



# Placement bio-inspiré de l'information dans les réseaux pair-à-pair

Hugo Pommier, Benoît Romito, François Bourdon

► **To cite this version:**

Hugo Pommier, Benoît Romito, François Bourdon. Placement bio-inspiré de l'information dans les réseaux pair-à-pair. Giroire, Frédéric and Mazauric, Dorian. JDIR, 2010, Sophia Antipolis, France. 2010. <inria-00468032>

**HAL Id: inria-00468032**

**<https://hal.inria.fr/inria-00468032>**

Submitted on 29 Mar 2010

**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.

# Placement bio-inspiré de l'information dans les réseaux pair-à-pair

Hugo Pommier et Benoît Romito et François Bourdon

GREYC UMR 6072, Université de Caen, Bd. du maréchal Juin, 14032 Caen CEDEX, France

Email : {hpommier,bromito,fbourdon}@info.unicaen.fr

**Résumé**—Dans cet article, nous présentons les bénéfices apportés par l'utilisation d'un système multi-agents pour gérer le placement de l'information dans une application de stockage de données décentralisée. Dans notre modèle, après une étape de fragmentation, chaque pièce d'information est associée à un agent mobile capable de prendre ses propres décisions. Pour contrôler le placement des agents dans un réseau de type pair-à-pair, nous appliquons des règles de flocking.

## I INTRODUCTION

Le stockage de données dans un réseau pair-à-pair nécessite que l'information soit robuste et persistante. Pour y parvenir, il est classique d'utiliser les codes d'effacement [1] qui découpent la donnée en fragments. Il vient ensuite le problème du placement de ces fragments dans le réseau [2]. Deux approches se dégagent. La première consiste à propager aléatoirement les morceaux d'information. Elle permet de limiter le coût de reconstruction en termes de bande passante, ainsi que l'effet goulot d'étranglement sur un pair donné. Cependant, la localisation des fragments reste coûteuse si elle n'est pas centralisée [3]. Une autre approche consiste à placer les fragments sur l'ensemble des voisins d'un pair  $p$  choisi [4]. Si  $p$  est en situation d'échec, ses voisins peuvent alors participer à la reconstruction de la donnée. Mais peu de pairs distincts peuvent participer durant une phase de reconstruction. Cette approche, généralement construite au-dessus d'une DHT (Distributed Hash Table) offre néanmoins de bonnes performances en termes de routage et de localisation.

Notre modèle essaye de prendre en compte les avantages de ces deux schémas. Nous associons à chaque morceau de donnée un agent mobile inter-agissant avec son environnement et nous régissons les déplacements de nos agents avec des règles de flocking inspirées du déplacement en nuée des oiseaux pour garder une cohésion locale entre eux. Ainsi, notre nuée de fragments peut se déplacer aléatoirement dans le réseau lorsque cela s'avère nécessaire tout en préservant un haut degré de proximité entre les agents.

## II LA COUCHE RÉSEAU PAIR-À-PAIR

La couche réseau logique est construite à l'aide de SCAMP [5], un protocole totalement décentralisé pour la gestion de groupes multicasts. Ce protocole a été initialement conçu pour supporter la diffusion fiable de l'information multicast à l'aide de protocoles épidémiques. Nous l'utilisons dans nos travaux pour ses bonnes propriétés en terme de tolérance aux fautes et de passage à l'échelle qui sont indispensables pour assurer la

sûreté de fonctionnement requise au niveau réseau. SCAMP construit un graphe aléatoire orienté suivant le modèle de Erdős et Rényi [6] et possédant un degré moyen qui converge vers  $(c+1)\log(n)$  avec  $c$  un paramètre du système permettant de moduler la redondance d'arcs. Cette propriété permet au graphe de rester connecté si la probabilité de perdre un arc est inférieure à  $\frac{c}{c+1}$ .

## III LA COUCHE AGENTS MOBILES

Nous utilisons des règles de flocking similaires à celles proposées par Reynolds [7], inspirées par le mouvements des oiseaux. Sa motivation a été de trouver des règles simples, facilement applicables à chaque agent, pour reproduire le vol en formation des oiseaux. Reynolds en a identifié trois :

- 1) **Cohésion** : les agents veillent à ne pas être trop éloignés de leur voisinage.
- 2) **Séparation** : les agents veillent à ne pas être trop proches de leur voisinage.
- 3) **Alignement** : les agents gardent la même direction que leur voisinage.

Dans ce modèle, le flocking n'est pas propre à chaque agent mais est un comportement émergent des interactions entre les membres du groupe. En associant les règles de Reynolds avec une distance maximale  $d$  entre deux éléments dans un réseau pair-à-pair estimée par le RTT (Round Trip Time), nous pouvons reproduire un comportement de nuée entre les fragments. Nous avons défini les règles suivantes :

- 1) Un pair ne peut stocker qu'un seul agent par document (règle de séparation);
- 2) Les agents se déplacent dans le but de se rapprocher des agents les plus éloignés de leur voisinage (règle de cohésion associée à une distance obtenue par RTT);
- 3)  $\lambda$  est la distance maximale entre deux fragments. Au-dessus de cette valeur, la règle de cohésion est violée.

L'algorithme 1 permet à un agent de se déplacer suivant les règles de flocking. L'algorithme consiste pour un agent à choisir un pair dans son voisinage qui ne viole pas les règles énoncées. A chaque fois qu'un agent se déplace sur un arc, il dépose des phéromones qui s'évaporent périodiquement. Lorsqu'un agent a le choix entre plusieurs destinations, il choisit celle qui a le taux de phéromones le plus bas. Ce mécanisme simple permet de répartir les déplacements sur l'ensemble du réseau.

---

**Algorithme 1** : Déplacement d'un fragment d'un pair  $x$ 

---

**Entrées** : Un pair  $x$ ,  $doc$  un document,  $f_{doc}$  un fragment  $\in doc$ ,  $V_\alpha$  le voisinage d'un pair  $\alpha$ ,  $\phi_\alpha$  le niveau de phéromones d'un pair  $\alpha$

**Sorties** : Un pair  $p$  où stocker  $f_{doc}$

$Occupes \leftarrow$  pairs  $\in V_x$  possédant des fragments  $\in doc$ ;

**pour chaque**  $y \in Occupes$  **faire**

**si**  $d(x, y) < \lambda$  **alors**  
        Retirer  $y \in Occupes$   
        (violation de la règle de cohésion);

$Libres \leftarrow$  pairs  $\in V_x$  possédant des fragments  $\notin doc$ ;

**pour chaque**  $y \in Occupes$  **et**  $z \in Libres$  **et**  $y \in V_z$  **faire**

**si**  $d(y, z) > \lambda$  **alors**  
        Retirer  $z \in Libres$   
        (application de la règle de cohésion);

Choisir un pair  $p \in Libres$  avec  $\phi_p$  minimum;

Déplacer  $f_{doc}$  sur  $p$  et incrémenter  $\phi_p$ ;

---

#### IV RÉSULTATS EXPÉRIMENTAUX

Nous avons déployé des agents mobiles, implémentés avec le framework JavAct<sup>1</sup>, dans un réseau de 100 machines construit avec SCAMP. La constante  $c$  de SCAMP a été fixée à 3. Chaque agent dépose 2 unités de phéromones à chaque déplacement et le taux d'évaporation a été fixé à 0,1% toutes les secondes. La figure 1 mesure la cohésion de plusieurs nuées au cours du temps. C'est à dire le nombre de fragments qui possèdent au moins un fragment du même fichier sur un pair voisin au cours du temps. On constate que les fragments des nuées de 10 et 20 fragments restent groupées dans le temps. La figure 2 montre le nombre de machines distinctes ayant été visitées au cours du temps. On voit clairement les bénéfices d'utiliser des phéromones qui permettent d'explorer la quasi-totalité du réseau.

#### V PERSPECTIVES ET CONCLUSION

Nous avons présenté les bases d'une nouvelle approche pour traiter le problème du placement et de la sûreté de l'information dans les systèmes de stockage décentralisés en utilisant des agents mobiles se déplaçant en nuée dans un réseau pair-à-pair. Les premiers résultats montrent que la cohésion de la nuée est préservée et que l'utilisation de phéromones par les agents mobiles permet une couverture du réseau quasi totale. Dans un réseau pair-à-pair, les fautes corrélées rendent indisponibles plusieurs pairs simultanément posant un problème si ces pairs sont responsables d'informations cruciales. Après une étape de clustering pour trouver un modèle de fautes corrélées et une phase d'observation par le déplacement, nous souhaitons trouver un positionnement de nuée minimisant le nombre de fragments sur des pairs corrélés. L'intérêt de la mobilité permettrait alors de repositionner la nuée à chaque fois que les changements dans le réseau l'imposent.

<sup>1</sup><http://www.javact.org>

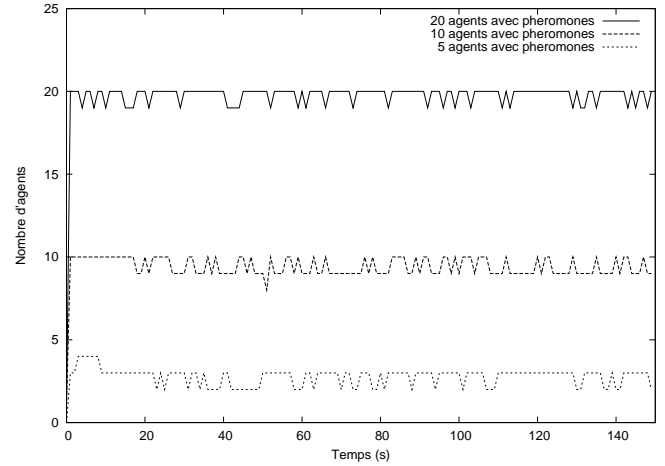


FIG. 1. Degré de cohésion de nuées de 5, 10, et 20 agents avec phéromones dans un réseau de 100 machines.

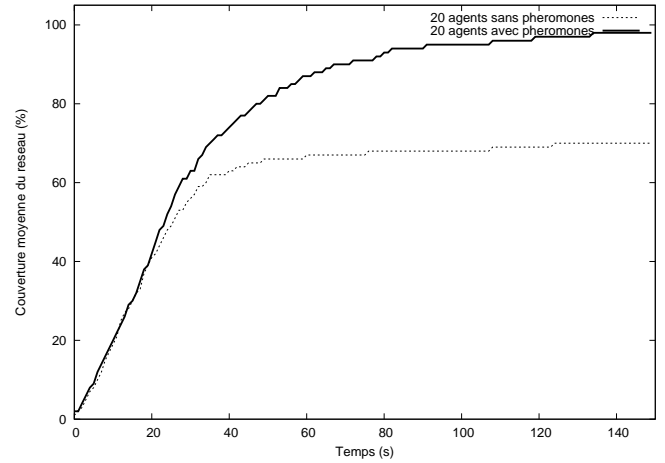


FIG. 2. Couverture moyenne du réseau pour une nuée de 20 fragments dans un réseau de 100 machines.

#### RÉFÉRENCES

- [1] H. Weatherspoon and J. Kubiatowicz, "Erasure coding vs. replication : A quantitative comparison," in *IPTPS '01 : Revised Papers from the First International Workshop on Peer-to-Peer Systems*. London, UK : Springer-Verlag, 2002, pp. 328–338.
- [2] F. Giroire, J. Monteiro, and S. Pérennes, "P2P storage systems : How much locality can they tolerate?" in *Proceedings of the 34th IEEE Conference on Local Computer Networks (LCN)*, 2009, pp. 320–323.
- [3] S. Ghemawat, H. Gobioff, and S.-T. Leung, "The google file system," *SIGOPS Oper. Syst. Rev.*, vol. 37, no. 5, pp. 29–43, 2003.
- [4] A. Haeberlen, A. Mislove, and P. Druschel, "Glacier : highly durable, decentralized storage despite massive correlated failures," in *NSDI'05 : Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation*. Berkeley, CA, USA : USENIX Association, 2005, pp. 143–158.
- [5] A. J. Ganesh, A.-M. Kermarrec, and L. Massoulié, "Scamp : Peer-to-peer lightweight membership service for large-scale group communication," in *Networked Group Communication*, 2001, pp. 44–55.
- [6] P. Erdős and A. Rényi, "On the evolution of random graphs," in *Publications of the Mathematical Institute of the Hungarian Academy of Sciences*, 1960, pp. 17–61.
- [7] C. W. Reynolds, "Flocks, herds and schools : A distributed behavioral model," in *SIGGRAPH '87 : Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, 1987, pp. 25–34.