sum_{i=1}^{2n} \binom{2n}{i} x^i y^{2n-i}$. En utilisant ce fait avec $x = 1$ et $y = 1$, on trouve que

$$\begin{aligned}
4^n &= 2^{2n} \\
&= \textstyle\sum_{i=1}^{2n} \binom{2n}{i} \\
&= \binom{2n}{1} + \ldots + \binom{2n}{n} + \ldots + \binom{2n}{2n} \\
&> \binom{2n}{n}.
\end{aligned}$$

c) Soit A l'ensemble des mots inacceptables[1], et B l'ensemble des suites avec $n-1$ "(" et $n+1$ ")". Il y a en tout $\binom{2n}{n-1}$ mots dans B (pour en construire un, on choisit la position de chacun des $n-1$ "(" parmi les $2n$ possibilités pour définir un tel mot). Nous montrerons que la cardinalité de A est égale à la cardinalité de B, ce qui terminera l'exercice.

Pour ce faire, nous associerons à chaque $a \in A$ un $b \in B$ de façon à ce que des différents éléments de A seront envoyés sur des différents éléments de B, et tel que tout élément de B soit atteignable de par cette application (i.e. nous construisons une application injective et surjective, donc bijective, de A à B).

Soit $a \in A$ fixé, et dans a considérons la première ")" qui rend le mot inacceptable. En remplaçant après cette ")" toutes les "(" par des ")" et toutes les ")" par des "(", on obtient un élément de B.

Cette application est bijective :

Vérifions que l'application est injective, i.e. que les different a's donnent des différents b : soient $a_1 \in A$ et $a_2 \in A$ avec $a_1 \neq a_2$. Il y a deux possibilites.

i) a_1 et a_2 sont égaux jusqu'au premier ")" qui les rend inacceptables et après ils sont differents. Dans ce cas ils restent different si après ce premier ")" on remplace tout les "(" par ")" et tout les ")" par "(".

[1]Rappelons que nous exigeons aussi pour un mot inacceptable qu'il y ait exactement n "(" et n ")".

ii) a_1 et a_2 sont different avant le premier ")" qui rend a_1 (ou a_2) inacceptable. Dans ce cas ils encore restent different après le remplacement parce qu'on ne change pas la structure des parenthèses qui sont avant le premier ")".

La preuve pour les éléments de B suit la meme idée, et nous l'omettons.

d) On sait qu'il y a en tout $\binom{2n}{n}$ mots, et le nombre de mots inacceptables est $\binom{2n}{n-1}$. Donc le nombre de mots acceptables sera.

$$\binom{2n}{n} - \binom{2n}{n-1} = \frac{(2n)! \cdot (n+1) - (2n)! \cdot n}{n! \cdot (n+1)!} = \frac{1}{n+1} \cdot \frac{(2n)!}{n! \cdot n!} = \frac{1}{n+1} \cdot \binom{2n}{n}$$

C'est égal à C_{n+1}.

e) Remarquer que chaque disposition de parenthèse correspond à un mot acceptable. Dans un produit parenthésé, effacer les éléments et les "(" et garder les ")" et les ".", ensuite remplacer tous les "." par "(". Par exemple :

$$\begin{array}{ccccc}
(a \cdot b) & = & \cdot) & = & () \\
((a \cdot b) \cdot c) & = & \cdot)\cdot) & = & ()() \\
(a \cdot (((b \cdot c) \cdot d) \cdot e)) & = & \cdot\cdot)\cdot)\cdot)) & = & (()()())
\end{array}$$

On remarque que le nombre de dispositions de parenthèses est exponentiel en n. Ce n'est pas donc une bonne idée de toutes les essayer pour trouver la plus efficace.