Chapitre 3

Outils mathématiques

L'étude de la complexité algorithmique fait appel à des outils mathématiques (théorie des probabilités, algèbre, algèbre combinatoire discrète, etc.). Calculer la complexité d'un algorithme consiste à determiner une *formule mathématique* qui exprime le coût de cet algorithme. Cette formule s'obtient en bornant des expressions de sommations et/ou de produits par un calcul de limite à l'infini de fonctions. Dans plusieurs situations les propriétés relationnelles entre les nombres réels s'appliquent aussi aux fonctions. Ce chapitre donne quelques rappels fondamentaux liés à l'utilisation des fonctions.

3.1 Sommation

Lorsqu'un algorithme contient une structure de contrôle répétitive, le temps d'exécution peut s'exprimer comme une somme des temps pour exécuter chaque instruction dans le corps de la boucle. Déterminer une fonction de complexité asymptotique exige le calcul d'expressions de sommation de série et savoir les borner.

3.1.1 Formules et propriétés des sommations

Soit une séquence de nombres $a_1, a_2, a_3, \ldots$, la somme infinie $a_1 + a_2 + \cdots + a_n + \ldots$

$$\sum_{k=1}^{\infty} a_k \tag{3.1}$$

est interprétée par la limite suivante :

$$\lim_{n \longrightarrow \infty} \sum_{k=1}^{k=n} a_k \tag{3.2}$$

- Si la limite n'existe pas, la série diverge, autrement elle converge.
- Si n n'est pas entier on suppose que la limite supérieure est $\lfloor n \rfloor$. [1]
- Si la somme commence avec $k = x$ et x n'est pas entier on supposera que la sommation commence avec l'entier $\lfloor x \rfloor$.

Linéarité

Soient $a_1, a_2, \ldots, a_n$ et $b_1, b_2, \ldots, b_n$ deux séquences de nombres et c un réel donné. La linéarité est une propriété définie par l'expression suivante : :

$$\sum_{k=1}^{k=n} (c * a_k + b_k) = c * \sum_{k=1}^{k=n} a_k + \sum_{k=1}^{k=n} b_k. \tag{3.3}$$

Cette propriété est aussi vérifiée pour les séries infinies convergentes. Elle est utile dans la manipulation des sommations qui contiennent des notations asymptotiques.

Exemple 3.1.1

$$\sum_{k=1}^{k=n} (\Theta(f(k)) = \Theta(\sum_{k=1}^{k=n} f(k)) \tag{3.4}$$

La fonction Θ s'applique à la variable k dans le membre gauche et à n dans le membre droit. Cette manipulation s'applique aussi aux séries infinies convergentes.

Séries arithmétiques

$$\sum_{k=1}^{k=n} k = 1 + 2 + 3 + \cdots + n = \frac{n * (n+1)}{2} = \Theta(n^2) \tag{3.5}$$

1. l'expression $\lfloor a \rfloor$ désigne le plus grand entier inférieur ou égal à a et $\lceil a \rceil$ désigne le plus petit entier supérieur ou égal à a.

Séries géométriques

Si x est un réel, $x \neq 1$, on a :

$$\sum_{k=0}^{k=n} x^k = 1 + x + x^2 + \cdots + x^n = \frac{(x^{n+1} - 1)}{x - 1} \tag{3.6}$$

est une série exponentielle (ou géométrique). Si $|x| < 1$ et la sommation est infinie, on obtient la série géométrique décroissante suivante :

$$\sum_{k=0}^{\infty} x^k = \frac{1}{1 - x} \tag{3.7}$$

3.1.2 Séries harmoniques

Pour des entiers n positifs, le $n^{\text{ième}}$ nombre harmonique vaut :

$$\begin{aligned} H_n &= 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \cdots + \frac{1}{n} \\ &= \sum_{k=1}^{k=n} \frac{1}{k} \\ &= \ln n + O(1). \end{aligned} \tag{3.8}$$

3.1.3 Séries emboîtées

Pour toute série quelconque $a_0, a_1, a_2, \ldots, a_n$ on écrit :

$$\sum_{k=1}^{n} (a_k - a_{k-1}) = a_n - a_0. \tag{3.9}$$

car chaque terme de $a_1, a_2, \ldots, a_{n-1}$ est ajouté et supprimé une seule fois. De même que

$$\sum_{k=0}^{n-1} (a_k - a_{k+1}) = a_0 - a_n. \tag{3.10}$$

Exemple 3.1.2

$$\sum_{k=1}^{n-1} \frac{1}{k * (k+1)} \tag{3.11}$$

On réécrit chaque terme sous la forme

$$\frac{1}{k*(k+1)} = \frac{1}{k} - \frac{1}{k+1} \tag{3.12}$$

d'où :

$$\begin{aligned} \sum_{k=1}^{n-1} \frac{1}{k*(k+1)} &= \sum_{k=1}^{n-1} (\frac{1}{k} - \frac{1}{k+1}) \\ &= a_0 - a_n \\ &= 1 - \frac{1}{n} \end{aligned} \tag{3.13}$$

3.1.4 Les produits

Le produit fini $a_1 * a_2 * a_3 * \cdots * a_n$ s'écrit :

$$\prod_{k=1}^{k=n} a_k \tag{3.14}$$

- Si $n = 0$ le produit vaut 1 par définition.
- Une formule de produit peut être convertie en une formule de sommation en passant par le logarithme :

$$\log(\prod_{k=1}^{k=n} a_k) = \sum_{k=1}^{k=n} \log a_k. \tag{3.15}$$

Exemple 3.1.3 *Exercices*

- *Trouver une formule simple pour :*

$$\sum_{k=1}^{k=n} (2 * k - 1). \tag{3.16}$$

- *En utilisant les séries harmoniques, montrer que*

$$\sum_{k=1}^{k=n} \frac{1}{2 * k - 1} = \ln(\sqrt{n}) + O(1) \tag{3.17}$$

- *En utilisant la série 3.7 et la propriété de convergence des dérivées de série convergente, calculez la somme suivante :*

$$\sum_{k=0}^{k=n} k * x^k. \tag{3.18}$$

3.2 Récurrences

3.2.1 Borner une sommation

La récurrence est une des techniques que l'on peut utiliser pour borner une sommation. Par exemple, on peut démontrer par récurrence que $\sum_{k=1}^{k=n} k = \frac{n*(n+1)}{2}$.

La récurrence peut aussi être utiliser pour prouver l'existence d'une borne.

Exemple 3.2.1 *Prouver que $\sum_{k=0}^{k=n} 3^k$ est en $O(3^n)$.*

On montre, par récurrence, qu'il existe une constante c et un entier n_0 telles que :

$$\sum_{k=0}^{k=n} 3^k \leq c * 3^n, \forall n \geq n_0. \tag{3.19}$$

- ***cas de base*** *: pour n=0 on a : $\sum_{k=0}^{k=n} 3^k = 3^0 = 1 \leq c * 1$ pour $c \geq 1$*
- ***hypothèse de récurrence*** *: on suppose la formule 3.19 est vraie à l'ordre n et montrons qu'elle reste vraie à l'ordre $n+1$.*
- $$\sum_{k=0}^{k=n+1} 3^k = \sum_{k=0}^{k=n} 3^k + 3^{n+1} \leq c * 3^n + 3^{n+1}$$
  Or $c * 3^n + 3^{n+1} = c * 3^{n+1}(\frac{1}{3} + \frac{1}{c}) \leq c * 3^{n+1}$ si $\frac{1}{3} + \frac{1}{c} \leq 1$ ce qui est vrai si $c \geq \frac{3}{2}$. C.Q.F.D

Remarque 3.2.2 *Attention aux erreurs !*

Dans la définition de O les constantes c et n_0 ne doivent pas varier avec n.

Contre-exemple : l'affirmation suivante : $\sum_{k=1}^{k=n} k = O(n)$ est fausse. La démonstration par récurrence suivante est fausse :

- ***cas de base*** *: pour n=0 on a : $\sum_{k=1}^{k=1} k = 1 \leq c * 1$ vraie pour $c \geq 1$*
- *hypothèse de récurrence : $\exists c, n_0$ telles que $\sum_{k=1}^{k=n} k \leq c * n, \forall n \geq n_0$*
- *Calculons $\sum_{k=1}^{k=n+1} k = \sum_{k=1}^{k=n} k + (n+1) \leq c*n + (n+1) = n*(c+1)+1$. La constante c croit avec n.*
- *Calculons $\sum_{k=1}^{k=n+2} k = \sum_{k=1}^{k=n} k + (n+1) + (n+2) \leq c * n + (n+1) + (n+2) = n*(c+2)+3$, on voit bien que la constante c n'est pas unique pour tout $n > n_0$.*

3.2.2 Borner un terme par le plus grand terme de la série

On peut borner une sommation en bornant chacun de ses termes par une borne supérieure intéressante. Il suffit dans plusieurs cas de borner tous les termes par le terme le plus grand, autrement dit :

$$\sum_{k=1}^{k=n} a_k \leq n * max\{a_1, a_2, \ldots, a_n\} \tag{3.20}$$

Exemple 3.2.3

$$n = 1 * 2 * 3 * \cdots * n \leq n * n * n * \ldots n = n^n. \tag{3.21}$$

$$\sum_{k=1}^{k=n} k \leq \sum_{k=1}^{k=1} n = n^2. \tag{3.22}$$

3.2.3 Borner une série par une série géométrique

Pour borner une série $\sum_{k=0}^{k=n} a_k$ par une série géométrique, on doit montrer qu'il existe une **constante** r telle que $r < 1$ et $a_{k+1}/a_k \leq r$. Si un tel r existe, alors $a_k \leq a_0 * r^k, \forall k \geq 0$. On obtient ainsi :

Exemple 3.2.4

$$\begin{aligned} \sum_{k=0}^{k=n} a_k \leq \sum_{k=0}^{\infty} a_0 * r^k &= \\ a_0 * \sum_{k=0}^{\infty} r^k &= \\ a_0 * \frac{1}{1-r}. \end{aligned} \tag{3.23}$$

Cette technique permet d'obtenir une meilleure borne que celle qui utilise le plus grand terme.

Exemple 3.2.5 *Trouver une borne pour* $\sum_{k=1}^{\infty} k/3^k$

- *Pour* $k = 1$, $a_1 = 1/3$

– *le rapport entre deux termes quelconques vaut :*

$$
\begin{aligned}
a_{k+1}/a_k = & \frac{(k+1)/3^{k+1}}{k/3^k} \\
& \frac{1}{3} * \frac{k+1}{k} \\
& \frac{1}{3} * (1 + \frac{1}{k}) \leq \frac{2}{3} \qquad \forall k \geq 1.
\end{aligned}
\tag{3.24}
$$

Tout terme a_k est donc borné par $a_0 * r^k = (1/3) * (2/3)^k$, d'où :

$$
\begin{aligned}
\sum_{k=1}^{\infty} k/3^k \leq & \sum_{k=1}^{\infty} (1/3) * (2/3)^k \\
& 1/3 * \frac{1}{1 - 2/3} = 1.
\end{aligned}
\tag{3.25}
$$

Remarque 3.2.6 *Pour appliquer cette méthode il est nécessaire de trouver $r < 1$ constant. Si r varie la méthode n'est pas applicable, comme pour la série divergente suivante :*

$$\sum_{k=1}^{\infty} 1/k = \lim_{n \longleftarrow \infty} \sum_{k=1}^{n} 1/k = \lim_{n \longleftarrow \infty} \Theta(\ln n) = \infty$$

Dans ce cas le rapport $a_{k+1}/a_k < r$ mais un r variable qui tend vers 1. Il n'y a pas de r constant, $r < 1$.

On peut appliquer cette méthode à partir d'un certain rang, à partir duquel il existe un r constant qui vérifie les conditions d'existence d'une série géométrique.

Exemple 3.2.7 *Par exemple, pour borner la série $\sum_{k=1}^{\infty} \frac{k^2}{2^k}$, on calcule la rapport a_{k+1}/a_k. On peut montrer que $a_{k+1}/a_k \leq 8/9$ si $k \geq 3$.*

$$
\begin{aligned}
\sum_{k=1}^{\infty} \frac{k^2}{2^k} = & \sum_{k=1}^{2} \frac{k^2}{2^k} + \sum_{k=3}^{\infty} \frac{k^2}{2^k} \\
& \leq O(1) + 9/8 \sum_{k=3}^{\infty} (8/9)^k \\
& O(1).
\end{aligned}
\tag{3.26}
$$

3.3 Calcul de sommes par intégrales

Définition 3.3.1 – *Une fonction est monotone croissante si $m \leq n \Longrightarrow f(m) \leq f(n)$.*
– *Une fonction est monotone décroissante si $m \leq n \Longrightarrow f(m) \geq f(n)$.*

Soit une expression de sommation $\sum_{k=m}^{n} f(k)$ à borner.

– Si f est une fonction **monotone et croissante** on a :

$$\int_{m-1}^{n} f(x)dx \leq \sum_{m}^{n} f(k) \leq \int_{m}^{n+1} f(x)dx \tag{3.27}$$

– Si f est une fonction **monotone et décroissante**, on a :

$$\int_{m}^{n+1} f(x)dx \leq \sum_{m}^{n} f(k) \leq \int_{m-1}^{n} f(x)dx \tag{3.28}$$

Exemple 3.3.2 *Bornons $\sum_{k=1}^{n} \frac{1}{k}$ par cette méthode. Puisque $f(k) = 1/k$ est monotone décroissante, alors :*

$$\int_{1}^{n+1} \frac{1}{x}dx \leq \sum_{1}^{n} \frac{1}{k} \leq \int_{0}^{n} \frac{1}{x}dx \tag{3.29}$$

Nous avons alors d'une part :

$$\begin{aligned} \sum_{k=1}^{n} \frac{1}{k} &\geq \int_{1}^{n+1} \frac{dx}{x} \\ &\geq \ln(n+1). \end{aligned} \tag{3.30}$$

et d'autre part :

$$\begin{aligned} \sum_{k=1}^{n} \frac{1}{k} &= 1 + \sum_{k=2}^{n} \frac{1}{k} \\ &\leq 1 + \int_{1}^{n} \frac{dx}{x} \\ &\leq 1 + \ln n \end{aligned} \tag{3.31}$$

3.4 Fonctions classiques

3.4.1 La partie entière

C'est une fonction monotone et croissante. Elle est définie comme suit : pour tout réel x,

1. le plus grand entier inférieur ou égal à x s'écrit $\lfloor x \rfloor$ (ou la partie entière de x) ;
2. le plus petit entier supérieur à x s'écrit $\lceil x \rceil$.

$$x - 1 < \lfloor x \rfloor \leq x \leq \lceil x \rceil < x + 1$$

Pour tout entier n :

$$\lfloor n/2 \rfloor + \lceil n/2 \rceil = n$$

3.4.2 La fonction exponentielle

Pour tout réel $a \neq 0, m, n$ on a les identités suivantes :

$$\begin{aligned} a^0 &= 1 \\ a^1 &= a \\ a^{-1} &= 1/a \\ (a^m)^n &= (a^n)^m = a^{mn} \\ a^m * a^n &= a^{m+n} \end{aligned}$$

- La fonction a^n est monotone et croissante pour $a \geq 1$. On pose $0^0 = 1$ si nécessaire ;
- comparaison avec un polynôme : $\lim_{n \longrightarrow \infty} \frac{n^b}{a^n} = 0$;
- d'où : $n^b = o(a^n)$, une fonction exponentielle avec une base strictement plus grande que 1 croît plus vite qu'un polynôme quelconque.
- si on s'interesse au réel 2.71828 . . . qui est la base e du logarithme népérien, on a :

  $$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots = \sum_{k=0}^{\infty} \frac{x^k}{k!}$$

  d'où $e^x \geq 1 + x$. L'égalité est vérifiée pour $x = 0$.

- Si $\mid x \mid \leq 1$, alors : $1 + x \leq e^x \leq 1 + x + x^2$.
- On s'intéresse au comportement asymptotique quand $x \longrightarrow 0$, on a $e^x = 1 + x + \Theta(x^2)$, et pour tout x : $\lim_{n \longrightarrow \infty}(1 + \frac{x}{n})^n = e^x$.

3.4.3 La fonction logarithme

Le logarithme d'un nombre dans une base c'est l'exposant qu'il faut donner à la base pour obtenir ce nombre. Par exemple 3 est l'exposant qu'il faut donner à 2 pour obtenir 8, donc $2^3 = 8 \Longrightarrow \log_2 8 = 3$.

- Notations
  1. Logarithme népérien : $\log_e n = \ln n$
  2. Logarithme binaire : $\log_2 n = \frac{\ln n}{\ln 2}$
  3. Exponentiation : $\log^k n = (\log n)^k$
  4. Composition : $\log \log n = \log(\log n)$
- Propriétés de calcul

  Pour tout réels : $a > 0, b > 0, c > 0$ et entier n on a :

  1. $a = b^{\log_b a}$
  2. $\log_c(ab) = \log_c a + \log_c b$
  3. $\log_b a^n = n \log_b a$
  4. $\log_b a = \frac{\log_c a}{\log_c b}$
  5. $\log_b(1/a) = -\log_b a$
  6. $\log_b a = \frac{1}{\log_a b}$
  7. $a^{\log_b n} = n^{\log_b a}$

Ainsi le changement de base d'un logarithme ne modifie la valeur de ce logarithme que d'un facteur constant, qui est négligé dans une notation O.

3.5 Méthodes de résolution des équations de récurrence

Généralement le temps d'exécution d'une fonction récursive s'exprime par une équation de récurrence. L'équation de récurrence décrit une fonction à partir de sa valeur sur des entrées plus petites.