# ALGORITHME EN ARBRE ET THÉORÈME DE RENOUVELLEMENT

*Hanène Mohamed*

***Résumé.*** — Un réseau de communication à accès multiple est un système distribué de $n$ noeuds, appelés *stations*, partageant un canal commun de transmission ne pouvant transmettre qu'un seul message par unité de temps. Ce canal renvoie une information ternaire, i.e. chaque station envoyant un message au réseau peut simultanément écouter le canal et ainsi détecter une *collision* s'il y a eu au moins deux essais de transmission, un *silence* si aucune station n'a envoyé de messages, ou alors un *succès* si une seule station a transmis son message vers le réseau.

**Question :** Comment un tel système sans contrôle central peut s'ordonner pour transmettre ses messages en un temps fini ?

On s'intéresse à l'algorithme décrit ci-dessus, appelé *algorithme en arbre* ou *diviser pour régner*, et ce d'un point de vue probabiliste.

## 1. Algorithme en arbre

Dans un réseau de communication *à accès multiple*, chaque station a pour objectif de transmettre son message. Soit $n$ la taille, inconnue, de ce réseau. On suppose que $n \geq 2$, le cas $n \in \{0, 1\}$ étant trivial. Ci-dessous, l'algorithme en arbre est décrit ;

— Initialisation déterministe : Au début de la première unité de temps, chaque station s'attribue un compteur initialisé à zéro, puis envoie son message via le canal de transmission. Comme $n \geq 2$, toutes les stations détectent une *collision*.

— Processus d'éclatement aléatoire : Chaque station S tire à *pile* ou *face* d'une manière indépendante. Les stations obtenant *pile* gardent un compteur nul. Les autres voient leurs compteurs augmentés de 1.

Pour une station S, il y a deux cas :

(1) Si son compteur $c_S = 0$, la station S est appelée *Active* ; S envoie son message via le canal et ainsi peut détecter

— un *succès* ; la station S a été la seule à essayer de transmettre son message. L'algorithme est fini pour S.

— une *collision* ; la station S tire à nouveau *pile* ou *face*.

(2) Sinon, la station S devient *Non Active* ; tant que son compteur est non nul $c_S \neq 0$, elle reste en attente et écoute le canal. Si elle détecte

— un *succès* ou un *silence* ; S diminue alors son compteur de 1 : $c_S \leftarrow c_S - 1$.
— une *collision* ; elle l'augmente de 1 : $c_S \leftarrow c_S + 1$.

Ainsi, à la fin de l'algorithme, tous les messages seront transmis.

| unités temps | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| Actives | A B C D | A B | | A B | A | B | CD | C | D |
| Compteur=1 | | C D | A B | C D | B | CD | | D | |
| Compteur=2 | | | C D | | CD | | | | |
| État du canal | *Collision* | *Collision* | *Silence* | *Collision* | *Succès* | *Succès* | *Collision* | *Succès* | *Succès* |

Formellement, l'algorithme part avec un groupe de $n$ objets qui sera divisé en 2 sous-groupes. La probabilité qu'un objet soit mis dans le sous-groupe de gauche est notée $p$, celle de droite $q$ où $p + q = 1$. Ainsi décrit, l'algorithme présente une structure arborescente, voir Fig1.

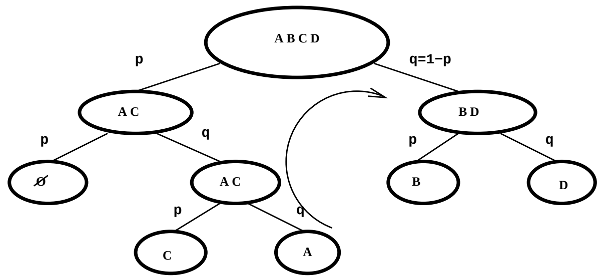



FIGURE 1. Structure d'arbre, cas de 4 stations A B C D ; $R_4 = 9$.

**Coût de l'algorithme :** C'est le nombre d'étapes nécessaires pour arriver à vérifier la condition d'arrêt sur chaque sous-groupe généré par l'algorithme. On note $R_n$ cette quantité quand le groupe initial est de taille $n$. C'est aussi le nombre de noeuds dans l'arbre associé à l'algorithme, voir Fig1.

Dans un contexte de réseaux de communication, $R_n$ est le temps (discret) nécessaire pour transmettre $n$ messages. La limite du ratio $\mathbb{E}(R_n)/n$, si elle existe, serait interprétée comme le temps moyen de transmission d'un message. Curieusement, cette limite n'existe pas toujours, d'ailleurs, pour le cas non biaisé $p = q = 1/2$, $\mathbb{E}(R_n)/n$ présente un comportement oscillatoire.

Cet exposé a pour objectif d'établir une formule exacte pour le *coût moyen de l'algorithme* $\mathbb{E}(R_n)$ et d'étudier le comportement asymptotique du ratio $\mathbb{E}(R_n)/n$ en utilisant un résultat de probabilités ; *le théorème de renouvellement*.

## 2. Coût moyen de l'algorithme

*Mesure d'éclatement :* On définit la distribution de probabilité $W$ sur $[0,1]$ comme étant la *mesure d'éclatement de l'algorithme*

$$W = p\,\delta_p + q\,\delta_q,\ \ \delta \text{ étant la masse de Dirac.}$$

**Équation de récurrence :** Le caractère récursif de l'algorithme se traduit par une relation de récurrence sur la suite $(R_n)_{n\geq 2}$

$$R_n \overset{\text{dist.}}{=} 1 + R_{S(n)} + R_{n-S(n)},\ \text{pour } n \geq 2,$$

où $R_0 = R_1 = 1$, $S(n) = \sum_{i=1}^{n} \mathbf{1}_{\{A=p\}}$, avec $A$ une variable aléatoire de loi $W$. Intuitivement, $S(n)$ est le nombre d'éléments mis dans le sous-groupe de gauche, ou encore le nombre des stations tirant pile au premier tirage. Ainsi, pour $n \geq 0$, l'équation de récurrence sur $R_n$ peut être réécrite

$$R_n \overset{\text{dist.}}{=} 1 + R_{S(n)} + R_{n-S(n)} - 2\,\mathbf{1}_{(n\leq 1)}. \tag{1}$$

A ce stade, notre objectif est de déterminer $\mathbb{E}(R_n)$ à partir de l'équation $(1)$, mais la dépendance entre les indices $S(n)$ et $n-S(n)$ rend sa manipulation difficile. Pour ce, on introduit la notion de *processus de Poisson* ;

*Processus de Poisson :* Soit $(t_n)_{n\geq 1}$ une suite non décroissante telle que $(t_{n+1} - t_n)$ soit une suite de variables *i.i.d.* de distribution exponentielle de paramètre 1. Pour $x \geq 0$, la variable $N_x$ définissant le nombre de $t_n$ dans l'intervalle $[0,x]$ est appelée *processus de Poisson* d'intensité 1.

**Poissonisation :**

$$(R_n)_{n\geq 0} \overset{\textit{Poisson}}{\to} (r(x),\ x \geq 0)$$

Si on considère un modèle Poissonnien, i.e. la taille du groupe initial à traiter $n$ est aléatoire suivant un processus de *Poisson* $(N_x,\ x \geq 0)$, on supprime le problème de la dépendance des indices. En effet,

$$(S(n),\ n - S(n)) \overset{\textit{Poissonisation}}{\to} (S(N_x),\ N_x - S(N_x)) \overset{\text{dist.}}{=} (N^1_{px},\ N^2_{qx})$$

où $(N^1_{px},\ N^2_{qx})$ désigne un couple de processus de Poisson indépendants d'intensités respectives $p$ et $q$.

La relation de récurrence (1) est ainsi transformée en une équation fonctionnelle

$$r(x) = r(px) + r(qx) + 2\,\mathbb{P}(N(x) \geq 1), \tag{2}$$

où la fonction $r$, *transformée de Poisson* du coût de l'algorithme, est définie par

$$r(x) = \mathbb{E}(R_{(N_x)}).$$

**Formulation probabiliste et schéma itératif :** L'équation fonctionnelle (2) est simplement réécrite en fonction d'une variable aléatoire $A$ de distribution $W$

$$r(x) = \mathbb{E}\left(\frac{r(Ax)}{A}\right) + 2\,\mathbb{P}(N(x) \geq 1).$$

Ainsi reformulée, la résolution de l'équation fonctionnelle (2) utilisant un schéma itératif simple implique que la *transformée de Poisson* du coût de l'algorithme est donnée par

$$r(x) = 1 + 2\,\mathbb{E}\left(\sum_{k=0}^{\infty} \frac{1}{\pi_k} \mathbf{1}_{(t_2 \le x\ \pi_k)}\right),$$

où

– $\pi_0 = 1$, et pour $k \ge 1$, $\pi_k = \prod_{i=0}^{k-1} A_i$, avec $(A_i)_{i\ge 0}$ une suite i.i.d. de distribution $W$.

– $t_2$ est la somme de deux variables aléatoires indépendantes de distribution exponentielle de paramètre 1.

**Dépoissonisation :**

$$(R_n)_{n\ge 0} \longleftarrow (r(x),\ x \ge 0)$$

Ayant obtenu une formule exacte pour la *transformée de Poisson* $r$, reste à déterminer le coût moyen de l'algorithme $\mathbb{E}(R_n)$. Cette transformation inverse est basée sur un argument probabiliste simple. Les points d'un processus de *Poisson* sur un intervalle $[0, x]$, sachant que le nombre de ces points est $n$, ont la même distribution que $n$ uniformes sur $[0, x]$ ;

$$(t_1, t_2, \ldots, t_n | N_x = n) \stackrel{\text{dist.}}{=} (x\, U_{1,n}, x\, U_{2,n}, \ldots, x\, U_{n,n}),$$

où $((U_{i,n})_{1\le i\le n}$ sont $n$ uniformes sur $[0, 1]$. Ainsi, pour le modèle binomial, i.e. de taille $n$ fixée, le coût moyen de l'algorithme est donné par

$$\mathbb{E}(R_n) = 1 + 2\,\mathbb{E}\left(\sum_{k=0}^{\infty} \frac{1}{\pi_k} \mathbf{1}_{(U_{2,n} \le \pi_k)}\right).$$

## 3. Analyse asymptotique du coût moyen

**Marche aléatoire associée :** On définit la marche aléatoire de saut $B = -\log A$,

$$S_k = -\log(\pi_k) = B_0 + \ldots + (B_{k-1}), k \ge 1.$$

Pour $x > 0$, le temps de dépassement $\nu(x)$ de la barrière $x$ par la marche aléatoire $(S_k)$ est défini par

$$\nu(x) = \inf\{k \ge 0;\ S_k > x\}.$$

Alors, le coût moyen de l'algorithme est donné par

$$\mathbb{E}(R_n) = 1 + 2\,\mathbb{E}\left(\sum_{k=0}^{\nu(-\log U_{2,n})-1} e^{S_k}\right).$$

**Théorème de renouvellement.** Le théorème de renouvellement appliqué à la marche aléatoire $(S_k)$ dépend de l'arithméticité du saut $B = -\log A$, *i.e.* s'il existe $\lambda > 0$ tel que $\mathbb{P}(B \in \lambda\mathbb{N}) = 1$. Le plus grand réel $\lambda$ vérifiant la propriété d'arithméticité définit le *pas* du saut B, ou encore le *pas exponentiel* de la variable aléatoire A.

***Remarque 3.1***. — Dans notre cas,

$$(-\log A\ )\ \text{arithmétique} \Leftrightarrow \left(\frac{\log p}{\log q} \in \mathbb{Q}\right).$$

— **Cas non-arithmétique :** L'analyse du comportement asymptotique du ratio $\mathbb{E}(R_n)/n$ revient à étudier celui

$$e^{-x}\ \psi(x) := \mathbb{E}\left(\sum_{k=0}^{\nu(x)-1} e^{S_k - x}\right) = \mathbb{E}\left(\sum_{k=1}^{\nu(x)} e^{S_{\nu(x)-k} - x}\right).$$

Le théorème de renouvellement dans sa version continue établit une limite en loi de la suite $\left(S_{\nu(x)-k} - x\right)_{k\geq 0}$ quand $x$ tend vers l'infini. Ainsi, on montre que

$$\lim_{x\to\infty} e^{-x}\ \psi(x) = \frac{-1}{\mathbb{E}(\log A)},$$

et par suite, que le ratio $\mathbb{E}(R_n)/n$ est asymptotiquement équivalent à $(-2/\mathbb{E}(\log A))$.

— **Cas arithmétique :** Si $\log p/\log q \in \mathbb{Q}$, alors il existe un couple d'entiers $(a, b)$; $b \neq 0$ tel que $b \log p = a \log q$. Notons $\lambda = -\log q/b$ le *pas exponentiel* de la variable aléatoire $A$, i.e. $-\log A \in \lambda\mathbb{N}$ presque sûrement, alors la marche aléatoire normalisée $S_k^*$

$$S_k^* := S_k/\lambda \in \mathbb{N}$$

est discrète. Le temps de dépassement associé est défini par

$$\nu^*(\frac{x}{\lambda}) = \nu^*(\lceil\frac{x}{\lambda}\rceil),$$

où $\lceil\ \rceil$ désigne la partie entière augmentée de 1 ; $\lfloor\ \rfloor + 1$.

L'analyse du comportement asymptotique du ratio $\mathbb{E}(R_n)/n$ revient alors à étudier celui

$$e^{-\lambda\lceil\frac{x}{\lambda}\rceil}\ \psi(x) := \mathbb{E}\left(\sum_{k=0}^{\nu^*(\lceil\frac{x}{\lambda}\rceil)-1} e^{\lambda(S_k^* - \lceil\frac{x}{\lambda}\rceil)}\right) = \mathbb{E}\left(\sum_{k=1}^{\nu^*(\lceil\frac{x}{\lambda}\rceil)} e^{\lambda(S_{\nu^*(\lceil\frac{x}{\lambda}\rceil)-k} - \lceil\frac{x}{\lambda}\rceil)}\right).$$

En utilisant le théorème de renouvellement dans sa version discrète appliquée à la marche aléatoire normalisée $S_k^*$ et au temps de dépassement $\nu^*$ de la barrière $\lceil\frac{x}{\lambda}\rceil$, on obtient que

$$\lim_{x\to\infty} e^{-\lambda\lceil\frac{x}{\lambda}\rceil}\ \psi(x) = \frac{-1}{\mathbb{E}(\log A)}\frac{\lambda e^{-\lambda}}{1-e^{-\lambda}}.$$

En notant la partie fractionnaire $\{z\} = z - \lfloor z \rfloor$, on montre que le ratio $\mathbb{E}(R_n)/n$ est asymptotiquement équivalent à $F(\log n/\lambda)$, où pour tout $z \geq 0$, la fonction $F$ est définie par

$$F(z) = \frac{-1}{\mathbb{E}(\log A)}\frac{\lambda e^{-\lambda}}{1-e^{-\lambda}}\int_0^{+\infty} e^{-\lambda\{z-\log y/\lambda\}}e^{-y}\,dy.$$

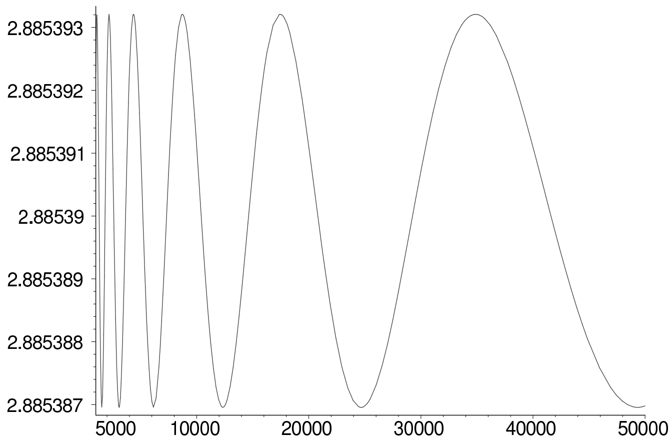
La fonction $F$ est périodique, d'où le comportement oscillatoire de l'algorithme.

**Conclusion :**

— Si $\log p / \log q \notin \mathbb{Q}$, alors *le coût moyen de l'algorithme est de l'ordre de* $n$.

— Sinon, il existe un couple d'entiers $(a, b)$; $b \neq 0$ tel que $b \log p = a \log q$. Alors *le coût moyen de l'algorithme est de l'ordre de* $nF(\log n/\lambda)$, où $F$ est une fonction périodique et $\lambda = -\log q/b$.

***Remarque 3.2***. — Le cas non biaisé $p = q = 1/2$, est arithmétique avec un *pas exponentiel* $\lambda = \log 2$. Ci-dessous le graphe de la suite : $n \to \mathbb{E}(R_n)/n$.




## Références


[1] John I. Capetanakis, *Tree algorithms for packet broadcast channels*, IEEE Transactions on Information Theory **25**, 1979.

[2] B. S. Tsybakov and V. A. Mikhaïlov, *Free synchronous packet access in a broadcast channel with feedback*, Problems Inform. Transmission **14**, 1978.

[3] P. Mathys and P. Flajolet, *Q-ary collision resolution algorithms in random access systems with free or blocked channel access*, IEEE Transactions on Information Theory **31**, 1985.

[4] Philippe Robert, *On the asymptotic behavior of some algorithms*, Random Structures & Algorithms **27**, 2005.

[5] Hanène Mohamed and Philippe Robert, *A probabilistic analysis of some tree algorithms*, Annals of Applied Probability **15**, 2005.


*Hanène Mohamed*

INRIA Rocquencourt, Domaine de Voluceau, B.P. 105, 78153 Le Chesnay Cedex France.

*E-mail :* `Hanene.Mohamed@inria.fr`