



**HAL**  
open science

## Exploration de graphes anonymes avec des jumelles

Jérémie Chalopin, Emmanuel Godard, Antoine Naudin

► **To cite this version:**

Jérémie Chalopin, Emmanuel Godard, Antoine Naudin. Exploration de graphes anonymes avec des jumelles. ALGOTEL 2015 - 17èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications, Jun 2015, Beaune, France. hal-01148649

**HAL Id: hal-01148649**

**<https://hal.science/hal-01148649v1>**

Submitted on 5 May 2015

**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.

# Exploration de graphes anonymes avec des jumelles

Jérémy Chalopin, Emmanuel Godard and Antoine Naudin

LIF, Université Aix-Marseille and CNRS, FRANCE

---

Nous nous intéressons à l'exploration de graphes par un agent mobile. Il est connu que sans information "globale" sur le graphe, un agent peut explorer et s'arrêter uniquement si le graphe est un arbre. C'est pourquoi nous équipons l'agent de "jumelles" lui permettant de voir le graphe induit par les sommets voisins du sommet courant. Dans cette étude, nous montrons que cette information locale donnée par les jumelles permet à l'agent d'explorer avec arrêt une plus large famille de graphes que les arbres. Nous prouvons la condition nécessaire suivante : un graphe est explorable seulement si son complexe de clique admet un revêtement universel fini. Puis, nous proposons un algorithme d'exploration universel pour tout graphe satisfaisant cette condition, prouvant ainsi que cette famille, appelée  $\mathcal{FC}$ , est maximum pour l'exploration avec arrêt. Pour finir, nous montrons qu'il n'existe pas de fonction calculable pouvant borner la complexité de tout algorithme d'exploration de  $\mathcal{FC}$ .

**Mots clés :** Agent mobile, exploration de graphes, Graphes anonymes, Revêtement Universel, Simplement Connexe.

---

## 1 Introduction

### 1.1 Le Modèle

Nous utilisons le modèle standard d'agent mobile (voir [Das13]). Un agent mobile est une entité mobile naviguant dans un graphe  $G = (V, E)$  de sommet en sommet via les arêtes. Chaque sommet  $v \in V(G)$  dispose d'une étiquette  $label(v)$ . De plus, les graphes sont anonymes, c.à.d., l'étiquette  $label(v)$  ne permet pas d'identifier formellement le sommet  $v$  de manière unique. Afin de permettre à l'agent de se déplacer dans le graphe, les sommets sont dotés d'un étiquetage des ports localement injectif  $\delta = \{\delta_v\}_{v \in V(G)}$  donnant pour un sommet  $v \in V(G)$ , une identité unique à chacun de ses voisins. Soit  $\delta_u(v)$ , le numéro de port en  $u$  désignant l'arête  $uv$ . Afin de pouvoir "revenir sur ces pas", l'agent connaît le numéro de port par lequel il accède au sommet courant.

Nous notons  $(G, \delta)$  un graphe  $G$  doté d'un étiquetage des ports  $\delta$ . Pour avoir des algorithmes robustes aux différents étiquetages des ports pour un même graphe, nous notons  $\mathcal{G}^\delta$  la famille contenant tous les graphes dotés d'un étiquetage des ports valide. Comme les numéros de ports sont invariant durant une exécution, nous noterons  $G$  pour désigner un graphe  $(G, \delta) \in \mathcal{G}^\delta$ . Un agent visitant un sommet du graphe peut utiliser des jumelles lui permettant de "voir" le graphe induit par les sommets adjacents au sommet visité ainsi que les numéros de ports présents dans ce graphe. Nous supposons que chaque sommet du graphe possède une étiquette encodant cette boule de rayon 1 autour du sommet. L'étiquette est disponible lorsque un agent visite le sommet. Comme seules ces contraintes sont spécifiques au modèle, nous pourrions ainsi réutiliser les techniques et algorithmes classiques sur des graphes étiquetés. Il est assez évident de voir que cet encodage par étiquettes est équivalent au modèle avec jumelles. En effet, les jumelles donnant juste une information locale, elles n'augmentent pas les possibilités de mouvements de l'agent. Nous appellerons cet étiquetage des sommets *étiquetage de jumelles*.

### 1.2 Exploration de graphes anonymes

Un algorithme  $\mathcal{A}$  est un *algorithme d'Exploration* si pour toute exécution sur un graphe  $G \in \mathcal{G}^\delta$  avec étiquetage de jumelles et pour tout sommet de départ  $v_0 \in V(G)$ , l'agent explore  $G$  et s'arrête, ou l'agent ne s'arrête jamais (exploration perpétuelle). Un graphe  $G$  est *explorable* s'il existe un

algorithme d'Exploration qui, pour tout sommet de d epart, explore  $G$  et s'arr ete. Un algorithme  $\mathcal{A}$  explore une famille de graphes  $\mathcal{F}$  si  $\mathcal{A}$  est un algorithme d'Exploration tel que pour tout  $G \in \mathcal{F}$ ,  $\mathcal{A}$  s'arr ete et pour tout  $G \notin \mathcal{F}$ ,  $\mathcal{A}$  explore  $G$  et s'arr ete ou ne s'arr ete jamais.

### 1.3 Nos R esultats

5 Nous nous int eressons   la calculabilit  distribu e du mod le. Tout d'abord, en s'appuyant sur un lemme standard de rel evement [Ang80], nous donnons une condition n ecessaire sur les graphes explorables avec jumelles. Une borne inf erieure sur le nombre de d placements requis par l'agent est aussi donn ee.

Nous montrons ensuite que cette condition caract erise exactement la famille maximum des graphes explorables. Cette famille maximum, not ee  $\mathcal{FC}$ , contient les graphes ayant un complexe de clique poss edant un rev etement universel fini.

Un algorithme d'Exploration est ainsi propos e pour  $\mathcal{FC}$ . Comme  $\mathcal{FC}$  est maximum, notre algorithme explore donc tous les graphes explorables. Notons que  $\mathcal{FC}$  est bien plus vaste que les arbres; elle contient entre autre tous les graphes ayant un complexe de clique simplement connexe (comme les graphes triangul es, les triangulations planaires ou encore les triangulations du plan projectif). Une version longue de ces r esultats est pr esent ee dans [CGN15].

## 2 D efinitions et Notations

**Graphes.** Nous consid erons des graphes sans boucle, ni ar ete multiple. Soit  $p = (v_0, \dots, v_k)$  un chemin dans un graphe  $G$  munis d'un  tiquetage des ports  $\delta$ . L' tiquetage induit par  $p \subset G$ , not e  $\lambda(p) = \{\delta_{v_i}(v_{i+1}) \mid 1 \leq i < k\}$ , est la s quence de num eros de ports suivi par  $p$  dans  $G$ . Nous notons par  $B_G(v, k)$ , le graphe induit par l'ensemble de sommets   distance au plus  $k$  de  $v$ . Nous supposons que l'  tiquette  $label(v)$  donn e au sommet  $v \in V(G)$  correspond   l' tiquetage de jumelles (vu plus haut), i.e.,  $label(v)$  encode un graphe isomorphe    $B_G(v, 1)$  (pr eservant l' tiquetage des ports). Nous pr esentons maintenant les rev etements, outils indispensables de cette  tude. Un morphisme  $\varphi : V(G) \rightarrow V(H)$  de  $G$  dans  $H$  est un *homomorphisme* si pour toute ar ete  $uv \in E(G)$ ,  $\varphi(u)\varphi(v) \in E(H)$ . Un homomorphisme  $\varphi : V(G) \rightarrow V(H)$  de  $G$  dans  $H$  est un *rev etement* si pour tout  $v \in V(G)$ ,  $\varphi|_{V(B_G(v, 1))}$  est une bijection entre  $B_G(v, 1)$  et  $B_H(\varphi(v), 1)$ . Ces d efinitions sont  tendues aux graphes  tiquet es  $(G, \delta, label)$ ,  $(H, \delta', label')$  avec les conditions suivantes :  $label(v) = label'(\varphi(v))$  et  $\forall uv \in E(G), \delta_u(v) = \delta'_{\varphi(u)}(\varphi(v))$ .

**Complexes Simpliciaux.** Les d efinitions suivantes sont standard [LS77].  tant donn e un ensemble  $V$ , un *simplexe* de dimension  $n$  est un sous-ensemble de  $V$     $n + 1$   l ements. Un *complexe simplicial*  $K$  est un ensemble de simplexes tel que pour tout simplexe  $s \in K$ ,  $s' \subseteq s$  implique que  $s' \in K$ . Un complexe simplicial est *connexe* si le graphe induit par les simplexes de dimension 0 (sommet) et 1 (ar ete) est connexe. Notons  $St(v, K)$  le sous-complexe de  $K$  contenant tout simplexe  $s$  de  $K$  contenant  $v$  et les sous-complexes induit par les sommets contenus dans  $s$ . Un *homomorphisme simplicial*  $\varphi : K \rightarrow K'$  de  $K$  dans  $K'$  est un morphisme tel que si  $s = \{v_0, \dots, v_k\} \in K$  est un simplexe, alors  $\{\varphi(v_0), \dots, \varphi(v_k)\}$  est aussi un simplexe dans  $K'$ . Pour finir, un homomorphisme simplicial  $\varphi : K \rightarrow K'$  est un *rev etement simplicial* si pour tout sommet  $v$  de  $V(K)$ ,  $\varphi|_{St(v, K)}$  est une bijection entre  $St(v, K)$  et  $St(\varphi(v), K')$ . Le *rev etement universel* d'un complexe  $K$ , not e  $\hat{K}$ , est un complexe (possiblement infini) tel que il existe un rev etement  $\varphi : \hat{K} \rightarrow K$  et tel que pour tout complexe  $K'$  et rev etement  $\varphi' : K' \rightarrow K$ , il existe un rev etement  $\varphi'' : \hat{K} \rightarrow K'$  et  $\varphi' \circ \varphi'' = \varphi$ . Le rev etement universel est unique   isomorphisme pr es. Le *complexe de clique* d'un graphe  $G$ , not e  $\mathcal{K}(G)$ , est le complexe simplicial form e par les cliques de  $G$ . La proposition suivante justifie l'utilisation des complexes de cliques.

**Proposition 2.1.** * tant donn e deux graphes  $G$  et  $H$  avec  tiquetage de jumelles et un morphisme  $\varphi : V(G) \rightarrow V(H)$ ,  $\varphi$  est un rev etement (pr eservant les  tiquettes) si et seulement si  $\varphi$  est un rev etement simplicial de  $\mathcal{K}(G)$  dans  $\mathcal{K}(H)$ .*

Nous noterons  $G$  pour d efinir soit un graphe avec  tiquetage de jumelles, soit son complexe de cliques. Deux chemins  $p, p'$  dans un graphe  $G$  sont dit * l ementairement homotope* si ils v erifient une des deux conditions suivantes,

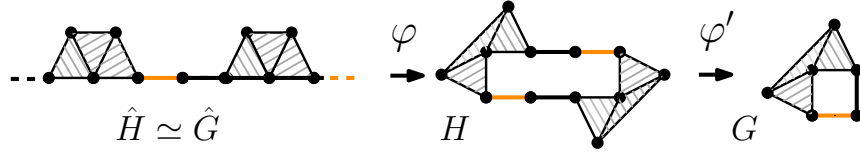


FIGURE 1: Exemples de revêtements simpliciaux.

- (i)  $p'$  peut être obtenu de  $p$  en insérant ou supprimant un sous-chemin  $(uvu)$  avec  $u, v \in V(G)$ .
- (ii)  $p$  et  $p'$  sont de la forme suivante :  $p = (v_0 \dots v_i q v_j \dots v_k)$  et  $p' = (v_0 \dots v_i q' v_j \dots v_k)$  où  $q$  et  $q'$  sont deux sous-chemins et  $q^{-1}q'$  est un triangle dans  $G$  ( $q$  ou  $q'$  peuvent être vides).

Un cycle  $c$  dans un graphe  $G$  est  $k$ -contractible s'il existe une séquence de  $k$  cycles  $\{c_1 \dots c_k\}$  tel que  $c_1 = c$ ,  $c_k = \{u\}$  avec  $u \in V(G)$  et pour tout  $1 \leq i < k$ ,  $c_i$  et  $c_{i+1}$  sont élémentairement homotopes. Un cycle est *contractible* s'il existe un  $k \in \mathbb{N}$  tel que  $c$  est  $k$ -contractible. Un complexe simplicial est *simplement connexe* si tous ses cycles sont contractibles. Ces complexes possèdent des propriétés combinatoires et topologiques intéressantes, mises en avant dans la proposition suivante.

**Proposition 2.2.** [LS77] Soit  $K'$  et  $K$  deux complexes, et un revêtement  $\varphi : K' \rightarrow K$ .  $K'$  est le revêtement universel de  $K$  si et seulement si  $K'$  est simplement connexe.

Nous montrerons par la suite qu'un agent ne peut distinguer entre deux graphes (avec étiquetages de jumelles) si il existe un revêtement de l'un vers l'autre (Lemme 3.1).

### 3 Résultat d'Impossibilité et Borne Inférieure

Dans le modèle classique, la seule famille maximale de graphes explorables est celles des arbres.

Nous adaptons donc cette classification aux graphes avec étiquetage de jumelles. Notons  $\mathcal{FC} = \{G \mid \text{le revêtement universel de } \mathcal{K}(G) \text{ est fini}\}$  et  $\mathcal{IC} = \{G \mid \text{le revêtement universel de } \mathcal{K}(G) \text{ est infini}\}$ . Remarquons que  $\mathcal{FC}$  possède une importante sous classe  $\mathcal{SC} = \{G \mid \mathcal{K}(G) \text{ est simplement connexe}\}$ . Les complexes représentés sur l'exemple de la Figure 1 appartiennent à  $\mathcal{IC}$ . Il y a deux revêtements simpliciaux représentés :  $\varphi$ , qui montre un revêtement depuis le revêtement universel (infini, à gauche) ; et  $\varphi'$  montrant que la taille d'un revêtement (au centre) est toujours un multiple de la taille du complexe revêtu (à droite), ici le double. Étant donné un graphe  $G$  et un algorithme d'Exploration  $\mathcal{A}$ , nous noterons  $(\Lambda_G^i, \text{pos}_G^i)$  l'état de l'agent à l'étape  $i \in \mathbb{N}$  de l'exécution où  $\Lambda_G^i$  est la mémoire de l'agent et  $\text{pos}_G^i$  est sa position courante dans  $V(G)$ . Le preuve du lemme suivant est standard (voir [YK96]).

**Lemme 3.1** (Lemme de relèvement). Soit  $\mathcal{A}$  un algorithme d'Exploration,  $G, H$  deux graphes avec étiquetage de jumelles et un revêtement  $\varphi : G \rightarrow H$ . Pour tout sommet de départ  $u \in V(G)$  et  $v \in V(H)$  tel que  $\varphi(u) = v$ , les exécutions de  $\mathcal{A}$  dans  $G$  et  $H$  vérifient que  $\forall i \in \mathbb{N}$ ,  $(\Lambda_G^i, \text{pos}_G^i) = (\Lambda_H^i, \text{pos}_H^i)$ .

Soit deux graphes  $G$  et  $H$  tels que durant les  $k$  premières étapes d'une exécution, un agent ne peut pas faire de distinction entre  $G$  et  $H$  (il suivra donc une même séquence de ports sur ses  $k$  premiers déplacements). Il s'arrêtera donc dans  $G$  en moins de  $k$  déplacements si et seulement s'il s'arrête dans  $H$  en moins de  $k$  déplacements. Donc, si  $|V(G)| > k + 1$  et que l'agent explore  $H$  en moins de  $k$  étapes, l'agent ne pourra pas explorer  $G$  avec le même algorithme. Par conséquent, un agent explorant un graphe  $G \in \mathcal{FC}$  devra donc se déplacer au moins  $|V(\hat{G})|$  fois pour être certain de ne pas s'arrêter trop tôt dans le revêtement universel  $\hat{G}$  du complexe  $G$  et ainsi respecter la spécification. La taille du revêtement universel est donc une borne inférieure pour l'Exploration d'un graphe  $G \in \mathcal{FC}$ . De plus, pour tout graphe  $G \in \mathcal{IC}$ , pour tout  $k \in \mathbb{N}$ , il existe un graphe  $H$  tel qu'un agent ne peut pas distinguer  $G$  de  $H$  en moins de  $k$  étapes. Par conséquent, aucun graphe  $G \in \mathcal{IC}$  n'est explorable. Nous obtenons le théorème suivant,

**Théorème 1.** Si un graphe  $G$  est explorable par un algorithme  $\mathcal{A}$  alors  $G \in \mathcal{FC}$ . De plus, le nombre de mouvements de l'agent sera supérieur à la taille du revêtement universel de  $G$ .

---

**Algorithm 1:** Exploration de  $\mathcal{FC}$

---

```

1 begin
2    $k = 0$ ;
3   repeat
4     Incr ementer  $k$  ;
5     Explorer   distance  $2k$  et calculer  $\mathcal{T}(v_0, 2k)$ ;
6     Chercher un graphe  $H$  tel que  $|V(H)| < k$ ,  $\exists \tilde{v}_0 \in V(H)$  et  $\tilde{v}_0 \sim_{2k} v_0$ ;
7   until  $H$  existe et tous ses cycles simples sont  $k$ -contractible;
8   Arr eter l'exploration;
```

---

## 4 Exploration de $\mathcal{FC}$

Nous proposons un algorithme d'Exploration pour la famille  $\mathcal{FC}$ . Le but de l'algorithme est de visiter une boule suffisamment grande avant de stopper l'agent. Pour d etecter quand arr eter l'agent, nous utilisons les *vues* des sommets [YK96], de fa on standard. La *vue* d'un sommet  $v$    distance  $k$  dans un graphe  $G$ , not   $\mathcal{T}(v, k)$ , contient tous les chemins possibles (avec tous les  tiquetages associ s) de longueur au plus  $k$  issus de  $v$  dans  $G$ . Nous notons  $v \sim_k v'$  si  $\mathcal{T}(v, k) = \mathcal{T}(v', k)$ . Pour prouver la correction de cet algorithme sur un graphe  $G \in \mathcal{FC}$ , nous prouvons que le graphe  $H$  trouv  par l'algorithme est le rev tement universel de  $G$ . Comme  $|V(G)| \leq |V(\hat{G})|$ ,  $G$  est bien inclus dans la boule explor e par l'agent. Notons que tester la  $k$ -contractibilit  des cycles simples est calculable et prouve la simple connexit  si le test r ussit. Pour prouver le th or me, nous montrons qu'un complexe simplement connexe ne ressemble localement   aucune sous-partie *stricte* d'un autre complexe. Pour cela, nous montrons que pour tous chemins  $p, q$  partant de  $\tilde{v}_0$  et menant   un m me sommet  $\tilde{v}$  de  $H$ , les deux chemins de  $G$  partant de  $v_0$  et  tiquet s  $\lambda(p)$  et  $\lambda(q)$  m nent   un m me sommet  $v$  de  $G$  tel que  $label(\tilde{v}) = label(v)$ . Ainsi, Nous pouvons d finir un homomorphisme  $\varphi : H \rightarrow G$  comme suit. Pour tout sommet  $\tilde{v} \in V(H)$ , on choisit un chemin  $\tilde{p}$  allant de  $\tilde{v}_0$     $\tilde{v}$  dans  $H$ . L'image de  $\tilde{v}$ ,  $\varphi(\tilde{v})$ , est le sommet  $v$  atteint par le chemin  tiquet   $\lambda(\tilde{p})$  depuis  $v_0$  dans  $G$ . Nous montrons ensuite que  $\varphi$  est un rev tement. Comme  $\mathcal{FC}$  est explorable (Correction *Algorithm1*) et que par le Th or me 1,  $\mathcal{IC}$  ne l'est pas, nous obtenons le th or me suivant.

**Th or me 2.**  $\mathcal{FC}$  est maximum pour l'exploration de graphes avec  tiquetage de jumelles.

20 Pour finir, nous montrons un r sultat de complexit  pour tout algorithme d'Exploration de  $\mathcal{FC}$ .

**Th or me 3.** Pour tout algorithme d'Exploration  $\mathcal{A}$  pour  $\mathcal{FC}$ , il n'existe pas de fonction calculable  $\tau : \mathbb{N} \rightarrow \mathbb{N}$  tel que pour tout graphe  $G \in \mathcal{FC}$ ,  $\mathcal{A}$  ex cute au plus  $\tau(|V(G)|)$  mouvements avant arr t.

25 Pour prouver ce th or me, nous proc dons par l'absurde en proposant une proc dure reposant sur cette fonction  $\tau$  (suppos e calculable) et simulant l'algorithme  $\mathcal{A}$  pendant  $\tau(|G|)$   tapes. Cette proc dure permet ainsi de d cider si un complexe simplicial est simplement connexe ou non. Ce probl me  tant ind cidable [Hak73], nous obtenons donc le r sultat.

## R f rences

- [Ang80] D. Angluin. Local and global properties in networks of processors. *Proceedings of the 12th Symposium on Theory of Computing*, pages 82–93, 1980.
- 30 [CGN15] J eremie Chalopin, Emmanuel Godard, and Antoine Naudin. Anonymous graph exploration with binoculars. Technical report, LIF, Universit  Aix-Marseille and CNRS, May 2015. arXiv : 1505.00599.
- [Das13] S. Das. Mobile agents in distributed computing : Network exploration. *Bulletin of EATCS*, 1(109), 2013.
- [Hak73] W. Haken. Connections between topological and group theoretical decision problems. In *Word Problems Decision Problems and the Burnside Problem in Group Theory*, volume 71 of *Studies in Logic and the Foundations of Mathematics*, pages 427–441. North-Holland, 1973.
- 35 [LS77] R.C. Lyndon and P.E. Schupp. *Combinatorial group theory*. Ergebnisse der Mathematik und ihrer Grenzgebiete. Springer-Verlag, 1977.
- [YK96] M. Yamashita and T. Kameda. Computing on anonymous networks : Part i - characterizing the solvable cases. *IEEE TPDS*, 7(1) :69–89, 1996.