École polytechnique — Majeure informatique

Promotion 2007

Cours de *conception et analyse d'algorithmes*

TD 7 – Algorithmes probabilistes

3 novembre 2010

1. Programmation linéaire, algorithmes probabilistes

On considère le problème SET-COVER pour lequel la donnée est un ensemble $E = \{1, 2, \ldots, n\}$ et une famille $\mathcal{F}$ de sous-ensembles $F_1, F_2, \ldots, F_m$ de E dont l'union est égale à E. On suppose de plus que chaque élément $j \in E$ est contenu dans exactement k sous-ensembles F_i. Il s'agit de trouver un nombre minimum de sous-ensembles $F_i, i \in I$ tels que $\bigcup_{i\in I} F_i = E$.

On se propose ici d'utiliser la programmation linéaire et un algorithme probabiliste pour résoudre ce problème. Pour cela on associe à chaque F_i une variable x_i qui prendra des valeurs comprises entre 0 et 1.

i. Exprimer le problème sous forme d'un programme linéaire en nombres entiers comportants n contraintes (en plus de $0 \leq x_i \leq 1$) et une fonction à minimiser. Ecrire ces contraintes pour l'exemple suivant $E = \{1, 2, 3, 4, 5\}$ et $F_1 = \{1, 3, 5\}$, $F_2 = \{2, 3, 4\}$, $F_3 = \{1, 2\}$, $F_4 = \{4, 5\}$, $F_5 = \{1, 2, 4\}$, $F_6 = \{3, 5\}$.

ii. Soit $\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_m$ la solution au problème lorsqu'on relache la condition nombres entiers pour autoriser des rationnels. Donner une borne inférieure pour l'optimum du cas entier en fonction de la solution $\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_m$.

iii. On applique la méthode suivante pour résoudre notre problème à partir de la solution en nombres rationnels $\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_m$: choisir si $i \in I$ avec probabilité $\hat{x}_i$. Cette méthode est-elle de type Monte-Carlo ou Las Vegas ?

Quelle est en fonction des $\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_m$ la probabilité pour qu'un élément j donné appartenant à $F_{j_1}, F_{j_2}, \ldots, F_{j_k}$ n'appartienne à aucun des F_i pour $i \in I$?

iv. Montrer que la probabilité que j soit dans un des $F_i, i \in I$ est minorée par $1 - \frac{1}{e}$.

v. Donner un algorithme qui après avoir résolu le problème linéaire en nombres rationnels effectue $m \log(n)$ tirages aléatoires pour donner une solution approchée à un facteur $O(\log n)$ et ceci avec probabilité supérieure ou égale à $\frac{1}{2}$.

2. Un algorithme probabiliste pour 2-SAT

Une formule 2-SAT consiste en p clauses $u_i = y_i \vee z_i$ sur n variables. On considère l'algorithme probabiliste suivant pour chercher une assignation satisaisant la formule (m étant un paramètre) :

– partir d'une affectation quelconque de valeurs aux variables $x = (x_1, \ldots, x_n)$.
– pour i de 1 à m faire
    s'il existe une clause c non satisfaite,
      changer la valeur d'une des deux variables choisie au hasard
    sinon renvoyer "satisfiable"

– renvoyer “non satisfiable”

i. L’algorithme est-il de type Monte Carlo ou Las Vegas ? Quelle est sa complexité. Préciser. Quelle est la probabilité d’erreur si la formule est non satisfiable ?

On suppose qu’il existe une affectation u satisfaisant toutes les clauses et on oublie momentanément la condition d’arrêt.

ii. Soit $E(x)$ le nombre moyen d’itérations avant de renvoyer “satisfiable” en partant d’une assignation x, et soit $E(i)$ le maximum des $E(x)$ sur toutes les assignations x telles que la distance de Hamming $d(x,u)$ soit inférieure ou égale à i. Montrer que :

$$\begin{cases} E(0) & = & 0 \\ E(n) & \leq & 1+E(n-1) \\ E(i) & \leq & 1+(E(i-1)+E(i+1))/2 \text{ pour } 1 \leq i \leq n-1. \end{cases}$$

iii. En déduire successivement que $E(i) \leq 2(n-i+1)-1+E(i-1)$ pour tout $i \in \{1\ldots,n-1\}$, puis une borne supérieure sur $E(n)$.

iv. Par l’inégalité de Markov, en déduire que si $m > 2n^2$, alors la probabilité d’obtenir l’affectation correcte est au moins égale à $1/2$.

v. Que penser d’une généralisation à 3-SAT ?

3. Coupure minimale

Soit G un graphe connexe non orienté, dont les arêtes peuvent être multiples, avec n sommets. Une *coupure* dans G est un ensemble d’arêtes dont la suppression rend G non connexe. Une coupure minimale est une coupure de cardinalité minimale. Voici un algorithme probabiliste pour la détermination d’une coupure minimale.

Contracter une arête $e=\{u,v\}$ signifie : fusionner ces deux sommets en un seul sommet w ; les arêtes incidentes à w sont les arêtes qui étaient précédemment incidentes à u ou à v ; puis, supprimer les boucles (induites par les arêtes reliants u et v).

L’algorithme contracte une suite d’arêtes choisies au hasard, jusqu’à ce que le graphe n’ait plus que deux sommets ; il renvoie alors comme valeur de coupure le nombre d’arêtes entre ces deux sommets.

i. Montrer que l’opération de contraction ne diminue pas la valeur de la coupe minimale.

ii. Soit k la taille de la coupe minimale. Montrer que G a au moins $kn/2$ arêtes.

iii. Quelle est la probabilité qu’après ℓ étapes, aucune des arêtes d’une coupe minimale C de taille k n’ait été contractée ? En déduire une borne inférieure sur la probabilité que l’algorithme renvoie la valeur d’une coupe minimale.

iv. Quelle est la complexité de l’algorithme ? Construire un algorithme polynomial renvoyant la bonne valeur avec probabilité $1-O(1/n^2)$.