Algorithmique et Programmation

Analyse des arbres splay

Exercice 4. Analyse des arbres splay

1. Soit T l'arbre avant la (double) rotation et T' l'arbre après.

   (a) Seul la taille de $|T(x)|$ et $|T(y)|$ changent. De plus, $\text{Rang}'(x) = \text{Rang}(y)$ (car la taille de leurs sous-arbres sont égaux) donc la différence de rangs est de

   $$\text{Rang}'(y) - \text{Rang}(x) \leq \text{Rang}'(x) - \text{Rang}(x)$$

   (b) Seul la taille de $|T(x)|$, $|T(y)|$ et $|T(z)|$ changent. De plus, $\text{Rang}'(x) = \text{Rang}(z)$ (car la taille de leurs sous-arbres sont égaux), donc la différence de rangs est de

   $$\begin{aligned}
   &\text{Rang}'(y) + \text{Rang}'(z) - \text{Rang}(y) - \text{Rang}(x)\\
   &= (\text{Rang}'(y) - \text{Rang}(x)) + (\text{Rang}'(z) - \text{Rang}(y))\\
   &\leq 2(\text{Rang}'(x) - \text{Rang}(x)) + (\text{Rang}'(z) - \text{Rang}'(x)) + (\text{Rang}'(y) - \text{Rang}'(x))
   \end{aligned}$$

   À noter que les deux derniers termes sont négatifs et il suffit de montrer que l'un d'entre eux est au plus -1.

   Puisque $T'(x)$ contient les sous arbre A, B, C, D en plus de x, y, z, nous avons $|T'(x)| = |T'(z)| + |T'(y)| + 1$. L'un des deux termes est inférieur à la moitié de $|T'(x)|$. Puisque nous prenons le log en base 2, nous avons soit $\text{Rang}'(z) \leq \text{Rang}'(x) - 1$ ou $\text{Rang}'(y) \leq \text{Rang}'(x) - 1$ (ou les deux).

   Dans chacun des deux cas, nous avons une borne supérieur de -1 sur un des deux derniers terme et donc une borne de $2(\text{Rang}'(x) - \text{Rang}(x)) - 1$ sur la différence.

   (c) Seul la taille de $|T(x)|$, $|T(y)|$ et $|T(z)|$ changent. De plus, $\text{Rang}'(x) = \text{Rang}(z)$ (car la taille de leurs sous-arbres sont égaux), donc la différence de rangs est de

   $$\begin{aligned}
   &\text{Rang}'(y) + \text{Rang}'(z) - \text{Rang}(y) - \text{Rang}(x)\\
   &= (\text{Rang}'(y) - \text{Rang}(x)) + (\text{Rang}'(z) - \text{Rang}(y))\\
   &\leq 2(\text{Rang}'(x) - \text{Rang}(x)) + (\text{Rang}'(z) - \text{Rang}'(x))\\
   &\leq 3(\text{Rang}'(x) - \text{Rang}(x)) + (\text{Rang}'(z) - \text{Rang}'(x)) + (\text{Rang}(x) - \text{Rang}'(x))
   \end{aligned}$$

   À noter que le derniers termes sont négatifs et il suffit de montrer qu'il est au plus -1.

   Puisque $T'(x)$ contient les sous arbre A, B, C, D en plus de x, y, z, nous avons $|T'(x)| = |T(x)| + |T'(z)| + 1$. L'un des deux termes est inférieur à la moitié de $|T'(x)|$. Puisque nous prenons le log en base 2, nous avons soit $\text{Rang}(x) \leq \text{Rang}'(x) - 1$ ou $\text{Rang}'(z) \leq \text{Rang}'(x) - 1$ (ou les deux).

   Dans chacun des deux cas, nous avons une borne supérieur de -1 sur un des deux derniers terme et donc une borne de $3(\text{Rang}'(x) - \text{Rang}(x)) - 1$ sur la différence.

2. Nous allons utiliser une fonction potentiel $\phi(T) = C_1 \sum_{v\in T} \lceil \log(|T(v)|)$ et montrer que pour chaque opération, la somme du temps pris et la différence de potentiel est au plus $C_2 \log n$ pour des constantes C_1, C_2 (pour lesquels nous pouvons choisir les valeurs plus tard).

   Puisque les trois opérations appliquent Splay à la fin de l'opération, nous allons d'abord analyser le temps de calcul et la diffénce de potentiel pour une application de Splay(T, x) lorsque x est profondeur h.

   Intuitivement, les trois opérations vont prendre temps $O(h)$ et la fonction potentiel augmente d'au plus $O(h)$, donc il faut que pour Splay, le temps de calcul soit $O(\log(n))$ et que la différence de potentiel *diminue* de $O(h)$. Nous vérifions que c'est bien le cas.

   Étant donné qu'une rotation (simple ou double) consiste à changer un nombre constant (au plus 7) de valeurs (les fils gauches et droites de x, y, z et le parent de z), son temps de calcul est d'au plus $7h$.

   Regardons maintenant la différence de potentiel. On note $\textsc{Rang}_i(x)$ la valeur de $\lceil \log(|T(x)|)$ après la $i$ème rotation (simple ou double) et d_i la différence de potentiel pour la $i$ème rotation. Soit k le nombre de rotations éffectués ($k = h/2$ si h est pair et $k = (h+1)/2$ si h est impair). La différence de potentiel pour Splay(T, x) est donc

$$
\begin{aligned}
\sum_{i}^{k} d_i &\leq C_1 \cdot (3(\textsc{Rang}_k(x) - \textsc{Rang}_{k-1}(x)) + \sum_{i=0}^{k-1} 3(\textsc{Rang}_{i+1)(x) - \textsc{Rang}_i(x))-1)} \\
&= C_1 \cdot (3(\textsc{Rang}_k(x) - \textsc{Rang}_0(x)) - (k-1)) \\
&\leq C_1 \cdot (3\log(n) - (h/2) + 1)
\end{aligned}
$$

   La deuxième inégalité vient du fait que la somme est téléscopique (les termes d'éléments successifs s'annulent). Cette borne de $C_1 \cdot (3\log(n) - (h/2) + 1)$ sur la différence de potentiel suffit pour le reste de la preuve.

   Nous pouvons maintenant nous tourner vers le temps de calcul et différences de potentiel pour les trois opérations insertion, suppression et recherche (hors l'opération Splay appliqué à la fin). Pour les trois, le temps de calcul est borné par $c_3 h$ pour une constante c_3 (et où h est la hauteur de l<élément sur lequel Splay sera appliqué). Et pour les trois, la différence de potentiel est majoré par $c_4 h$ pour une constante c_4 (puisqu'au plus un noeud dans l'arbre change).

   Donc, en choisissant $C_1 \geq 2(c_3 + c_4)$ (par ex, $C_1 = 2c_3 + c_4$), nous avons bien un coût total de $C_1 \cdot (3\log(n) - (h/2) + 1) + c_3 h + c_4 h \leq 3C_1 \log(n) + C_1 = O(\log n)$ par opération.