



**HAL**  
open science

## Collecter un nombre inconnu de coupons

Philippe Jacquet, Mohamed Lamine Lamali, Fabien Mathieu

► **To cite this version:**

Philippe Jacquet, Mohamed Lamine Lamali, Fabien Mathieu. Collecter un nombre inconnu de coupons. CORES 2018 - Rencontres Francophones sur la Conception de Protocoles, l'Évaluation de Performance et l'Expérimentation des Réseaux de Communication, May 2018, Roscoff, France. pp.1-4. hal-01787252

**HAL Id: hal-01787252**

**<https://inria.hal.science/hal-01787252>**

Submitted on 7 May 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Collecter un nombre inconnu de coupons

Philippe Jacquet<sup>1</sup>, Mohamed Lamine Lamali<sup>2</sup> et Fabien Mathieu<sup>1</sup>

<sup>1</sup> Nokia Bell Labs France

<sup>2</sup> LaBRI – Université de Bordeaux

---

Dans les algorithmes de cartographie d’Internet, l’un des problèmes clés à résoudre est celui de la découverte des différentes interfaces de sortie d’un routeur en cas d’équilibrage de charge : combien de sondes doit-on envoyer tester la connexion pour être raisonnablement certain d’avoir découvert toutes les interfaces ? Une des solutions existantes est une variante du problème du collecteur de coupons, qui consiste à associer à chaque nombre d’interfaces découvertes un nombre de sondes au-delà duquel on considère que l’on peut s’arrêter. Dans cet article, nous donnons une formule close pour l’association et prouvons sa quasi-optimalité.

**Mots-clés :** Collecteur de coupons, équilibrage de charge, métrologie

---

## 1 Introduction

Le projet Paris Traceroute<sup>†</sup> vise à fournir des outils permettant de réaliser une cartographie d’Internet. L’un des problèmes à résoudre est celui de l’existence de routes multiples entre une source et une destination données. Un même routeur est en effet susceptible de relayer un paquet à travers plusieurs interfaces, par exemple pour équilibrer la charge. La FIGURE 1 montre l’exemple d’un routeur susceptible de choisir parmi  $k$  interfaces distinctes pour atteindre la destination.

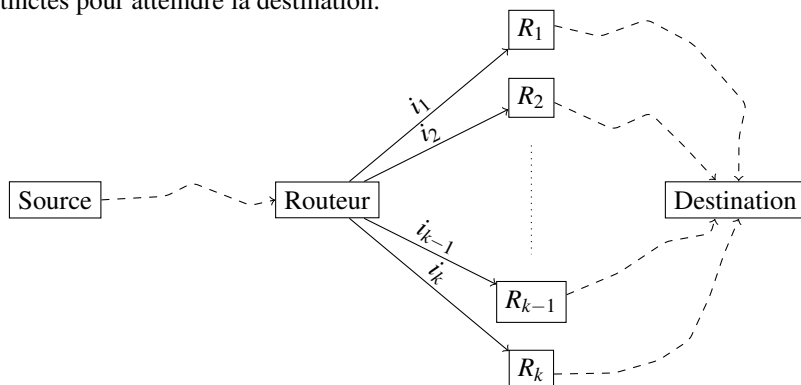


FIGURE 1: Un routeur à  $k$  interfaces de sortie.

Pour détecter toutes les routes possibles, on peut envoyer des sondes qui seront redirigées vers l’une ou l’autre des interfaces. La question est : combien de sondes faut-il envoyer afin d’être raisonnablement certain d’avoir détecté les  $k$  interfaces de sortie ?

Le *Multipath Detection Algorithm* (MDA) a été proposé pour résoudre le problème dans le cas où chaque nouvelle sonde est envoyée au hasard sur l’une des interfaces de manière i.i.d. uniforme [AFT07, VATF09]. Le principe consiste à choisir des coefficients  $n_k$  qui vont déterminer les conditions d’arrêt du sondage : si strictement moins de  $k$  interfaces ont été découvertes au bout de  $n_k$  sondes, le sondage s’arrête et renvoie le nombre d’interfaces découvertes, sinon  $k$  est incrémenté. Le problème est de choisir des paramètres  $n_k$  qui soient aussi petits que possible tout en garantissant que le nombre réel d’interfaces, quel qu’il soit, est

---

<sup>†</sup> <https://paris-traceroute.net>

renvoyé par l'algorithme avec une probabilité de  $1 - \varepsilon$ , le paramètre de précision  $\varepsilon$  étant fixé. L'algorithme cherche donc à résoudre un problème de collecte de coupons où le nombre total de coupons distincts est inconnu.

Les travaux précédents résolvent le problème par une analyse de l'espace des  $(i, k)$  du nombre d'interfaces  $k$  découvertes après  $i$  sondes [AFT07, VATF09]. Il en résulte une formule récurrente qui détermine des coefficients  $n_k$ . Comme la formule n'est pas explicite, il est nécessaire de construire une table des valeurs de  $n_k$  adaptée pour chaque valeur de  $\varepsilon$ .

Après quelques résultats préliminaires donnés dans la Section 2, nous proposons en section 3 notre principal résultat : une nouvelle approche qui ne nécessite pas de parcourir l'espace des états et donne une formule simple et explicite de calcul des  $n_k$  quand l'équilibrage de charge est uniforme. Nous montrons que les coefficients obtenus sont quasi-optimaux (dans un sens qui sera défini plus tard) et améliorent la compréhension des cas d'échec. Nous indiquons enfin dans la section 4 comment adapter la situation aux cas de répartition non-uniforme de la charge.

## 2 Nombre connu de coupon

Pour commencer, fixons  $k \geq 2$  et  $0 < \varepsilon < 1$ . Comment choisir le plus petit  $n_k$  tel qu'avec probabilité  $1 - \varepsilon$ ,  $n_k$  sondes trouvent les  $k$  interfaces ?

**Lemme 1.** *Soit  $P(k, n_k)$  la probabilité de ne pas découvrir  $k$  interfaces au bout de  $n_k$  essais. Alors*

$$1 - \left(1 - \left(1 - \frac{1}{k}\right)^{n_k}\right)^k \leq P(k, n_k) \leq k \left(1 - \frac{1}{k}\right)^{n_k} \quad (1)$$

*En particulier, pour avoir  $P(k, n_k) \leq \varepsilon$ , il est suffisant d'avoir*

$$n_k \geq \left\lceil \frac{\ln\left(\frac{\varepsilon}{k}\right)}{\ln\left(\frac{k-1}{k}\right)} \right\rceil \quad (2)$$

*et nécessaire d'avoir*

$$n_k \geq \left\lceil \frac{\ln\left(1 - (1 - \varepsilon)^{1/k}\right)}{\ln\left(\frac{k-1}{k}\right)} \right\rceil. \quad (3)$$

*Démonstration.* Soit  $M_i$  l'événement *ne pas trouver une interface  $i$  donnée*. Le terme  $\left(1 - \frac{1}{k}\right)^{n_k}$  est tout simplement  $\mathbf{P}(M_i)$ . Le terme de droite de l'équation (1) vient de l'inégalité de Boole : la probabilité de manquer au moins une interface est inférieure à  $\sum_i \mathbf{P}(M_i)$ . Le terme de gauche découle de l'observation que les événements  $M_i$  et  $M_{i'}$  sont anti-corrélés pour  $i \neq i'$ , i.e.,  $\mathbf{P}(M_i | M_{i'}) \leq \mathbf{P}(M_i)$  : si une interface n'a pas été trouvée, les autres interfaces se partagent la totalité des sondes, ce qui augmente la probabilité de les trouver. Les événements  $\bar{M}_i$  sont donc aussi anti-corrélés. On en déduit  $\mathbf{P}(\cap_i \bar{M}_i) \leq \prod_i \mathbf{P}(\bar{M}_i)$ , ce qui permet d'obtenir le terme de gauche. Les équations (2) et (3) se déduisent immédiatement de (1).  $\square$

Il est intéressant de constater que les deux bornes sont très proches. D'un point de vue théorique, on a en effet  $1 - (1 - \varepsilon)^{1/k} = \frac{\varepsilon}{k} + O(\varepsilon^2)$ . Numériquement, le ratio des deux bornes est très proche de 1 même pour des valeurs de  $\varepsilon$  relativement élevées.

Le lemme 1 montre que choisir  $n_k = \left\lceil \frac{\ln\left(\frac{\varepsilon}{k}\right)}{\ln\left(\frac{k-1}{k}\right)} \right\rceil$  est une manière simple de garantir que la probabilité de ne pas détecter  $k$  interfaces est inférieure à  $\varepsilon$ , mais aussi que ce choix de  $n_k$  est proche de l'optimal (en pratique, il est presque tout le temps égal).

## 3 Nombre inconnu de coupons

L'algorithme MDA doit utiliser au minimum les valeurs suggérées par le lemme 1 pour être capable de renvoyer  $k$  valeurs au bout de  $n_k$  essais avec une précision  $\varepsilon$ . Par contre, ces valeurs pourraient ne pas être suffisantes, car l'algorithme peut s'arrêter avant d'avoir envoyé  $n_k$  sondes : au bout de  $n_2$  sondes s'il ne voit qu'une interface, au bout de  $n_3$  sondes s'il n'en voit que deux, etc. Le Théorème 1 montre que ce risque est minime.

Collecter un nombre inconnu de coupons

**Théorème 1.** On considère l'algorithme MDA utilisant comme paramètres  $n_k = \lceil \frac{\ln(\frac{\varepsilon}{k})}{\ln(\frac{k-1}{k})} \rceil$ . Si  $\varepsilon < 1/2$ , la probabilité que l'algorithme ne trouve pas le bon nombre d'interfaces est inférieure à  $\varepsilon + O(\varepsilon^{\log_2(3)})$ .

*Idée de la preuve.* À  $k$  fixé, on montre par l'inégalité de Boole que la probabilité que MDA renvoie  $i < k$  est inférieure à  $\binom{k}{i} \left(\frac{i}{k}\right)^{n_{i+1}}$ . On obtient ainsi une borne à la probabilité  $P$  que l'algorithme MDA ne renvoie pas la bonne réponse :

$$P \leq \sum_{i=1}^{k-1} \binom{k}{i} \left(\frac{i}{k}\right)^{n_{i+1}}. \quad (4)$$

On définit ensuite  $B_i(k) = \binom{k}{i} \left(\frac{i}{k}\right)^{\frac{\ln(\frac{\varepsilon}{i+1})}{\ln(\frac{i}{i+1})}}$ . Comme  $B_i(k) \geq \binom{k}{i} \left(\frac{i}{k}\right)^{n_{i+1}}$ , il nous suffit de montrer que  $\sum_{i=1}^{k-1} B_i(k) = \varepsilon + O(\varepsilon^{\log_2(3)})$ .

Chaque  $B_i(k)$  borne la probabilité que l'algorithme renvoie  $i$  quand il y a  $k$  interfaces. Pour  $k = 1$ , l'algorithme ne peut pas se tromper. Pour  $k = 2$ , on vérifie que  $B_1(2) = \varepsilon$ . Pour  $k \geq 3$ , on montre successivement les résultats suivants en étudiant la séquence des  $(B_i(k))_{1 \leq i \leq k-1}$  :

- Les valeurs aux extrémités sont  $B_1(k) = \varepsilon^{\log_2(k)}$  et  $B_{k-1}(k) = \varepsilon$ .
- En utilisant la formule de Stirling, on construit une fonction continue  $b$  sur  $[0, 1]$  telle que  $b(i/k) = B_i(k)$ . L'étude de  $b'$  montre que la séquence des  $B_i$  est bimodale, d'abord décroissante puis croissante. En particulier, pour tout  $1 \leq \ell < k$ , on a  $\sum_{i=1}^{\ell} B_i(k) \leq \ell \max(B_1(k), B_{\ell}(k))$ .
- On montre que pour  $i = k - 2$ , on a  $B_{k-2}(k) = O(\varepsilon^{\log_2(3)})$ .
- À proximité de  $k$ , l'étude de  $b'$  permet de prouver que la séquence des  $B_i$  se comporte à peu près comme une suite géométrique de raison  $e$ . Plus précisément, pour  $j = O(\ln(k))$ , on a  $B_{k-2-j}(k) = O(B_{k-2}(k)e^{-j}) = O(\varepsilon^{\log_2(3)}e^{-j})$ .

On décompose ensuite la somme en utilisant comme pivot  $\ell = k - 2 - \lceil \log(k) \rceil$ . On a ainsi

$$P \leq \ell \max\{B_1(k), B_{\ell}(k)\} + \sum_{\ell+1}^{k-2} B_i(k) + B_{k-1}(k). \quad (5)$$

On constate tout d'abord que  $\ell B_1(k) \leq (k-2)\varepsilon^{\log_2(k)} \leq \varepsilon^{\log_2(3)}$  (l'hypothèse  $\varepsilon < 1/2$  permet de montrer que la borne est maximale en  $k = 3$ ), puis que  $B_{\ell}(k) = O(\varepsilon^{\log_2(3)}/k)$ . Cela indique que le premier terme de (5) est en  $O(\varepsilon^{\log_2(3)})$ . Le deuxième terme est également en  $O(\varepsilon^{\log_2(3)})$  en raison du caractère géométrique de la séquence des  $B_i(k)$  quand  $i$  est proche de  $k$ . Le dernier terme vaut  $\varepsilon$  et conclut la preuve.  $\square$

Le théorème 1 montre que le principal risque d'erreur en utilisant MDA avec les paramètres  $n_k = \lceil \frac{\ln(\frac{\varepsilon}{k})}{\ln(\frac{k-1}{k})} \rceil$  est de renvoyer  $k - 1$  au lieu de  $k$ , avec un risque inférieur à  $\varepsilon$ . La probabilité d'une erreur plus importante (renvoyer un nombre d'interfaces strictement inférieur à  $k - 1$ ) est négligeable devant  $\varepsilon$ .

Afin d'évaluer l'effet pratique du terme en  $\varepsilon^{\log_2(3)}$  du théorème 1, la FIGURE 2 montre l'écart relatif entre la borne supérieure donnée par l'équation (4) et  $\varepsilon$ , en fonction de  $k$ , pour quelques valeurs de  $\varepsilon$ . On vérifie qu'à  $\varepsilon$  fixé, l'écart est uniformément borné en  $k$ . On observe que pour une précision grossière comme 0.25, l'écart est important : jusqu'à 35% de risque d'erreur supplémentaire pour les petites valeurs de  $k$ , 15% sur les grandes. Comme prédit par le théorème, cet écart se réduit avec  $\varepsilon$ , et pour une précision fine de  $\varepsilon = 0.01$ , la plupart des valeurs de  $k$  ont une garantie inférieure à  $\varepsilon$ , avec un dépassement de l'ordre de 1% sur quelques valeurs. Si l'on veut une plus grande maîtrise du taux d'erreur, les évaluations numériques suggèrent une heuristique simple pour  $\varepsilon \leq 0.1$  : comme le risque observé semble empiriquement borné par  $\varepsilon(1 + \varepsilon)$  (soit une borne meilleure que celle donnée par le théorème 1), on pourrait utiliser  $\varepsilon/(1 + \varepsilon)$  au lieu de  $\varepsilon$  dans le calcul des  $n_k$ .

## 4 Équilibrage de charge non-uniforme

Nous supposons maintenant un équilibrage de charge i.i.d. mais non-uniforme : certaines interfaces peuvent être plus probables que d'autres. Il est facile de voir qu'il est impossible de concevoir un algorithme permettant de trouver toutes les interfaces avec une précision donnée : si une des interfaces a une probabilité de sélection arbitrairement faible, alors le nombre de sondes nécessaires pour la découvrir peut

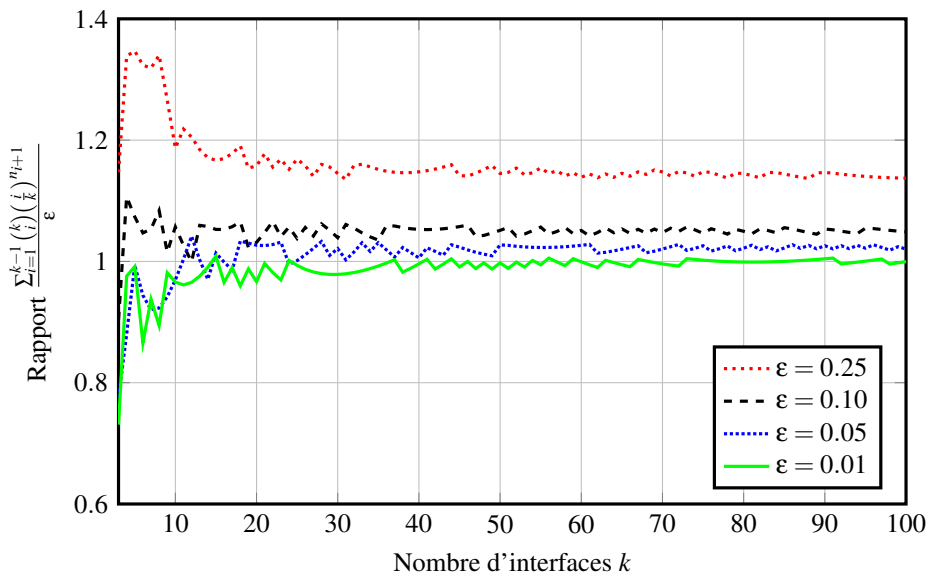


FIGURE 2: Rapport entre la borne supérieure d'erreur donnée par l'équation (4) et la précision cible  $\epsilon$ .

être arbitrairement grand. Il est cependant possible d'adapter les résultats précédents si l'on accepte de ne découvrir que les interfaces importantes. Cette *importance* peut être définie de deux manières, absolue et relative, ce qui donne les deux résultats suivants.

**Lemme 2** (Importance absolue). Soit  $0 < \alpha < 1$  un paramètre d'importance absolue et  $0 < \epsilon < 1$  une précision. Au bout de  $n_\alpha = \left\lceil \frac{\log(\frac{\epsilon}{1-\alpha})}{\log(1-\alpha)} \right\rceil$  sondes, la probabilité d'avoir découvert toutes les interfaces dont la probabilité de sélection est supérieure ou égale à  $\alpha$  est supérieure à  $1 - \epsilon$ .

**Théorème 2** (Importance relative). Soit  $0 < \alpha < 1$  un paramètre d'importance relative et  $0 < \epsilon < 1$  une précision. Pour tout  $k \geq 2$ , on choisit pour  $n_k$  le plus petit entier tel que  $(k-1)(1 - \frac{1}{k-1+\alpha})^{n_k} + (1 - \frac{\alpha}{k-1+\alpha})^{n_k} < \epsilon$ . La probabilité que l'algorithme MDA ne trouve pas toutes les interfaces dont la probabilité de sélection est au moins une fraction  $\alpha$  de la probabilité de sélection de l'interface la plus probable est inférieure à  $\epsilon + O(\epsilon^{\log_2(3)})$ .

Les preuves sont omises car relativement longues, mais il s'agit essentiellement d'adapter les résultats obtenus pour le cas uniforme en considérant les pires cas possibles pour une importance donnée.

## 5 Conclusion

Nous avons présenté des formules closes permettant de dimensionner le nombre de sondes à envoyer pour détecter toutes les interfaces d'un routeur pratiquant un équilibrage de charge uniforme. Ces formules ont l'avantage d'éviter un calcul complexe de l'espace des états possibles et peuvent se généraliser à un équilibrage non-uniforme. Nous espérons qu'elles pourront permettre d'améliorer les algorithmes de cartographie d'Internet.

## Références

- [AFT07] Brice Augustin, Timur Friedman, and Renata Teixeira. Multipath tracing with Paris traceroute. In *E2EMON 2007 - 5th IEEE/IFIP Workshop on End-to-End Monitoring Techniques and Services*, pages 1–8. IEEE, May 2007.
- [VATF09] Darryl Veitch, Brice Augustin, Renata Teixeira, and Timur Friedman. Failure Control in Multipath Route Tracing. In *IEEE Infocom*, pages 1395–1403. IEEE, April 2009.