Comparaison tri fusion/tri rapide

Pierron Théo

13 avril 2014

On étudie ici deux tris utilisant le paradigme diviser pour régner.

Tri fusion

Le tri fusion se déroule en trois étapes :

- on sépare le tableau au milieu
- on trie récursivement les deux tableaux
- on fusionne les deux tableaux. On compare les deux minima, on met le plus petit au premier emplacement du résultat, et on réitère.

La première phase est en $O(1)$ et la troisième nécessite n comparaisons (une pour chaque élément à placer). La complexité $T(n)$ du tri d'un tableau de taille n vérifie donc

$$T(n) = T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + T\left(\left\lceil \frac{n}{2} \right\rceil\right) + n$$

Le théorème marteau pilon sur les complexités des algorithmes diviser pour régner permet alors de conclure : $T(n) = O(n\ln(n))$. On expose ici une méthode moins brutale.

Étudions la complexité du tri d'un tableau de taille 2^k. On a

$$T(2^k) = 2T(2^{k-1}) + 2^k \text{ donc } \frac{T(2^k)}{2^k} = \frac{T(2^{k-1})}{2^{k-1}} + 1$$

Autrement dit, cette dernière suite est arithmétique, et, comme $T(1) = 0$, on a $T(2^k) = k2^k$.

Si n est quelconque, il existe k tel que $n \leqslant 2^k < 2n$. Alors, comme la complexité est croissante, on a

$$T(n) \leqslant k2^k \leqslant 2kn = 2n\lceil \log_2(n) \rceil$$

Donc $T(n) = O(n\ln(n))$.

Tri rapide

Le tri rapide se déroule aussi en trois étapes :

- on choisit un pivot
- on sépare le tableau en deux tableaux contenant respectivement les éléments plus grands et plus petits que x, ce qui se fait en $n-1$ comparaisons
- on trie récursivement ces deux tableaux

On note que le tri rapide peut s'exécuter en place.

Le pire cas est obtenu en choisissant les pivots tels qu'à chaque séparation, l'un des deux tableaux obtenus soit de taille 0. Alors la complexité $C_p(n)$ du tri rapide d'un tableau de taille n dans le pire cas vérifie :

$$C_p(n) = n - 1 + C_p(n-1) = O(n^2)$$

On va ici étudier la complexité moyenne $C(n)$ quand on prend des pivots aléatoires uniformes, c'est-à-dire que la taille du tableau contenant les éléments plus petits que le pivot suit une loi uniforme sur $[\![0, n-1]\!]$. On a donc

$$C(n) = n - 1 + \frac{1}{n}\sum_{k=0}^{n-1}(C(k) + C(n-1-k)) = n - 1 + \frac{2}{n}\sum_{k=0}^{n-1}C(k)$$

On a

$$(n+1)C(n+1) - nC(n) = n(n+1) + 2\sum_{k=0}^{n}C(k) - n(n-1) + 2\sum_{k=0}^{n-1}C(k) = 2n + 2C(n)$$

Donc

$$\frac{C(n+1)}{n+2} - \frac{C(n)}{n+1} = \frac{2n}{(n+1)(n+2)}$$

et finalement

$$\frac{C(n)}{n+1} = 2\sum_{k=0}^{n-1}\frac{k}{(k+1)(k+2)} \leqslant 2\sum_{k=0}^{n-1}\frac{1}{k+1} \leqslant 2\ln(n)$$

Donc $C(n) \leqslant 2\ln(2)(n+1)\log_2(n)$ donc $C(n) = O(n \log n)$.

On a donc deux tris, l'un ayant une complexité temporelle optimale dans tous les cas, l'autre ayant une complexité optimale en moyenne, mais étant en place.