Détection de communautés recouvrantes orientée sommet

Maël Canu, Marie-Jeanne Lesot, Adrien Revault d'Allonnes

Détection de communautés recouvrantes orientée sommet

Maël CANU [1,2], **Marie-Jeanne LESOT** [1,2],
Adrien REVAULT D'ALLONNES [3]

1. *Sorbonne Universités, UPMC Univ Paris 06, UMR 7606, LIP6, F-75005, Paris, France*
   <first_name>.<last_name>@lip6.fr
2. *CNRS, UMR 7606, LIP6, F-75005, Paris, France*

3. *Université Paris 8, EA 4383, LIASD, F-93526, Saint-Denis, France*
   allonnes@ai.univ-paris8.fr

RÉSUMÉ. *Les sommets multi-appartenants constituent une caractéristique importante à prendre en compte lors de la conception d'une méthode de détection de communautés. Nous proposons dans cet article LOCNeSs, un algorithme de détection utilisant une approche orientée sommet. LOCNeSs permet une implémentation complètement décentralisée et limite la propagation, deux caractéristiques utiles pour une utilisation dans un environnement décentralisée ou très distribué. L'algorithme modélise des préférences entre sommets, basées sur l'attachement préférentiel, afin d'aggréger ces sommets pour former des communautés. Une étude expérimentale permet de montrer que LOCNeSs identifie les sommets multi-appartenants de façon exhaustive et pertinente.*

ABSTRACT. *Overlapping vertices are an important characteristic to consider when designing a community detection method. In this article, we propose LOCNeSs, a detection algorithm using a vertex-oriented approach. It allows a decentralised implementation limiting information propagation in the graph, which is suitable to be used in a heavily decentralised or distributed environment. The algorithm models preference between vertices using a measure based on preferential attachment, and then aggregates these vertices to from communities. An experimental study shows that LOCNeSs accurately identifies relevant overlapping vertices.*

MOTS-CLÉS : *exploration de graphes, exploitation de graphes, détection de communautés, méthodes orientées sommets, méthodes décentralisées*

KEYWORDS: *graph mining, graph exploitation, community detection, oriented vertex method, decentralised method*

1. Introduction

La détection de communautés est une tâche largement répandue dans le domaine de l'analyse automatique des graphes, elle consiste à identifier automatiquement dans un réseau les sous-ensembles denses répondant à une définition donnée, mais non universelle (Fortunato, 2009). Ces sous-ensembles peuvent être *disjoints*, c'est-à-dire que chaque sommet n'appartient qu'à une seule communauté, ou *recouvrants*, autorisant l'appartenance d'un même sommet à plusieurs communautés simultanément.

Identifier des zones recouvrantes (« frontières ») est utile, le fait d'appartenir à plusieurs communautés dénotant en général un rôle spécifique et remarquable, par exemple un relais, une interface ou un médiateur entre ces communautés : dans un réseau de co-publications cela peut être des chercheurs effectuant des travaux pluridisciplinaires, dans un réseau de communication cela peut être un point de peering (échange de traffic). Ce rôle spécifique est en cela minoritaire, en effet tous les sommets dans un graphe de terrain ne peuvent pas l'endosser : un sommet situé au coeur d'une communauté interagit en son sein, sa mono-appartenance ne présente aucun doute. La présence excessive de sommets multi-appartenants dans un graphe amènerait les communautés à être « mélangées » entre elles de façon importante, l'intérêt et la pertinence de leur identification en serait amoindris.

La méthode proposée, **LOCNeSs** (**L**ocating **O**verlapping **C**ommunities in **Ne**twork **S**tructure**s** est une méthode de détection de communautés identifiant les sommets multi-appartenant par une approche orientée sommet, elle a été conçue dans le but de permettre une implémentation décentralisée. Cela lui confère un avantage non négligeable pour, par exemple, être utilisée dans les réseaux mobiles ad-hoc (MANET) ou simplement pour être exécutée facilement dans un environnement de calcul distribué. L'organisation de cet article est la suivante : la section 2 dresse un état de l'art de la détection de communautés, disjointes et recouvrantes, et en particulier des méthodes orientées sommet, la section 3 décrit l'algorithme LOCNeSs en détails, la section 4 présente quatre expériences visant à valider l'approche proposée et la section 5 conclut l'exposé du travail.

2. Etat de l'art

2.1. Détection de communautés disjointes

Il existe de nombreuses méthodes de détection de communautés (Fortunato, 2009). Elles diffèrent sur un certain nombre d'aspects tels que l'approche utilisée pour le traitement du graphe (locale ou globale) ou la définition de la communauté (maximisation d'un critère de qualité, agglomération de sommets selon des règles...) Il existe entre autres des approches par optimisation de fonction, analyse du spectre du graphe, modèles statistiques, markoviens, ou basés sur la convergence de marches aléatoires (Fortunato, 2009 ; Bedi, Sharma, 2016). Nous détaillons ici deux approches principales liées à la méthode proposée : la propagation de labels et les méthodes orientées sommet. En effet, ces deux familles permettent une détection locale et décentralisée.

Les méthodes présentées forment des communautés disjointes, et ne sont donc pas comparables à la méthode LOCNeSs présentée dans cet article.

2.1.1. Propagation de labels

Cette famille a vu de nombreuses méthodes apparaître ces dernières années (Raghavan *et al.*, 2007), offrant de très bonnes performances. Le principe de base consiste à affecter un identifiant unique à chaque sommet (label), qui propagera ensuite cet identifiant, ainsi que les identifiants qu'il a lui-même précédemment reçus, à tous ses voisins jusqu'à stabilisation (pas de nouveau label reçu). Chaque sommet retient finalement l'identifiant qu'il a le plus reçu, et tous les sommets ayant retenu le même identifiant sont placés dans la même communauté. Les principaux inconvénients de cette méthode sont la propagation intensive, chaque identifiant étant propagé au moins une, souvent plusieurs fois à travers tout le graphe, ainsi que l'incertitude quant à la durée de la stabilisation de la propagation (Tian *et al.*, 2013).

2.1.2. Orientée sommet

Les méthodes orientées sommet reposent sur l'utilisation de certains sommets comme *agents*, *graines* ou encore *leaders*, comme base pour effectuer une détection locale de communautés. Chaque communauté comprend un leader et les sommets de son voisinage qui sont jugés proches, ou préférés, nommés followers. Ce type de communauté est parfois désigné comme *ego-centrée*. Bien que les méthodes orientées graines et leaders soient légèrement différentes, nous utiliserons les termes *leader* et *centré-leader* pour la description de ces deux familles.

Une première étude (Andersen, Lang, 2006) a montré la viabilité de l'approche, et des études ultérieures ont apporté une preuve théorique liant la présence d'un fort coefficient de clustering local, généralement présent dans les communautés graphes de terrain, avec la possibilité d'obtenir des sous-graphes de bonne conductance par exploration du voisinage. Des méthodes basées sur l'expansion d'une communauté à partir d'un leader, telles que *Top-Leaders* (Rabbany *et al.*, 2010), considérent les préférences potentielles d'un follower pour différents leaders. Les résultats sont améliorés itérativement à la manière d'un algorithme k-moyennes jusqu'à convergence. *Leader-Follower* (Shah, Zaman, 2010) définit chaque communauté comme une clique, ce qui est un postulat très fort, généralement non vérifié dans les réseaux réels.

La méthode de (Danisch *et al.*, 2013) scrute les variations des valeurs de mesures de proximités en un leader et les sommets de son voisinnage, lorsque ces valeurs chutent elle estime que les limites de la communauté ont été atteintes.

YASCA (Kanawati, 2014) étend les communautés autour des leaders de façon gloutonne et les fusionne en utilisant du clustering d'ensemble. LICOD (Yakoubi, Kanawati, 2014) débute par une sélection fine des leaders avant de calculer un classement des meilleures appartenances aux communautés pour chaque follower, les préférences et appartenances étant mises à jour jusqu'à stabilisation en utilisant des stratégies basées sur la théorie du choix social.

Canu *et al.* (2015) considèrent chaque sommet comme un leader potentiel et utilisent des mesures de préférences entre sommets afin de constituer des interpréférences entre sommets. Les communautés sont les groupes disjoints de sommets liés par ces interpréférences. Il est à noter dans ce cas que la communautée n'est pas ego-centrée au sens propre : un ou plusieurs leaders sont au coeur de toutes les interpréférences.

2.2. Détection de communautés recouvrantes

Les sommets multi-appartenants étant une caractéristique fréquente des réseaux complexes, de nombreuses méthodes de détection de communautés les prenant en comptent existent (Xie *et al.*, 2013). Si beaucoup sont adaptées de méthodes disjointes, des méthodes originales ont également été proposées, par exemple la recherche avec expansion ou percolation de k-cliques, ou des solutions basées sur la théorie des jeux. Bien que la plupart de ces méthodes identifient correctement les communautés, par rapport à une vérité terrain, elles manquent souvent de précision ou de pertinence concernant les sommets multi-appartenants (Xie *et al.*, 2013). Nous présentons ici quelques méthodes de détection adaptées pour traiter la multi-appartenance, ainsi que deux méthodes orientées sommet liées au travail présenté dans cet article.

COPRA (Gregory, 2010) et SLPA (Xie *et al.*, 2011) sont des algorithmes de propagation de labels, reprenant les principes de (Raghavan *et al.*, 2007). COPRA introduit un coefficient d'appartenance pour étendre afin d'adapter cette dernière à la multi-appartenance. SLPA définit deux roles : *speaker* et *listener*. Chaque sommet devient speaker à tour de rôle et propage son label pendant que les autres écoutent, ce jusqu'à stabilisation, qui peut prendre un temps arbitrairement long comme dans la plupart des alorithmes de propagation de label. Des méthodes par clustering et optimisation de fonction objectif locale ont également été adaptées, par exemple OSLOM (Lancichinetti *et al.*, 2011).

Parmi les méthodes orientées sommet détectant les communautés recouvrantes, l'algorithme *iLCD* (Cazabet, Amblard, 2011) est basée sur des agents : chaque sommet du graphe est considéré comme un agent et chaque arête comme un lien entre deux agents. Chaque agent connaît un certain nombre de règles et d'actions, telles que rejoindre une communauté existante ou en créer une nouvelle, qui lui permettent de se rapprocher ou non d'autres agents, pour finalement former des communautés.

(Whang *et al.*, 2013) proposent une méthode d'expansion à partir d'une graine, utilisant le PageRank personnalisé comme mesure de préférence, qui s'est révélée efficace lors de travaux antérieurs (Gleich, Seshadhri, 2012). Cette méthode n'est pas décentralisée et utilise beaucoup la propagation de données dans le graphe.

3. Méthode proposée : LOCNeSs

Cette section présente l'algorithme proposé LOCNeSs, pour **L**ocating **O**verlapping **C**ommunities in **Ne**twork **S**tructure**s**, une méthode orientée sommet destinée à découvrir des communautés recouvrantes dans les graphes. Nous décrivons tout d'abord les

principes sous-jacents, en particulier l'extension de l'attachement préférentiel utilisée, puis l'algorithme lui-même. Nous discutons également de quelques propriétés de LOCNeSs : complexité, propagation et stabilité.

Nous utilisons les notations suivantes : $G = (V, E)$ est un graphe connexe simple non orienté et non pondéré, où V est l'ensemble des sommets et E celui des arêtes. Nous posons $n = |V|$ et $m = |E|$, d_v est le degré d'un sommet $v \in V$, et $\Gamma(v)$ l'ensemble de ses voisins. $\mathcal{C} = \{c_1, ..., c_k\}$ est l'ensemble des k communautés formées après la détection et $C : V \to \mathcal{P}(\mathcal{C})$ donne l'ensemble des communautés auxquelles appartient v. Si $|C(v)| > 1$, alors v est multi-appartenant. L'ensemble de tous les leaders sélectionnés par un sommet v est noté A_v

3.1. Principes de LOCNeSs

LOCNeSs repose sur un ensemble de leaders préalablement constitués à l'aide d'une méthode de détection orientée sommet (cf. section 2). Pour prendre en compte la multi-appartenance, il est nécessaire qu'un sommet follower puisse être associé avec plusieurs leaders.

L'implémentation réalisée et évaluée dans cet article repose sur la méthode de Canu et al. (2015) qui autorise chaque sommet à être leader et follower simultanément, réalisant des calculs locaux sur le voisinage de chaque sommet. Cela lui permet d'être totalement décentralisée. Chaque sommet choisit son leader, ce qui mène à une structure d'interpréférence entre sommets. Une étape finale de fusion regroupe tous les sommets se préférant, chaque groupe indépendant formant une communauté. Nous ajoutons à cette base deux améliorations, détaillées plus loin :

1. un follower peut sélectionner plusieurs leaders,
2. l'étape finale de fusion est divisée en deux parties.

L'association leader-follower proposée se fait en utilisant le principe de l'attachement préférentiel (Barabási, Albert, 1999), mécanisme fréquemment rencontré dans la formation de réseaux complexes, en particulier de réseaux sociaux. Selon ce principe, les entités d'un réseau disposant de nombreuses liaisons ont tendance à attirer les nouvelles entités joignant ce réseau, créant des zones denses. Nous en tirons le mécanisme de formation de communautés orienté sommet qui pousse les followers à se lier à des sommets avec lesquels ils partagent beaucoup de voisins, de haut degré si possible, ces sommets étant considérés comme les leaders.

3.2. Description de l'algorithme

Nous décrivons ici les étapes importantes de LOCNeSs. Le pseudo-code de l'étape 2 est détaillé dans l'algorithme 1.

Etape 1 - Constitution des A_v. Il est présumé qu'un ensemble de *leaders potentiels* existe ou est calculé préalablement à cette étape, commune à la plupart des méthodes orientées sommet.

A la fin de cette étape, chaque sommet v est associé à un ensemble A_v de leaders qu'il aura sélectionnés. Il utilise à cette fin une fonction de sélection $f : V \to \mathbb{R}^+$, qu'il applique à chacun de ses leaders potentiels et garde ceux de valeur maximale, ou supérieure à un seuil. Dans l'implémentation choisie, basée sur (Canu *et al.*, 2015), les leaders potentiels sont les voisins directs et f est l'*agreement* : le nombre de sommets en commun parmi les k voisins de plus haut degré (de v, et du leader potentiel).

Etape 2.1 - Affectation à une communauté, fusion. Nous proposons d'adapter l'attachement préférentiel (cf. section 3.1) au cas multi-appartenant. Nous définissons pour cela pour chaque v un, et un seul, leader principal $\hat{a}_v$. Ce leader sert de base pour l'étape de fusion. Dans l'implémentation évaluée, $\hat{a}_v$ est défini comme le sommet de A de degré maximal : $\hat{a}(v) = \arg\max_{a \in A_v} d_a$.

Une étape de fusion est ensuite mise en œuvre, chaque sommet v formant au départ sa propre communauté, puis fusionnant les communautés $C(v), C(\hat{a}_v)$. Cette manière de procéder permet de garder une structure communautaire globale cohérente et donne expérimentalement le meilleur compromis de résultats (détection de la taille des communautés, identification des sommets multi-appartenants, stabilité). D'autres solutions, consistant par exemple à fusionner les communautés des followers avec celles de tous leurs leaders produit trop d'agrégation, et donc peu de communautés de très grandes tailles, ce qui ne correspond pas aux vérités terrain.

Etape 2.2 - Affectation des communautés additionnelles. Une fois l'étape précédente de fusion terminée, chaque v tel que $|A_v| > 1$ (multi-appartenant) se voit ajouté à toutes les communautés $C(u), u \in A_v \setminus \hat{a}_v$, soit les communautés de ses leaders sauf celle de son leader principal avec qui elle a déjà fusionnné. C'est cette étape qui apporte la multi-appartenance.

3.3. Propriétés

Cette section discute de la quantité de propagation générée par LOCNeSs en terme de nombre et taille de messages, de sa stabilité ainsi que de sa complexité temporelle.

Comme mentionné dans la section 2.1, une **propagation** excessive est un désavantage pour une méthode de détection décentralisée. Nous présentons ici une estimation du gain de LOCNeSs, basé sur l'utilisation de la méthode (Canu *et al.*, 2015). Chaque sommet envoyant uniquement des informations à ses voisins, qui lui en envoient en retour, on peut estimer à $2n\bar{d}$ messages de taille k le volume échangé, $\bar{d}$ étant le degré moyen du graphe. Tous les envois de messages dans l'algorithme ayant un volume analogue, le volume globale peut-être estimer en moyenne à $\mathcal{O}(n\bar{d})$. A contrario, un algorithme à propagation de labels classique, par inondation, envoie un label de chaque sommet vers chaque autre du graphe, passant par chaque arête donc. On peut estimer le nombre moyen de messages échangés dans ce cas comme étant de l'ordre de $\mathcal{O}(nm)$. Pour un graphe de 5000 sommets avec $\bar{d} = 10$, le nombre total de messages échangés est dix fois inférieur pour LOCNeSs.

Algorithm 1 LOCNeSs - Processus pour chaque sommet

Require: $G = (V, E)$ un graphe,
$\{A_v : v \in V\}$ ensemble des leaders pour chaque v
Ensure: $C \subset \mathcal{P}(V)$ ensemble de communautés, partition des sommets de G
Step 2.1
for each $v \in V$ **do**
  $\hat{a_v} \leftarrow \arg\max_{a \in A_v} d_a$
  fusion$(C(v), C(\hat{a_v}))$ (cf. Description, étape 2.1)
end for

Step 2.2
for each $v \in V, |A_v| > 1$ **do**
  for all $a_v \in A_v \setminus \{\hat{a_v}\}$ / $v \notin C(a_v)$ **do**
    $C(a_v) \leftarrow C(a_v) \cup \{v\}$
  end for
end for

LOCNeSs offre également une bonne **stabilité**. En effet, sa conception le rend quasi-déterministe, la seule exception étant le cas où plusieurs $\hat{a_v}$ peuvent être indifféremment choisis à l'étape 2. Dans ce cas, un seul est retenu aléatoirement. Les résultats sont donc globalement stables, comme montré expérimentale section 4, LOCNeSs est robuste par rapport aux configurations de graphes et au déterminisme.

Enfin, la **complexité temporelle** de LOCNeSs est, comme pour nombre de méthodes de détection, difficile à établir (Fortunato, 2009). La complexité du processus pour sommet (Algo. 1) peut être estimée à $\mathcal{O}(d_v^2)$, car l'ensemble A_v de chaque sommet v contient au plus tous ses voisins. La complexité de la fusion dépend de la taille des groupes d'interpréférences, donc des communautés, soit $\mathcal{O}(|\bar{c}|)$ en moyenne, où $|\bar{c}|$ est la taille moyenne d'une communauté. La complexité moyenne estimée au final est donc de $\mathcal{O}(\bar{d}^2 + |\bar{c}|)$.

4. Expériences

Nous avons réalisé plusieurs expériences afin d'étudier sur des graphes artificiels (benchmarks) ainsi qu'un graphe de terrain, le comportement et la validité des résultats de LOCNeSs, comparé à d'autres algorithmes, et en particulier sa capacité à identifier correctement et pertinemment des sommets multi-appartenants malgré son cadre de conception très contraint (décentralisation).

Les trois premières expériences utilisent des graphes artificiels générés grâce au benchmark LFR (Lancichinetti *et al.*, 2008), 10 instances sont créées pour chaque jeu de paramètres. Chaque algorithme est ensuite exécuté 10 fois sur chaque instance, et les résultats donnés sont les moyennes et écart-types sur ces 100 lancements. Les valeurs des paramètres du benchmark sont celles utilisées par Xie *et al.* (2013). Pour

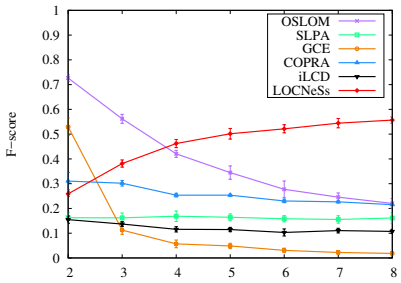

FIGURE *1. Identification des sommets multi-appartenants : F-score en fonction de O_m*

rappel, les principaux sont n : nombre de sommet, O_n : nombre de sommets recouvrants (en pourcentage de n), O_m : nombre exact d'appartenances pour chaque sommet multi-appartenant, $\mu \in [0, 1]$: relation de densité intra/extra communautaire, rendant plus (faibles valeurs) ou moins (fortes valeurs) difficile la détection, et enfin les fourchettes de taille de communautés s (small, entre 10 et 50 sommets par communauté) et b (big, entre 20 et 100 sommets). Les autres paramètres gardent leur valeur par défaut donnée dans (Xie *et al.*, 2013).

Nous utilisons également les critères classiques recommandés par Xie *et al.* (2013) pour comparer les résultats à la vérité terrain : F-score pour l'identification des sommets multi-appartenants, NMI et Omega Index pour la comparaison des partitions de communautés. Les deux premiers sont compris entre $[0, 1]$ et l'Omega Index entre $]-1, 1]$. Tous sont à maximiser. Bien que mesurant des grandeurs similaires, la NMI et l'Omega Index fonctionnent différemment et leurs résultats ne sont pas toujours corrélés, la NMI (basée sur l'entropie) donne de l'importance au fait d'avoir les mêmes paires de sommets dans les mêmes communautés, et considère peu la structure globale de la partition, par exemple le fait d'avoir le même nombre de communautés de tailles similaires, contrairement à l'Omega Index qui se dégrade rapidement si l'on perd la structure globale de la partition. Nous comparons les résultats de LOCNeSs à COPRA, GCE, iLCD, OSLOM et SLPA (cf. section 2). Les valeurs des paramètres requis par ces algorithmes sont fixés conformément à (Xie *et al.*, 2013).

4.1. Qualité de l'identification des sommets multi-appartenants

Cette expérience utilise le F-score pour montrer la capacité de LOCNeSs à identifier les sommets multi-appartenants, comparés aux autres méthodes présentées. Les paramètres de benchmark utilisés sont $n = 5000, \mu = 0.3, O_n = 10\%$, taille s. O_m varie. Les résultats (Fig. 1) montrent que le F-score de LOCNeSs augmente lorsque O_m croît, contrairement à toutes les autres méthodes pour lesquelles il décroît ou reste stable. Ceci est dû au bon rappel de LOCNeSs (non montré ici par manque de place) : 34% pour $O_m = 2$ à 85% pour $O_m = 8$, là ou SLPA, par exemple, stagne aux alentours de 10%. Cependant, la précision est moins bonne : 20% pour $O_m = 2$ à 41% pour $O_m = 8$, la où SLPA fluctue entre 40 et 50%. Autrement dit, LOCNeSs identifie

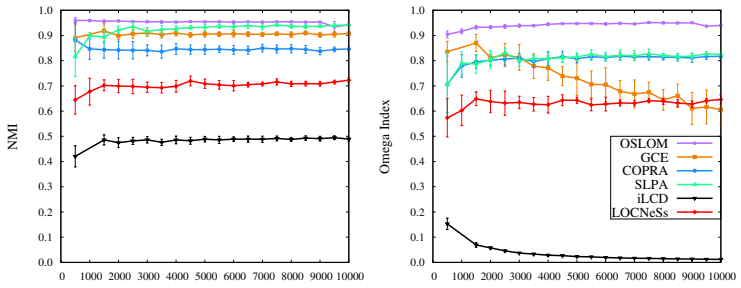

FIGURE 2. *Comparaison de la stabilité de détection en terme de NMI et Omega Index, en fonction de n*

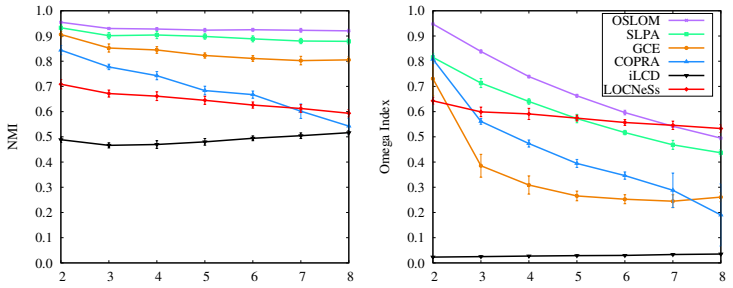

FIGURE 3. *Comparaison de la qualité des partitions en terme de NMI et Omega Index, en fonction de O_m*

plus de sommets multi-appartenant qu'il n'en existe réellement dans la vérité terrain, là où d'autres méthodes comme SLPA par exemple, en identifient moins. On notera que le F-score de LOCNeSs surpasse les autres pour $O_m \geq 4$ en restant croissant, et est toujours meilleur que iLCD, signe que la méthode n'est pas perturbée par la multi-appartenance à un grand nombre de communautés différentes à la fois.

4.2. Qualité des partitions

Le but de cette expérience est de mesurer la sensibilité de LOCNeSs aux paramètres n (Fig. 2) et O_m 3. Les paramètres fixés sont $\mu = 0.3, O_n = 10\%$, tailles de communautés s, ainsi que $O_m = 2$ (Fig. 2), et $n = 5000$ (Fig. 3). On observe que, bien qu'offrant une NMI et un Omega Index plus bas que les méthodes à optimisation et propagation, LOCNeSs reste relativement stable dans tous les cas. En effet on ne note pas de différence de plus de 0.1 sur les critères, et les écarts-types restent raisonnables. LOCNeSs reste également toujours meilleure que *iLCD*.

La figure 3 montre que l'Omega Index de LOCNeSs baisse plus rapidement que pour les autres méthodes, signe que cette dernière parvient bien à identifier les sommets multi-appartenants (cf. expérience précédente) sans trop dégrader la structure communautaire globale, ie. identifie des sommets pertinents.

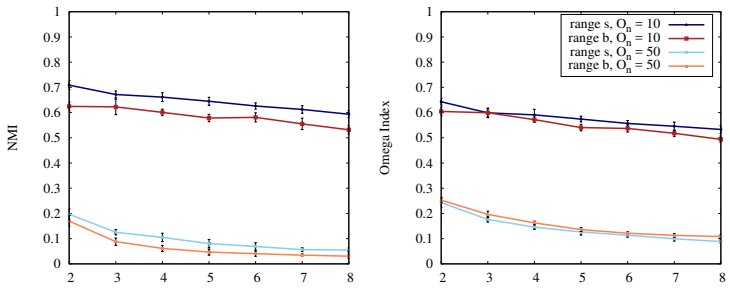

FIGURE 4. *Impact de O_n et des tailles s, b en fonction de O_m pour LOCNeSs seulement*

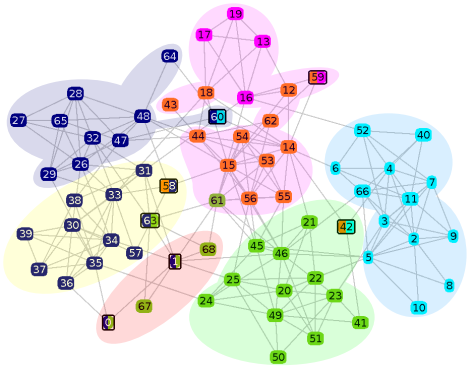

FIGURE 5. *Visualisation des résultats de la détection sur Highschool Network. Couleur de sommet : communauté selon LOCNeSs, couleur de fond : vérité terrain. Les sommets multi-appartenants sont en noir.*

4.3. Proportion de sommets recouvrants, tailles des communautés

Nous mesurons, avec cette expérience, l'impact de deux paramètres principaux : la proportion de sommets multi-appartenants O_n et les tailles de communautés s et b, en NMI en fonction de O_m. Les résultats (Fig. 4) sont présentés uniquement pour LOCNeSs par manque de place. Des expériences similaires peuvent être trouvées dans Xie *et al.* (2013). Les paramètres du graphe sont $n = 5000, \mu = 0.3$. Deux valeurs de O_n sont testées : $O_n = 10\%, O_n = 50\%$, cette dernière valeur n'étant pas réaliste (cf. section 1). Comme l'on peut s'y attendre, les NMI pour $O_n = 50\%$ chutent fortement, les tailles s et b restant néanmoins corrélées. En effet, la moitié des sommets de chaque communauté appartient également à une autre, ce qui efface en grande partie les limites de ces communautés et donc complexifie grandement leur détection. Les petites communautés sont légèrement mieux identifiées que les grosses, un résultat déjà noté par Xie *et al.* (2013).

4.4. Réseau réel : High School Network

Cette dernière expérience illustre visuellement le découpage produit par LOCNeSs sur le graphe d'un ensemble d'élèves d'un collège (Xie *et al.*, 2013). Les six communautés terrain (couleurs de fond) correspondent aux différentes classes auxquelles ils appartiennent. La figure 5 présente les résultats d'un lancement type de LOCNeSs. On observe que LOCNeSs ne détecte pas la communauté rouge (#0, #1, #67, #72), la plus petite du graphe, mais identifie les sommets #0 et #1 comme multi-appartenants, les plaçant dans les communautés jaune et verte. La différence de densité des sous-graphes n'est alors pas suffisante pour créer une communauté à part et #67, #68 sont placés dans la communauté verte. Le même phénomène, en plus important, conduit à ne pas découvrir la communauté jaune, qui est vue comme une continuité de la communauté bleue. La communauté magenta est scindée en deux. Cependant, Xie *et al.* (2013) expliquent que cette communauté comporte deux sous-groupes naturels formés par des élèves de couleur de peau différentes. LOCNeSs réussit à capturer les liens plus distendus et répercute cette scission en deux communautés distinctes. Le sommet #59 est décrit comme étant à la "frontière entre [ces] deux sous-groupes de la même classe" (Xie *et al.*, 2013); LOCNeSs l'identifie comme sommet multi-appartenant.

5. Conclusion et perspectives

Nous avons présenté LOCNeSs, un algorithme orienté sommet pour détecter des communautés recouvrantes dans des graphes. Son implémentation est destinée à être décentralisée et à limiter la propagation de messages dans le réseau lors de l'exécution. Ces caractéristiques sont particulièrement appréciables pour traiter des réseaux mobiles opportunistes. Nous pensons également qu'elles permettent une implémentation très efficace dans des solutions de traitements de données massives « Think Like a Vertex » telles que Pregel ou Giraph. Nous avons montré avec des expériences que LOCNeSs obtient effectivement de bons résultats, en particulier en ce qui concerne la qualité de détection des sommets multi-appartenants.

Des pistes intéressantes poursuivant ce travail sont le traitement des données massives, mentionné ci-dessus, ainsi que le traitement de réseaux dynamiques, qui posent des contraintes supplémentaires de natures très différentes. Nous pensons que la robustesse de LOCNeSs (cf. sections 3.3 et 4) offre une base permettant d'obtenir une bonne stabilité lors de la détection dynamique, ce qui est un atout essentiel.

Remerciements : Ce travail a été réalisé dans le cadre du projet ANR Homo Textilus (financement ANR-11-SOIN-007). Merci à Chi Dan Pham pour l'aide sur le nom de l'algorithme.

Bibliographie

Andersen R., Lang K. J. (2006). Communities from Seed Sets. In *Proc. of the 15th Intl. Conf. on World Wide Web*, p. 223–232.

Barabási A.-L., Albert R. (1999). Emergence of Scaling in Random Networks. *Science*, vol. 286, n° 5439, p. 509–512.

Bedi P., Sharma C. (2016). Community detection in social networks. *WIREs Data Mining Knowl. Discov.*, vol. 6, n° 3, p. 115–135.

Canu M., Detyniecki M., Lesot M.-J., Revault d'Allonnes A. (2015). Fast community structure local uncovering by independent vertex-centred process. In Proc. of the 2015 IEEE/ACM Intl. Conf. on Advances in Social Networks Analysis and Mining, p. 823–830.

Cazabet R., Amblard F. (2011). Simulate to Detect: A Multi-agent System for Community Detection. In *Proc. of the 2011 IEEE/WIC/ACM Web Intelligence and Intelligent Agent Technology*, vol. 2, p. 402–408.

Danisch M., Guillaume J.-L., Grand B. L. (2013, mai). Une approche à base de proximité pour la détection de communautés egocentrées. In *Actes de la 4ème Conférence sur les Modèles et l'Analyse des Réseaux : Approches Mathématiques et Informatique (MARAMI'13)*

Fortunato S. (2009). Community detection in graphs. *Phys. Rep.*, p. 75–174.

Gleich D. F., Seshadhri C. (2012). Vertex Neighborhoods, Low Conductance Cuts, and Good Seeds for Local Community Methods. In *Proc. of the 18th ACM Intl. Conf. on Knowl. Discov. and Data mining*, KDD'12, p. 597–605.

Gregory S. (2010). Finding overlapping communities in networks by label propagation. *New Journal of Physics*, vol. 12, p. 1–26.

Kanawati R. (2014). YASCA: an ensemble-based approach for community detection in complex networks. In *Computing and Combinatorics*, p. 657–666.

Lancichinetti A., Fortunato S., Radicchi F. (2008). Benchmark graphs for testing community detection algorithms. *Phys. Rev. E*, vol. 78, n° 4, p. 046110.

Lancichinetti A., Radicchi F., Ramasco J. J., Fortunato S. (2011). Finding Statistically Significant Communities in Networks. *PLOS ONE*, vol. 6, n° 4, p. e18961.

Rabbany R., Chen J., Zaıane O. R. (2010). Top leaders community detection approach in information networks. In *Proc. of the 4th SNA-KDD workshop.*

Raghavan U. N., Albert R., Kumara S. (2007). Near linear time algorithm to detect community structures in large-scale networks. *Phys. Rev. E*, vol. 76, n° 3, p. 036106.

Shah D., Zaman T. (2010). Community detection in networks: The leader-follower algorithm. In *Proc. of the 2010 NIPS Workshop on Net. Across Discipl. in Theory and Appl.*.

Tian Y., Balmin A., Corsten S. A., Tatikonda S., McPherson J. (2013). From "Think Like a Vertex" to "Think Like a Graph". *Proc. VLDB Endow.*, vol. 7, n° 3, p. 193–204.

Whang J. J., Gleich D. F., Dhillon I. S. (2013). Overlapping community detection using seed set expansion. In *Proc. of the 22nd ACM Intl. Conf. on Info. & Knowl. Manage.*, p. 2099–2108.

Xie J., Kelley S., Szymanski B. K. (2013). Overlapping Community Detection in Networks: The State-of-the-art and Comparative Study. *ACM Comput. Surv.*, vol. 45, n° 4, p. 43.

Xie J., Szymanski B. K., Liu X. (2011). SLPA: Uncovering Overlapping Communities in Social Networks via A Speaker-listener Interaction Dynamic Process. In *Proc. of the ICDM 2011 Workshop on DMCCI.*

Yakoubi Z., Kanawati R. (2014). LICOD: A Leader-driven algorithm for community detection in complex networks. *Vietnam J. Comput. Sci.*, vol. 1, n° 4, p. 241–256.