Exercices sur les algorithmes probabilistes

Florian Bourse

Arbres ternaires complet

On considère un arbre ternaire dont chaque nœud interne possède exactement 3 fils et toutes les feuilles sont à profondeur h. On associe à chaque feuille une valeur booléenne. La valeur d'un nœud interne est la majorité des valeurs de ses fils. On cherche à évaluer la valeur de la racine.

Question 1. Montrer que pour tout algorithme déterministe $\mathcal{A}$, il existe une instance du problème pour laquelle $\mathcal{A}$ devra lire les valeurs de toutes les $n = 3^h$ feuilles pour évaluer correctement la valeur de la racine.

Question 2. On considère l'algorithme récursif qui consiste à tirer aléatoirement deux sous-arbres et n'évaluer le troisième que si les deux premiers sont en désaccord. Montrer qu'en espérance, le nombre de feuilles dont l'algorithme lit la valeur est $O(n^{0.9})$.

Trouver le billet

On considère le problème suivant : parmi n boîtes, numérotées de 1 à n, une et une seule contient un billet, les autres étant vides. Le but est de trouver le billet en ouvrant le moins de boîte possible.

On considère l'algorithme suivante :

— Tirer à pile ou face :
Pile : ouvrir les boîtes dans l'ordre croissant ;
Face : ouvrir les boîtes dans l'ordre décroissant.

Question 3. Montrer qu'en espérance, cet algorithme ouvre $\frac{n+1}{2}$ boîtes.

Question 4. Montrer que cette solution est optimale en espérance.

Arbres binaires de recherche

On considère l'opération suivante : étant donnés n éléments distincts et comparables, les insérer dans un arbre binaire de recherche (ABR).

Question 5. Quelle est la complexité de la recherche d'un élément dans l'ABR résultant de cette opération au pire cas en fonction de n ? Et au meilleur cas ? Donner un ordre d'insertion lorsque les éléments sont ceux de $[\![1, 7]\!]$ pour lequel le pire cas est atteint et un ordre pour lequel le meilleur cas est atteint.

Afin d'éviter le pire cas, on insère les n éléments en les choisissant dans un ordre aléatoire : on considère que les $n!$ permutations possibles des éléments sont des ordres d'insertion équiprobables. On note H_n la variable aléatoire correspondant à la hauteur de l'arbre obtenu, $X_n = 2^{H_n}$, R_n le rang de la racine de l'arbre parmi les n éléments (c'est-à-dire sa position si les n éléments étaient triés) et $Z_{n,i}$ la variable aléatoire valant 1 si $R_n = i$ et 0 sinon.

Question 6. Montrer que, si $R_n = i$ alors $X_n = 2\max(X_{i-1}, X_{n-i})$.

Question 7. Montrer que $X_n = \displaystyle\sum_{i=1}^{n} Z_{n,i} \times 2\max(X_{i-1}, X_{n-i})$.

Question 8. En observant que $Z_{n,i}$ est indépendante de X_{i-1} et X_{n-i}, montrer que $\mathbb{E}[X_n] \leq \frac{4}{n} \displaystyle\sum_{k=0}^{n-1} \mathbb{E}[X_k]$.

On admet qu'une telle relation de récurrence implique que $\mathbb{E}[X_n] \leq \frac{1}{4}\binom{n+3}{n}$.

Question 9. L'inégalité de Jensen garantit que si f est une fonction convexe sur un intervalle I et X est une variable aléatoire sur I alors $f(\mathbb{E}[X]) \leq \mathbb{E}[f(X)]$. Déduire de ce qui précède que $\mathbb{E}[H_n] \leq O(\log n)$ et conclure.