TP Programmation Dynamique
L'Énigme du Chameau dans le Désert

19 septembre 2006

Un chameau a besoin d'une banane pour avancer d'un kilomètre. Il peut porter jusqu'à *cap* ($cap \in \mathbb{N}^*$) bananes sur son dos. Il possède initialement un tas de n ($n \in \mathbb{N}$) bananes. On suppose qu'il progresse sur une droite. A quelle distance maximale de son point de départ peut-il aller ? Il s'agira ici de rédiger et d'implémenter un algorithme de programmation dynamique pour résoudre l'énigme. Les questions théoriques sont à répondre sur papier.

Exemple Supposons que $cap = 3$ et $n = 6$. La solution optimale est 4. Il charge 3 bananes, fait un kilomètre pour aller jusqu'à un point A, où il dépose une banane. Il lui reste alors une banane dont il se sert pour retourner au point de départ. Il charge les bananes restantes , soit 3 bananes, il va jusqu'au point A (en consommant une banane). Il charge la banane qui se trouve au point A, il a alors 3 bananes sur le dos, et fait ensuite trois kilomètres. Soit un total de quatre kilomètres.

1 Sauts élémentaires

On note $b[x, y]$ le nombre maximal de bananes qu'il peut avancer à une distance y (en faisant potentiellement des allers et retours – a/r –) , partant d'un tas de x bananes, sans faire de tas intermédiaires (on parlera alors de *saut élémentaire*).

Dans l'exemple, le trajet du chameau se décompose en deux sauts élémentaires, un premier de longueur 1 et un deuxième de longueur 3, et on a $b[6, 1] = 3$ et $b[3, 3] = 0$.

Une *stratégie de saut élémentaire* est donc une suite (possiblement vide) d'a/r suivi d'un aller.

Question 1 Avec $cap = 50$, montrer que $b[170, 10] \geq 100$ (donner deux stratégies différentes).

Question 2 : Conditions d'existence d'une stratégie Donner deux conditions nécessaires respectivement sur x et cap, en fonction de y, pour qu'une stratégie existe.

Question 3 *Dans la suite, on suppose qu'il existe une stratégie.*

Que vaut $b(x, y)$ si $x = y$? si $x < y$? Trouver des valeurs de cap, x, et y telles que le chameau ne puisse pas épuiser tout son stock de bananes x, i.e., après avoir amener $b[x, y]$ bananes, il reste encore des bananes au point de départ (on demande que la stratégie optimale choisie pour amener les bananes implique au moins un a/r, mais on ne demande pas de prouver que cette stratégie est optimale).

Question 4 : A/R ou simple Aller ? Donner deux conditions nécessaires (respectivement sur x et y) pour qu'un a/r soit possible (attention, ne pas oublier que le chameau doit faire un dernier aller).

Question 5 Déduire des questions précédentes une formule de récurrence pour $b[x, y]$. *Indication : il n'est pas forcément optimal de faire un a/r, même si cela est possible.* Implémenter un algorithme de programmation dynamique qui prend en entrée x, y, et cap, et qui retourne $b[x, y]$. Tester votre algorithme sur des exemples simples. Donner un argument simple montrant que $b[x, y] < x$, pour tout $y > 0$.

Question 6 : complexité Quelle est la complexité de votre algorithme ? Est-ce polynomial ?

2 Implémentation du solveur

Dans cette section il vous est demandé d'écrire et d'implémenter un algorithme qui résoud l'énigme.

ENTREE : x, y, cap

SORTIE : $d[x]$, éloignement maximal du point de départ

Question 6 La résolution se fait en décomposant le parcours en sauts élémentaires. Le chameau part d'un tas x, déplace le maximum de bananes à une certaine distance y du tas d'origine, pour créer un nouveau tas de taille $b[x, y]$, il recommence ensuite le processus à partir du nouveau tas. On admettra qu'il existe toujours une solution optimale qui utilise cette stratégie, i.e. qu'il existe toujours **au plus** deux tas de bananes potentiellement utiles, et que ces deux tas sont consécutifs. Pour trouver une solution optimale, il faut donc essayer toutes les suites de sauts élémentaires possibles et retourner une solution qui va maximiser la distance qui éloigne le chameau de son point de départ.

1. Montrer qu'il n'y a pas unicité de la solution optimale

2. Quelle est la taille maximale d'un saut élémentaire (à exprimer en fonction de x et cap) ?
3. Écrire une formule de récurrence pour $d[x]$, en fonction de x, et $b[.,.]$. Implémenter cette fonction par un algorithme de programmation dynamique. Tester l'algorithme (pour information, avec $cap = 500$, on doit obtenir $d[1500] = 766$).

Question 7 Modifier l'algorithme afin de pouvoir récupérer une stratégie optimale quelconque, et l'afficher. ("il prend 3 bananes, avance de 4, revient, reprend 3 bananes, ...")

Modifier l'algorithme afin de pouvoir récupérer la solution optimale qui fait les plus petits sauts élémentaires (i.e. on choisit celle qui fait le plus petit premier saut, et le plus petit deuxième saut sachant qu'elle a fait le plus petit premier saut, etc...). De même avec les plus grands sauts.

Question 8 Quelle est la complexité de votre algorithme ? Est-ce polynomial ?

3 Optimisation

Il est possible d'optimiser l'algorithme de calcul des sauts élémentaires en remarquant que le chameau a toujours intérêt à être chargé autant qu'il peut. Il s'agira ici de le prouver et de modifier l'algorithme.

Pour faciliter la preuve, nous allons formaliser la notion de stratégie de saut élémentaire. Une stratégie $S^{x,y}$ de saut élémentaire, pour x et y, est une suite finie $(a_1, \ldots, a_m, r)$ d'entiers strictement positifs telle que :

- $2y \leq a_i \leq cap$ pour tout $1 \leq i \leq m$
- $y \leq r \leq cap$
- $r + \Sigma_{i=1}^{m} a_i \leq x$

Les entiers a_1,...,a_m correspondent aux charges du chameau lors des a/r, l'entier r correspondant à la charge du dernier aller.

Par exemple, les stratégies qui consistent à charger le chameau maximalement sont de la forme $(cap, cap, \ldots, cap, r)$ où $0 < r \leq cap$.

Dans la suite, on se place dans les conditions d'existence d'une stratégie de saut élémentaire.

Question 9 Soit $S^{x,y} = (a_1, \ldots, a_m)$ une stratégie. On note $B(S^{x,y})$ le nombre de bananes déplacées en adoptant la stratégie $S^{x,y}$. Exprimer $B(S^{x,y})$ en fonction de x, y et m.

Soit $S^{x,y}_{max}$ la stratégie qui consiste à charger maximalement le chameau. Exprimer $B(S^{x,y}_{max})$ en fonction de x et y.

Question 10 Supposons que la stratégie $S^{x,y}_{max}$ fait k a/r. Montrer que $B(S^{x,y}) <= B(S^{x,y}_{max})$. *Indications : commencer par les cas $k > m$, et $k = m$. Pour le cas $k < m$, majorer le gain en bananes d'une stratégie quelconque par rapport* *à $S_{max}^{x,y}$, i.e. le nombre de bananes restantes au point de départ, après application de la stratégie $S_{max}^{x,y}$, comparer ce gain par rapport au coût d'un a/r supplémentaire, conclure.*

Question 11 Exprimer alors $b[x, y]$ en fonction de x et y. Modifier l'algorithme du calcul de $b[x, y]$. Quelle est sa complexité ? Est-ce polynomial ?

4 Rendement

Question 12 Le rendement est le rapport $d(x)/x$. Ecrire un algorithme qui prend en entrée une borne X, et retourne l'ensemble $\{d(x)/x \mid 1 \leq x \leq X\}$ en temps $O(X * cap)$ (Facultatif : afficher le résultat par une courbe à l'aide de l'outil `gnuplot`). Que remarquez-vous ?

5 Questions Bonus

Etudier la forme de la solution qui fait successivement les plus petits sauts élémentaires (cf Question 7). Discuter.

Étendre l'algorithme avec une consommation variable : le chameau consomme un nombre de bananes proportionnel à son chargement. Formaliser cette notion et étendre l'algorithme.

Utiliser votre algorithme pour résoudre une variante de cette énigme posée par Einstein (2ème énigme d'Einstein) : un chameau doit parcourir 1000 kms de désert, ne peut porter que 1000 bananes à la fois, il consomme 1 banane/km, et possède initialement 3000 bananes, combien lui en restera-t-il au maximum à la fin ?