Preuve du Master Theorem

Inès Dussuet, Adrien Bazoge, Maëlle Brassier

16 décembre 2018

1 Introduction

En informatique, il existe différentes techniques algorithmiques. L'une d'entre elle, **diviser pour régner**, se base sur le principe de récursivité. Elle se décompose en plusieurs étapes : diviser de façon récursive le problème initial en sous-problèmes de même type jusqu'à ce qu'ils soient assez faciles à résoudre directement. Les solutions de ces sous-problèmes sont ensuite combinées afin de calculer une solution au problème initial. Nous nous intéressons ici au **Master Theorem**, utilisé en analyse de complexité des algorithmes, qui produit une analyse asymptotique pour les relations réccurentes présentes dans les algorithmes diviser pour régner.

2 La Preuve

Comme nous l'avons décrit précédement, les algorithmes diviser pour régner résolvent un nombre $\boldsymbol{a}$ sous-problèmes de taille $\boldsymbol{n/b}$ et combinent leurs réponses en un temps $\mathcal{O}(n^d)$, avec a,b,d > 0.

Le temps d'exécution d'un algorithme récursif se définit comme :

$$\mathrm{T(n)} \leq a{*}T(n/b){+}\mathrm{O(n}\ ^{d})$$

avec la taille des données **n** divisée par une constante **b** et l'appel récursif appelé un nombre constant **a**. Le terme général se décompose en trois cas :

$$\begin{cases} \mathrm{O(n}^d), & \text{si } d > log_b a \qquad (1) \\ \mathrm{O(n}^d logn), & \text{si } d = log_b a \qquad (2) \\ \mathrm{O(n}^{log_b a}), & \text{si } d < log_b a \qquad (3) \end{cases}$$

Nous allons appliquer notre récurrence en crééant un arbre. Cet arbre comportera un nombre **a** de branches et un nombre a^i de sommets au niveau i de la récurrence. Ces derniers seront labelisés par T(n/b^i). La hauteur de l'arbre sera de $\log_b n$ jusqu'à ce que les sommets soient de valeur T(1) et donc de niveau $a^{\log_b n}$ (1). La largeur quant à elle se définit comme $a^{\log_b n}$ ce qui est équivalent à $n^{\log_b a}$. À partir de ces observations, on peut déduire que T(n) correspond à la somme des valeurs de tous les noeuds de l'arbre.

La solution de la récurrence est donc :

$$T(n) = \sum_{i=0}^{\log_b n} a^i O(n/b^i)^d$$

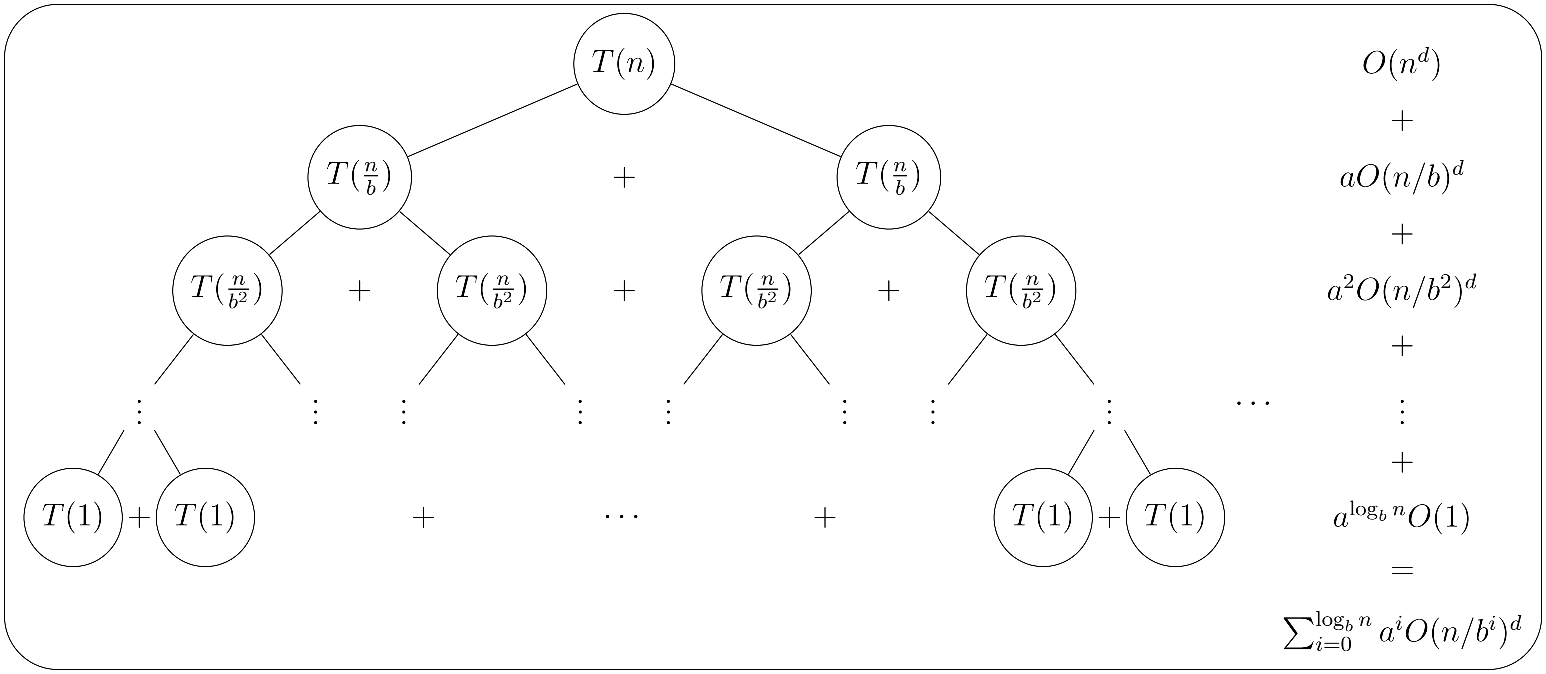

Figure 1. *Arbre de récurrence avec $a = 2$*

On a donc :

$$\begin{aligned}
T(n) &= 1 \times O(n)^d + aO\left(\frac{n}{b}\right)^d + a^2 O\left(\frac{n}{b^2}\right)^d + ... + a^k O\left(\frac{n}{b^k}\right)^d \\
&= O(n)^d \left[1 + a\left(\frac{1}{b}\right)^d + a^2\left(\frac{1}{b^2}\right)^d + ... + a^k\left(\frac{1}{b^k}\right)^d\right] \\
&= O(n^d) \underbrace{\left[1 + \left(\frac{a}{b^d}\right) + \left(\frac{a}{b^d}\right)^2 + ... + \left(\frac{a}{b^d}\right)^k\right]}_{\text{Suite géométrique de raison } \frac{a}{b^d}}
\end{aligned}$$

Suite géométrique s, pour $r \neq 1$:

$$s = y + yr + yr^2 + yr^3 + ... + yr^{n-1} = \sum_{k=0}^{n-1} yr^k = y\left(\frac{1 - r^n}{1 - r}\right)$$

Dans notre cas, $y = 1$ donc $s = \frac{1-r^n}{1-r}$

Cas 1 :

Si $a < b^d$, alors $r = \frac{a}{b^d} < 1$
$r < 1 \Rightarrow$ la suite est décroissante
donc la somme des termes est du même ordre de grandeur que le premier terme, soit O(1).

$$T(n) = O(n^d \times 1)$$
$$= O(n^d)$$

Cas 2 :

Si $a = b^d$, alors $r = 1$,
$r = 1 \Rightarrow$ tous les termes de la suite sont égaux à 1.
Tous les termes $= k + 1$ termes

$$T(n) = O(n^d(k+1))$$
$$= O(n^d\, k)$$

On sait que $n = b^k$, donc :

$$k = log_b\, n$$
$$k = O(log\, n)$$

$$T(n) = O(n^d\, k)$$
$$= O(n^d\, log\, n)$$

Cas 3 :

Si $a > b^d$, alors $r > 1$
$r > 1 \Rightarrow$ la suite est croissante
donc la somme des termes est du même ordore de grandeur que le dernier terme, soit $O(\left(\frac{a}{b^d}\right)^k)$

$$T(n) = O(n^d\ \left(\frac{a}{b^d}\right)^k)$$
$$= O(n^d\, a^k\ \left(\frac{1}{b^d}\right)^k)$$

$a^k > n^d$ et $a^k > \left(\frac{1}{b^d}\right)^k$ donc :

$$T(n) = O(a^k)$$

Or, on sait que $k = log_b\, n$:

$$T(n) = O(a^{log_b\, n})$$
$$= O(n^{log_b\, a})$$

3 Conclusion

Dans ce rapport, nous avons cherché à démontrer la preuve du Master Theorem. Nous avons tout d'abord établi son énoncé, à savoir sa récurrence et ses différents cas. Puis, en effectuant plusieurs récursions, nous avons obtenu son arbre de récurrence qui nous a permis par la suite de retrouver une suite géométrique. En utilisant les propriétés de ces suites, notamment la raison r, nous avons pu faire apparaître les 3 cas et ainsi établir la preuve du Master Theorem.