Algorithmique 1
Récurrence

Hajer Akid & Sandrine Julia

Semestre 4

Complexité d'un algortihme récursif ?

Relation de récurrence

$$T(n) = \sum_{i=1}^{k} T(n_i) + f(n)$$

avec

- n : taille en entrée
- k : nombre d'appels récursifs
- n_i : taille en entrée de l'appel récursif
- $f(n)$: coût en dehors des appels récursifs

Relation de récurrence

Exemple (Multiplication de 2 entiers)

Soient x et y deux entiers non négatifs de n chiffres, donner un algorithme pour calculer $x \cdot y$

```
recMult(x, y, n)
  si (n = 1) alors                              O(1)
    retourner x*y                               O(1)
  sinon
    m ← n/2                                     O(1)
    p ← 10^m                                    O(1)
    a ← x/p                                     O(1)
    b ← x % p                                   O(1)                  = O(n)
    c ← y/p                                     O(1)
    d ← y % p                                   O(1)   = O(n)
    ac ← recMult(a, c, m)
    ad ← recMult(a, d, m)
    bc ← recMult(b, c, m)
    bd ← recMult(b, d, m)
    retourner 10^n * ac + p * (ad + bc) + bd    O(n)
```

$T(n) = 4T\left(\frac{n}{2}\right) + n$

De la relation de récurrence vers la complexité

Méthode de substitution

- Marche toujours
- Approche mathématiques
- Demande de l'intuition
- Réécrire $T(n) \leq \sum_i c_i f_i(n) - p_r$

  avec p_r la partie résiduelle qui doit être positive

Méthode de substitution

Exemple ($T(n) = 4T\left(\frac{n}{2}\right) + n \in O(n^3)$)

$$\begin{aligned}
T(1) &= 1\\
T(k) &\leq ck^3\\
T(n) &= 4T\left(\tfrac{n}{2}\right) + n\\
T(n) &\leq 4c\left(\tfrac{n}{2}\right)^3 + n\\
&= c\tfrac{n^3}{2} + n\\
&= cn^3 - \left(c\tfrac{n^3}{2} - n\right)\\
T(n) &\leq cn^3 \text{ si } c\tfrac{n^3}{2} - n \geq 0, \forall c \geq 2
\end{aligned}$$

On a bien $T(n) \in O(n^3)$

Méthode de substitution

Exemple ($T(n) = 4T\left(\frac{n}{2}\right) + n \in O(n^2)$)

$$\begin{aligned}
T(1) &= 1\\
T(k) &\leq ck^2\\
T(n) &= 4T\left(\tfrac{n}{2}\right) + n\\
T(n) &\leq 4c\left(\tfrac{n}{2}\right)^2 + n\\
&= cn^2 + n\\
&= cn^2 - (-n)
\end{aligned}$$

Ne fonctionne pas car $-n$ n'est pas positif

Méthode de substitution

Exemple ($T(n) = 4T\left(\frac{n}{2}\right) + n \in O(n^2)$)

$$\begin{aligned}
T(1) &= 1\\
T(k) &\leq c_1k^2 - c_2k\\
T(n) &= 4T\left(\tfrac{n}{2}\right) + n\\
T(n) &\leq 4\left(c_1\left(\tfrac{n}{2}\right)^2 - c_2\tfrac{n}{2}\right) + n\\
&= c_1n^2 - 2c_2n + n\\
&= c_1n^2 - c_2n - (c_2n - n)\\
T(n) &\leq c_1n^2 - c_2n - (c_2n - n) \text{ si } c_2n - n \geq 0, \forall c_2 \geq 1
\end{aligned}$$

On a bien $T(n) \in O(n^2)$

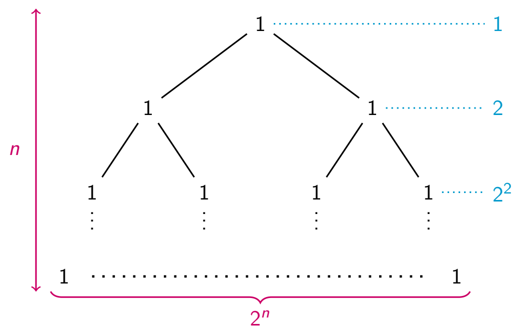
De la relation de récurrence vers la complexité

Méthode de l'arbre de récursion

- Quand on n'a pas l'intuition
- Dessin de l'arbre de recursion
- Calcul du coût par niveau

Méthode de l'arbre de récursion

Exemple ($T(n) = 2T(n-1) + 1$)

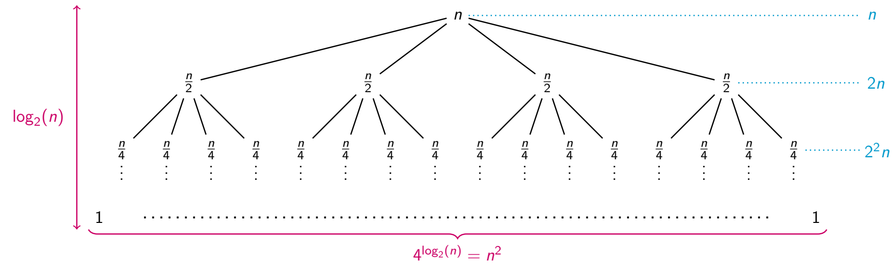


$$T(n) = \left(\sum_{i=0}^{n-1} 2^i\right) + 2^n = 2^n - 1 + 2^n = 2^{n+1} - 1$$

$T(n) \in O(2^n)$

Méthode de l'arbre de récursion

Exemple ($T(n) = 4T\left(\frac{n}{2}\right) + n$)

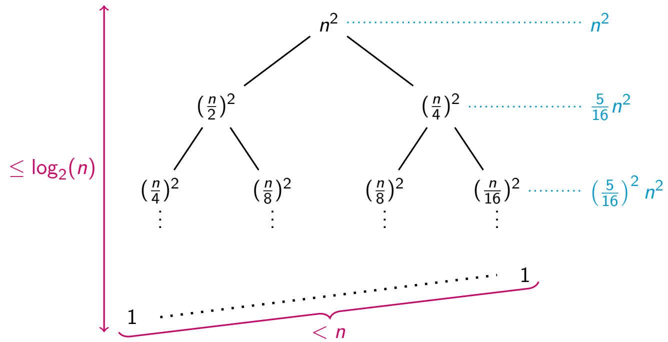


$$T(n) = \left(\sum_{i=0}^{\log_2(n)-1} 2^i n\right) + n^2 = n\left(\sum_{i=0}^{\log_2(n)-1} 2^i\right) + n^2$$
$$= n(2^{\log_2(n)} - 1) + n^2$$
$$= n(n-1) + n^2$$
$$= 2n^2 - n$$

$T(n) \in O(n^2)$

Méthode de l'arbre de récursion

Exemple ($T(n) = T\left(\frac{n}{2}\right) + T\left(\frac{n}{4}\right) + n^2$)



$$T(n) < \left(\sum_{i=0}^{\log_2(n)-1} \left(\frac{5}{16}\right)^i n^2\right) < n^2\left(\sum_{i=0}^{\infty} \left(\frac{5}{16}\right)^i\right) = \frac{16}{11}n^2$$

$T(n) \in O(n^2)$

De la relation de récurrence vers la complexité

Méthode 1

Uniquement quand la relation de récurrence est de la forme $T(n) = aT(n-b) + f(n)$

- a, b des constantes positives $a, b > 0$
- $f(n) \in O(n^c \log_2^d(n))$ avec $c, d \geq 0$

Deux cas :

1. $a = 1$: $T(n) \in O(nf(n))$
2. $a > 1$ et $d = 0$: $T(n) \in O(a^{\frac{n}{b}} f(n))$

Méthode 1

Exemple ($T(n) = 2T(n-1) + 1$)

- $a = 2, b = 1, c = 0, d = 0$
- $a > 1, d = 0 \Rightarrow$ cas 2

$T(n) \in O(2^n)$

Exemple ($T(n) = T(n-1) + n$)

- $a = 1, b = 1, c = 1, d = 0$
- $a = 1 \Rightarrow$ cas 1

$T(n) \in O(n^2)$

De la relation de récurrence vers la complexité

Méthode Master – 1989

Uniquement quand la relation de récurrence est de la forme $T(n) = aT(\frac{n}{b}) + f(n)$

- a une constante $a \geq 1$
- b une constante $b > 1$
- $f(n)$ une fonction asymptotiquement positive

Trois cas :

1. $f(n) \in O(n^c)$ avec $b^c < a$: $T(n) \in O(n^{\log_b(a)})$
2. $f(n) \in O(n^c \log_2^d(n))$ avec $d \geq 0$ une constante et $b^c = a$: $T(n) \in O(n^c \log_2^{d+1}(n))$
3. $f(n) \in O(n^c)$ avec $b^c > a$: $T(n) \in O(f(n))$

Méthode Master

Exemple ($T(n) = 4T\left(\frac{n}{2}\right) + n$)

- $a = 4, b = 2, c = 1$
- $b^c = 2^1 = 2$
- $b^c < a \Rightarrow$ cas 1

$T(n) \in O(n^2)$

Exemple ($T(n) = 3T\left(\frac{n}{2}\right) + n$ – Karatsuba)

- $a = 3, b = 2, c = 1$
- $b^c = 2^1 = 2$
- $b^c < a \Rightarrow$ cas 1

$T(n) \in O(n^{\log_2(3)})$

Méthode Master

Exemple ($T(n) = 2T\left(\frac{n}{2}\right) + n$ – Tri fusion)

- $a = 2, b = 2, c = 1$
- $b^c = 2^1 = 2$
- $b^c = a \Rightarrow$ cas 2

$T(n) \in O(n \log_2(n))$

Exemple ($T(n) = 2T\left(\frac{n}{2}\right) + n^2$)

- $a = 2, b = 2, c = 2$
- $b^c = 2^2 = 4$
- $b^c > a \Rightarrow$ cas 3

$T(n) \in O(n^2)$

De la relation de récurrence vers la complexité

Méthode Akra-Bazzi – 1998

Généralisation de la méthode Master pour les relations de récurrence de la forme $T(n) = \sum_{i=1}^{k} T(b_i * n) + f(n)$

- b_i une constante positive $b_i < 1$
- k une constante $k \geq 1$
- $f(n) \in O(n^c \log_2^d(n))$ avec $c, d \geq 0$

On pose $e = \sum_{i=1}^{k} b_i^c$

Trois cas :

1. $e < 1$: $T(n) \in O(n^c \log_2^d(n))$
2. $e = 1$: $T(n) \in O(n^c \log_2^{d+1}(n))$
3. $e > 1$: $T(n) \in O(n^x)$ avec x l'unique solution de $\sum_{i=1}^{k} b_i^x = 1$

Méthode Akra-Bazzi

Exemple $\left(T(n) = 4T\left(\frac{n}{2}\right) + n\right)$

- $k = 4, b_i = \frac{1}{2}, c = 1, d = 0$
- $e = \sum_{i=1}^{k} b_i^c = \sum_{i=1}^{4} \frac{1}{2} = 2 > 1 \Rightarrow$ cas 3
- $\sum_{i=1}^{4} \left(\frac{1}{2}\right)^x = 1 \iff \left(\frac{1}{2}\right)^x = \frac{1}{4}, x = 2$

$T(n) \in O(n^2)$

Exemple $\left(T(n) = T\left(\frac{n}{2}\right) + T\left(\frac{n}{4}\right) + n^2\right)$

- $k = 2, b_1 = \frac{1}{2}, b_2 = \frac{1}{4}, c = 2, d = 0$
- $e = \sum_{i=1}^{k} b_i^c = \left(\frac{1}{2}\right)^2 + \left(\frac{1}{4}\right)^2 = \frac{5}{16} < 1 \Rightarrow$ cas 1

$T(n) \in O(n^2)$