Détection de communautés chevauchantes utilisant la propagation de labels robuste et fonction d'appartenance

Jean−Philipp Attal* — **Maria Malek*** — **Marc Zolghadri****

* *EISTI: , 95000 Cergy, France, Laboratoire Quartz*
Email: jal@eisti.eu,mma@eisti.eu
** *SUPMECA, 93407 Saint-Ouen, France, Laboratoire Quartz*
Email: marc.zolghadri@supmeca.fr

RÉSUMÉ. *La propagation de labels est l'une des méthodes les plus rapides pour la détection de communautés, de complexité quasi−linéaire en terme d'arêtes. Il s'agit d'une méthode locale où chaque nœud possède son propre label qui change par interaction avec son voisinage. Malheureusement, cette méthode présente trois inconvénients majeurs: (i) une mauvaise propagation peut mener à de trop grandes communautés (le problème des communautés géantes). (ii) l'instabilité de la méthode, (iii) l'impossibilité de trouver des communautés chevauchantes. Dans cet article, nous proposons d'utiliser une méthode existante pour stabiliser la propagation de labels à laquelle sera adjointe une fonction d'appartenance permettant de faire le chevauchement de communautés. De par cette fonction, les noeuds sont assignés et répliqués le nombre de fois nécessaire dans les communautés jugées chevauchantes. Il est également possible, si l'utilisateur le souhaite, d'imposer un nombre fixe de communautés auxquelles un noeud pourrait appartenir.*

ABSTRACT.

MOTS-CLÉS : *détection de communautés*$_1$, *propagation de labels*$_2$, *barrages*$_3$, *ensemble learning*$_4$.

KEYWORDS:

R^{e} soumission à *TAL VV-NN*, le 28/05/2015

1. Introduction

La plupart des réseaux représentant des systèmes réels montrent des caractéristiques propres comme des groupes de noeuds fortement liés entre eux (que l'on appelle des communautés) et peu avec le reste du graphe. Un réseau de collaboration de chercheurs en est un exemple, où les groupes de noeuds sont des personnes travaillant sur des thèmes similaires. Une étude comparative des méthodes de détections de communautés a été effectuée par Fortunato et al. (Fortunato, 2010).

Dans cet article, nous exposons un nouvel algorithme de détection de communautés chevauchantes, basé sur une méthode de détection de coeurs par propagation de labels déjà existante. Nous exposons la méthode de propagation de labels générale et quelques variantes à la section 2. Dans la section 3, nous exposons notre proposition algorithmique pour la détection de communautés chevauchantes. En section 4, nous exposons les résultats expérimentaux sur des graphes réels que nous comparons avec d'autres algorithmes issus de la littérature. Finalement, en section 5, nous concluront et traiterons de nos futures perspectives.

2. L'approche par propagation de labels

2.1. *La propagation de labels standard*

La méthode de propagation de labels (Raghavan *et al.*, 2007) est basée sur la transmission d'un label d'un nœud à ses voisins. Un état d'équilibre est atteint lorsque chaque noeud a son label égal à celui de la majorité de ses voisins. Soit un graphe $G = (V, E)$, avec V l'ensemble des sommets ($|V| = n$) et E l'ensemble des arêtes ($|E| = m$). A chaque étape, chaque noeud met à jour son label selon les labels de ses voisins, en utilisant un vote. Le label du noeud x prendra le label majoritaire de ses voisins. En notant c_x le label du noeud x, et par $N^l(x)$ l'ensemble du voisinage du noeud x avec le label l, l'affectation d'un label au noeud x est donnée par la formule suivante :

$$c_x = \arg\max_l |N^l(x)| \quad [1]$$

A la fin du processus, les nœuds ayant le même label représentent une communauté. Cette méthode peut être effectuée de manière *synchrone* ou *asynchrone*. La méthode asynchrone signifie que la mise à jour d'un label d'un noeud est sue par tous les autres noeuds du graphe immédiatement. Son label est utilisé pour la mise à jour des labels des autres noeuds. Ce qui n'est pas le cas du mode asynchrone, où la mise à jour des labels utilise les labels des noeuds à la précédente propagation.

Cet algorithme étant très instable, il serait souhaitable d'avoir une méthode de stabilisation. Pour ce faire, une méthode consiste à lancer plusieurs fois l'algorithme

non déterministe et à considérer les nœuds qui apparaissent le plus souvent ensemble dans une même communauté. On appelle ces nœuds dont la fréquence d'apparition est très forte, des *cœurs*. Nous proposons d'utiliser la méthode de Seifi et al. (Seifi *et al.*, 2013). Elle consiste à utiliser une matrice de fréquence, spécifiant le nombre de fois que chaque paire de noeuds apparaît dans les mêmes communautés. Les auteurs de cette méthode l'ont appliquée en utilisant la méthode de Louvain (Blondel *et al.*, 2008). Il s'agit d'une méthode agglomérative, dont la contraction est effectuée par optimisation locale d'une fonction de qualité , la modularité (Newman et Girvan, 2004). Les auteurs ont montré qu'ils avaient réussi à stabiliser la méthode de Louvain, et à trouver des coeurs stables.

Soit $\mathcal{N}$ le nombre de fois que l'algorithme non déterministe est lancé. A chaque essai, nous notons chaque paire de nœuds qui apparait dans une même communauté. Il est ainsi possible de définir une matrice $P_{ij}^{\mathcal{N}} = [p_{ij}]_{n\times n}^{\mathcal{N}}$ telle que p_{ij} représente la fréquence d'appartenance des nœuds i et j à une même communauté après les $\mathcal{N}$ essais. p_{ij}, étant une probabilité,$\forall (i,j) \in V \times V$ a une valeur comprise entre 0 et 1. Un nombre proche de 1 signifie que les noeuds i et$j$ sont souvent ensemble durant les $\mathcal{N}$ essais. Pour trouver les coeurs, on crée un nouveau graphe $G' = (V, E')$ où E' représente l'ensemble des arêtes créées à partir de la matrice de fréquence en utilisant un seuil $\alpha \in [0, 1]$ permettant de faire apparaître des composantes connexes. Les composantes connexes sont ici les coeurs, qui correspondent à nos communautés. G' est appelé le graphe α-seuillé. α est un paramètre qui influence le fait de créer des connections dans le nouveau graphe G', et par conséquence, le nombre de composantes connexes. D'après les études menées par Seifi et al. (Seifi *et al.*, 2013), de faibles valeurs de α conduisent à peu de composantes connexes alors que de fortes valeurs de α mènent à beaucoup de composantes connexes. La propagation de labels avec détection de coeurs fut utilisée dans nos précedentes recherches (Attal et Malek, 2015), avec des résultats en terme de qualité de partitionnement très encourageants.

Il existe de très nombreuses variantes de la propagations de labels qui ont été implémentées à la fois pour la détection de communautés chevauchantes et non chevauchantes, le lecteur trouvera un état de l'art avec (Xie *et al.*, 2013).

3. Fonctions d'appartenances pour le chevauchement de communautés

La proposition algorithmique pour la détection de communautés chevauchantes est basée sur l'aspect sociologique d'un individu vis−à−vis de ses relations mais également sur une vision plus large, par observation topologique des communautés avec lesquelles il est lié.

Notre proposition algorithmique repose sur le graphe seuillé G', résultant de la matrice de co-fréquence $P_{ij}^{\mathcal{N}}$. L'idée est de pouvoir utiliser la pondération des liens que l'on peut trouver dans la matrice de co-fréquence comme un degré de relation d'amitié et la structure topologique des communautés pour assigner un noeud à ces dernières, selon certaines conditions. Le nouveau graphe G' est alors projeté sur le graphe originel G, tout en respectant sa topologie. Cela signifie que les arêtes présentes dans G' mais pas dans G ne seront pas considérées, afin de ne pas mettre de liens absurdes. Ce peut être le cas s'il y a des conflits entre personnes ou que des individus refusent de parler à d'autres individus, comme dans le graphe de Zachary entre le manager et l'entraîneur. Nous utilisons ainsi l'information stockée dans $P_{ij}^{\mathcal{N}}$ pour pondérer G. Cela nous permet de voir les paires de noeuds ayant une forte probabilité d'être ensemble en terme communautaire. En utilisant le seuil α sur le nouveau graphe pondéré, nous obtenons les communautés et les arêtes entre communautés (AEC). Les noeuds qui sont à la frontière de leurs communautés et reliés à d'autres sont de possibles candidats pour le chevauchement. Pour savoir si ces noeuds candidats sont de potentiels futurs noeuds chevauchants, nous proposons les fonctions d'appartenances suivantes.

Considérant un noeud candidat x , l'idée est de mesurer le pouvoir d'appartenance qu'a ce noeud en observant ses communautés avoisinantes et leurs structures topologiques. Nous écrivons $\mathcal{C}_x = \{\mathcal{C}_1^x, ..., \mathcal{C}_K^x\}$, les différentes communautés auxquelles le noeud x est lié. Si x est lié à K différentes communautés, nous le notons par $|\mathcal{C}_x| = K$.

3.0.1. ***Fonction 1*** *Fonction d'appartenance basée sur la densité*

Nous considérons dans cette première configuration la densité des communautés avec les poids sur les liens liés au noeud x. En considérant les k différentes communautés présentes dans le voisinage du noeud x, nous proposons *la fonction d'appartenance basée sur la densité* :

$$f_d : x \times \{C_1^x, ..., C_K^x\} \longmapsto \mathbb{R}_+$$

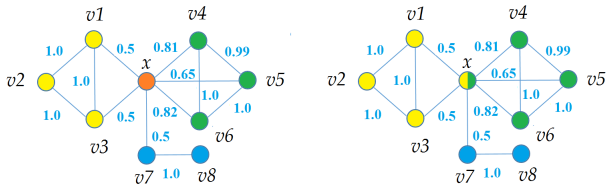
$$f_d(x, \{C_1^x, ..., C_K^x\}) = \max_{c \in \binom{\mathcal{C}_K^x}{j}, j \in \{1,...,|C_K^x|\}} (\frac{1}{|c|} \times \sum_{i \in c} \omega_{x,i} d^S(i))$$

oú $\omega_{x,i}$ est le poids de l'arête liant le noeud x au noeud i et c étant une liste de combinaisons de communautés. Le coefficient binomial $\binom{\mathcal{C}_K^x}{j}$ permet de calculer les j combinaisons dans un ensemble de $\mathcal{C}_K^x$, les éléments étant les communautés. Cela permet au noeud x d'appartenir à une ou plusieurs communautés. La configuration ayant le score le plus élevé permettra l'assignement du noeud x aux communautés choisies, en les répliquant. Un noeud avec une forte pondération sur ses liens connectés à des communautés ayant de fortes densités aura plus de chances d'être chevauchant qu'un noeud avec de faibles pondérations sur ses arêtes connectées

à des communautés de faibles densités. Cependant, dans certaines situations, les chevauchements ne peuvent pas se faire. Cela peut être le cas si un noeud x est très faiblement lié à ses communautés avoisinantes, elles mêmes de faibles densités, avec une valeur de f_d relativement faible. Basé sur ce constat, le chevauchement sera effectué si et seulement si $f_d(x, \{C_1^x, ..., C_K^x\}) \geq \frac{1}{|c|} \sum_{S \in c} d^S$ oú d^S est la densité du sous graphe (ici la communauté) S.

Nous notons dans la formule ci-dessus que $j \in \{1, ..., |C_K^x|\}$, mais il est tout à fait possible de forcer le fait qu'un noeud candidat soit chevauché par au moins L communautés, en écrivant $j \in \{L, ..., |C_K^x|\}$, tout en respectant la contrainte portant sur la densité. Cela permet à l'utilisateur de choisir le nombre de communautés auxquelles peut appartenir un noeud candidat.

Supposons que nous ayons le graphe Fig. 1, avec la partition $P = \{C_1, C_2, C_3, C_4\}$ telle que $C_1 = \{v_1, v_2, v_3\}$,$C_2 = \{v_4, v_5, v_6\}$, $C_3 = \{v_7, v_8\}$ et $C_4 = \{x\}$, résultante de la propagation de labels avec matrice de co-fréquence. La question est de savoir si le noeud x peut appartenir à plusieurs communautés. On observe que la meilleure configuration pour l'obtention de communautés chevauchantes sur le noeud x est donnée avec C_1 et C_2. Le noeud x est ainsi répliqué dans la combinaison oú les densités sont les plus élevées, et les pondérations des liens sont les plus fortes.



Expérimentations		
combinaisons	f_{cc} score	f_d score
$f_{cc}(x, \{C_1, C_2, C_3\})$	1.093	1.26
$f_{cc}(x, \{C_1, C_2\})$	**1.64**	**1.64**
$f_{cc}(x, \{C_1, C_3\})$	0.5	0.75
$f_{cc}(x, \{C_2, C_3\})$	1.14	1.39

Figure 1. *Après avoir calculé f_d et f_{cc} sur le noeud x, x appartient à deux communautés.*

3.0.2. ***Fonction 2*** *Fonction d'appartenance basée sur le coefficient de clustering*

Le coefficient de clustering (CC) (Watts et Strogatz, 1998) est une mesure d'analyse des réseaux sociaux de regroupement des noeuds dans un réseau. Il mesure à quel point le voisinage d'un sommet est connecté, et calcule plus exactement la probabilité que deux individus liés à une tierce personne soient également liés.

Le CC a une forme globale et une forme locale. La première (globale) concerne le graphe dans son ensemble alors que la seconde (locale) ne concerne que le noeud.

Pour un noeud $x \in G$, le CC est défini par :

$$CC_x = \frac{\text{nombre de triangles connectés au noeud } x}{\text{nombre de triplets centrés autour du noeud } x}$$

oú le triplet centré autour du noeud x est un sous-graphe connexe à trois noeuds. Par défaut, si le degré du noeud x est de 1 ou de 0, nous posons $CC_x = 0$.

Le coefficient de clustering global pour le graphe G est calculé en utilisant la valeur locale $CC_x, \forall x \in G$

$$CC(G) = \frac{1}{n} \sum_{x \in G} CC_x$$

Par définition, nous avons $0 \leq CC_x \leq 1, \forall x \in G$ et $0 \leq CC \leq 1$. Pour un noeud x, plus grand est son CC, meilleur sera son voisinage en terme de clique.

Nous définissons *la fonction d'appartenance basée sur le CC* : $f_{cc} : x \times \{C_1^x, ..., C_K^x\} \longmapsto \mathbb{R}_+$ tel que :

$$f_{cc}(x, \{C_1^x, ..., C_K^x\}) = \max_{c \in \binom{C_K^x}{j}, j \in \{1,...,|C_K^x|\}} (\frac{1}{|c|} \times \sum_{i \in c} \omega_{x,i} CC^S(i))$$

L'idée d'utiliser le CC est que le noeud candidat observera les connections au sein des communautés, un nombre de triangles important et la présence de leaders, caractéristique des graphes de terrains dont la distribution des degrès des noeuds suit une loi faible.

3.1. *Proposition algorithmique pour la détection de communautés chevauchantes*

Nous exposons la propagation de labels avec détection de coeurs et chevauchement (CDLPOV), algorithme 1. $f_{appartenance}$ réfère à la fonction choisie par l'utilisateur, à savoir f_d ou f_{CC}. $SNM(S)$ est la mesure d'analyse des réseaux sociaux utilisée concernant l'aspect topologique de la communauté S, à savoir la densité ou le CC. En faisant varier α dans un intervalle et avec un pas spécifique, nous pouvons obtenir un dendrogramme chevauchant.

4. Experimentations pourtant sur le CDLPOV

Pour établir les comparaisons, nous utilisons des mesures supervisées (lorsque nous connaissons les vraies communautés) telles que l'information mutuelle normalisée (NMI) (Ana et Jain, 2003) dans sa version chevauchante, l'indice Omega et le

Algorithme 1 Le CDLP avec fonction d'appartenance (CDLPOV)

Input : Un graphe $G = (V, E)$, le seuil α, $\mathcal{N}$ le nombre de lancements
Output : Les communautés chevauchantes de G

Allouer une matrice de co-fréquence vide
Lancez $\mathcal{N}$ fois la propagation de labels asynchrone
Remplir la matrice de co-fréquence avec les résultats des $\mathcal{N}$ LPA
Créer un nouveau graphe $G' = (V, E')$ de $P_{ij}^{\mathcal{N}}$ avec des arêtes dont la pondération est supérieure ou égale à α
Projeter le graphe G' sur G avec la pondération (mais en enlevant les arêtes présentes dans G' mais pas dans G)
Créer une partition $P = \{P_1, ..., P_{\mathcal{C}}\}$ en considérant les $\mathcal{C}$ composantes connexes comme coeurs
Calculer les arêtes entre communautés (AEC)
$Cand \leftarrow []$ {*Cand* est une liste de candidats potentiel au chevauchement}
Pour chaque noeud x ayant une arête dans AEC
$Cand$.ajouter(x)
Fin Pour
$P^{Ov} \leftarrow P$
Pour chaque noeud x dans $Cand$
$Cand$.ajouter(x)
Fin Pour
Pour chaque noeud x dans $Cand$
Si $f_{appartenance}(x, \{C_1^x, ..., C_K^x\}) \geq \sum_{S \in \{C_1^x, ..., C_K^x\}} SNM(S)$
Dupliquer le noeud x dans les communautés correspondantes de P^{Ov}
Fin Si
Fin Pour
Retournez la partition $P^{Ov} = \{P_1^{Ov}, ..., P_{\mathcal{C}}^{Ov}\}$.

nombre de communautés $\#$ et non supervisées telle que la modularité Q (Nicosia *et al.*, 2009) dans sa version chevauchante. Nous prenons $\mathcal{N} = 100$ pour notre paramétrisation. Les réseaux sur lesquels nous faisons marcher nos algorithmes sont décrits Table 1, le réseau de karaté de Zackary (Zachary, 1977) (Zac), un réseau de football (Girvan et Newman, 2002) (Foot), un réseau de livres politiques (Krebs, 2004) (Pol), un réseau de dophins (Lusseau *et al.*, 2003) (Dol), un réseau de co-auteurs scientifiques (Newman, 2006) (NS) et un réseau de musiciens (Gleiser et Danon, 2003)(jazz).

Characteristics of some networks										
réseaux	$\|V\|$ et $\|E\|$	Densité	diamétre	CC		réseaux	$\|V\|$ et $\|E\|$	Densité	diamétre	CC
Zachary	$34 \setminus 78$	0.139	5.0	0.256		Pol	$105 \setminus 441$	0.081	7.0	0.348
Foot	$115 \setminus 615$	0.094	4.0	0.407		NS	$1589 \setminus 2742$	0.002	17.0	0.693
Dol	$62 \setminus 159$	0.084	8.0	0.309		Jazz	$198 \setminus 2742$	0.140	6	0.52

Tableau 1. *Caractéristiques de certains réseaux avec (CC) le coefficient de clustering et le nombre de communautés (#)*

4.0.1. *Zachary Karate Club*

Le graphe est caractérisé par deux communautés.

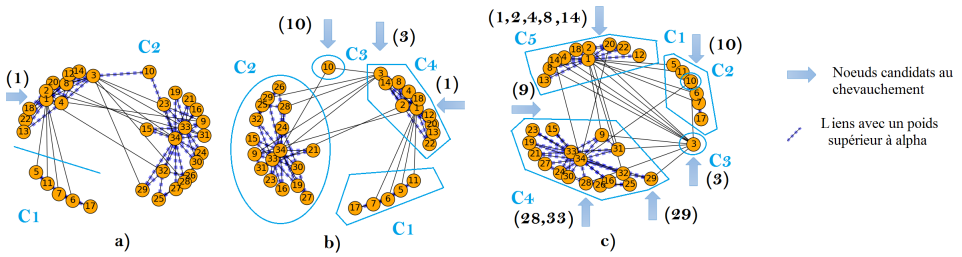


Figure 2. *Communautés avec différentes valeurs de α en utilisant f_d et f_{cc}, a) $\alpha \geq 0.6$, b) $\alpha \geq 0.7$ c) $\alpha \geq 0.8$*

Sur la figure 2, les candidats pour le chevauchement avec f_d et f_{cc} sont les mêmes. Pour $\alpha \leq 0.6$, seul le noeud 10 est chevauchant. Il est assigné à la communauté C_2 avec f_{cc} alors qu'il est assigné à deux communautés avec f_d (qui sont C_3 et C_4). Cela vient du fait que la communauté C_2 a un nombre plus important de triangles que C_4. Pour $\alpha \geq 0.8$, le noeud 3 qui est connu dans la littérature comme étant chevauchant, est assigné à deux communautés avec f_d (C_5 et C_4), mais juste à une communauté avec f_{cc} (C_4). Le noeud 1 est répliqué dans une communauté (C_2) pour chacune des deux méthodes. D'après les résultats de la table 2, plus la valeur de α est élevée, plus le nombre de candidats pour le chevauchement devient important. Cela vient du fait que les tailles des communautés diminuent alors que α croît. De ce fait, le pourcentage AEC devient plus important, augmentant les noeuds candidats. Même si les noeuds chevauchants sont les mêmes selon f_d et f_{cc} jusque $\alpha \geq 0.9$, la qualité des résultats est meilleure en utilisant f_d plutôt que f_{cc}. La valeur de la modularité la plus élevée est pour $\alpha \geq 0.7$ (0.62 pour chacune des méthodes) avec les valeurs les plus fortes pour le NMI et pour l'indice Ω.

Résultat avec f_d et f_{cc} sur le club de Karate								
f_d	Cand	CandOv	AEC	Q_{Ov}^{Nic}	Ω	F_1	NMI	#
$\alpha \geq 0.6$	47.058%	2.94% (1)	17.95%	0.3986	0.0645	0.65	0.2365	2
$\alpha \geq 0.7$	41.17%	8.8235% (3)	16.0%	0.6210	0.7110	0.857	0.5178	4
$\alpha \geq 0.8$	55.88%	32.352% (11)	26.92%	0.4202	0.4923	0.7499	0.3488	5
$\alpha \geq 0.9$	55.88%	32.352% (11)	26.92%	0.4202	0.4923	0.7499	0.3488	5
f_{cc}	Cand	CandOv	AEC	Q_{Ov}^{Nic}	Ω	F_1	NMI	#
$\alpha \geq 0.6$	47.058%	2.941% (1)	17.948%	0.3986	0.064	0.65	0.2365	2
$\alpha \geq 0.7$	41.176%	8.823% (3)	16.0%	0.6210	0.7110	0.8571	0.5178	3
$\alpha \geq 0.8$	55.882%	32.353% (11)	26.923%	0.4202	0.4923	0.7499	0.3488	5
$\alpha \geq 0.9$	55.882%	32.353% (11)	26.923%	0.4202	0.4923	0.7499	0.3488	5

Tableau 2. *Cand : candidats possibles , CandOv : Pourcentage de noeuds chevauchants*

4.0.2. *Les dauphins de Nouvelle Zélande*

Résultats avec f_d et f_{cc} sur le réseau de dauphins								
f_d	Cand	CandOv	AEC	Q_{Ov}^{Nic}	Ω	F_1	NMI	#
$\alpha \geq 0.5$	51.61% %	0.0%	20.38%	0.7959	1.0	1.0	1.0	2
$\alpha \geq 0.6$	54.838%	6.451% (4)	24.050%	0.7502	0.6165	.8571	0.5936	4
$\alpha \geq 0.7$	64.51%	8.0645% (5)	30.57%	0.7144	0.4777	0.7499	0.457	5
$\alpha \geq 0.8$	61.29%	19.3548% (12)	29.30%	0.6052	0.4777	0.6184	0.4421	8
$\alpha \geq 0.9$	77.41%	25.81% (16)	43.94%	0.5415	0.3549	0.5333	0.2456	12
f_{cc}	Cand	CandOv	AEC	Q_{Ov}^{Nic}	Ω	F_1	NMI	#
$\alpha \geq 0.5$	51.613%	0.0%	20.382%	0.7959	1.0	1.0	1.0	2
$\alpha \geq 0.6$	54.838%	6.451% (4)	24.051%	0.7502	0.6125	.8571	0.5936	4
$\alpha \geq 0.7$	64.516%	8.0645% (5)	30.57%	0.7144	0.4294	0.7499	0.457	5
$\alpha \geq 0.8$	61.29%	19.3548% (12)	29.299%	0.6062	0.4777	0.6184	0.4421	8
$\alpha \geq 0.9$	77.419%	35.483% (22)	43.949%	0.4412	0.5882	0.5489	0.2772	12

Tableau 3. *Cand : candidats possibles, CandOv : Pourcentage de noeuds chevauchants*

Le graphe est composé de deux communautés, qui regroupent les mâles et les femelles. A partir de la table 3, pour $\alpha \geq 0.5$, l'algorithme et les fonctions trouvent les deux communautés, sans aucune réplication, avec un NMI, un indice d'omega et un F_1 score de 1.0. En augmentant la valeur de α, la taille des communautés diminue tandis que le nombre de candidats possibles pour le chevauchement augmente. Le pourcentage de noeuds répliqués est le même pour $\alpha \geq 0.6$ jusqu'à $\alpha \geq 0.8$. Malgré cela, les deux méthodes ne répliquent pas les noeuds candidats de la même manière. f_{cc} produit la même qualité en terme de communautés mais réplique davantage de noeuds que f_d, surtout pour de fortes valeurs de α.

4.0.3. *Les livres politiques de Krebs*

C'est un réseau de livres politiques datant de l'élection présidentielle américaine de 2004 et vendus sur le site de vente en ligne *Amazon.com*. Ce graphe comporte trois communautés au sens politique, à savoir les démocrates, les républicains et le centre sur l'échiquier politique.

Resultats avec f_{cc} et f_d sur les livres politiques de Kreb								
f_d	Cand	CandOv	EBC	Q_{Ov}^{Nic}	Ω	F_1	NMI	#
$\alpha \geq 0.4$	24.762%	0.0%	5.215%	0.8342	0.6671	0.7883	0.4521	2
$\alpha \geq 0.5$	26.67%	0.95% (1)	6.576%	0.834	0.6538	0.7844	0.4940	2
$\alpha \geq 0.6$	31.43%	1.90% (2)	7.709%	0.8448	0.6755	0.7128	0.3867	3
$\alpha \geq 0.7$	32.38%	5.71% (6)	9.070%	0.7596	0.6863	0.6638	0.3541	4
$\alpha \geq 0.8$	35.24%	15.24% (16)	9.977%	0.6533	0.6672	0.5655	0.2901	7
f_d	Cand	CandOv	EBC	Q_{Ov}^{Nic}	Ω	F_1	NMI	#
$\alpha \geq 0.4$	24.761%	0.0%	5.215%	0.8342	0.6671	0.7883	0.5039	2
$\alpha \geq 0.5$	26.666%	0.952% (1)	6.576%	0.8342	0.6538	0.7744	0.4940	2
$\alpha \geq 0.6$	31.428%	0.952% (1)	7.709%	0.8443	0.6760	0.7189	0.4485	3
$\alpha \geq 0.7$	32.380%	5.714% (6)	9.070%	0.7821	0.6871	0.6531	0.3401	4
$\alpha \geq 0.8$	35.238%	15.238% (16)	9.977%	0.6533	0.6669	0.5318	0.2796	7

Tableau 4. *Cand : Candidats possibles , CandOv : Pourcentage de noeuds chevauchants*

D'après la table 4, c'est pour $\alpha \geq 0.5$ que les premiers noeuds chevauchants apparaissent. Les résultats sont très similaires entre les deux méthodes , néanmoins, f_d donne un taux de réplication pour les noeuds candidats plus élevé selon nos observations. Le nombre de communautés augmente lentement en fonction de la valeur α, comme celui des candidats au chevauchement. De $\alpha \geq 0.4$ à $\alpha \geq 0.8$, la majorité des noeuds chevauchants sont neutres sur le plan politique américain.

4.1. *Analyse comparative*

Nous comparons nos propositions algorithmiques avec celles issues de la littérature les plus utilisées, à savoir : CFinder (Palla *et al.*, 2005), COPRA ($\nu = 2$ and $\nu = 3$ (Gregory, 2010), ν étant le nombre de communautés auquel un noeud appartient), OSLOM (Lancichinetti *et al.*, 2011), SLPA (Xie *et al.*, 2011), et CONGA (Gregory, 2007).

Analyse comparative													
Networks	F_1	Ω	NMI	Q_{Ov}^{Nic}	#	%	Networks	F_1	Ω	NMI	Q_{Ov}^{Nic}	#	%
Zac $\#2$							**Dol** $\#2$						
CFinder	0.48	0.35	0.18	0.52	3	5.88%	CFinder	0.57	0.35	0.26	0.66	4	3.72%
OSLOM	**0.86**	**0.84**	**0.80**	**0.748**	2	2.94%	OSLOM	**1.0**	**0.914**	**0.852**	0.742	2	1.61%
CONGA	0.65	0.113	0.274	0.441	2	2.94%	CONGA	0.85	0.892	0.821	0.746	2	3.22%
$COPRA_2$*	0.281	0.266	0.228	0.414	11.3	5.58%	$COPRA_2$*	0.933	0.788	0.751	0.693	10.8	0.52%
$COPRA_3$*	0.684	0.359	0.347	0.452	6.4	12.64%	$COPRA_3$*	0.893	0.767	0.701	0.677	3.7	7.73%
SLPA*	0.86	0.633	0.564	**0.608**	2.12	2.20%	SLPA*	0.56	0.754	0.632	0.742	3.44	2.00%
CDLPOV f_d*	0.852	0.711	**0.518**	0.621	4	8.82%	CDLPOV f_d*	**1.0**	**1.0**	**1.0**	**0.796**	2	0.0%
CDLPOV f_{cc}*	0.852	0.711	**0.518**	0.621	4	8.82%	CDLPOV f_{cc}*	**1.0**	**1.0**	**1.0**	**0.796**	2	0.0%
Foot $\#12$							**Pol** $\#4$						
CFinder	0.701	0.64	0.55	0.51	13	6.9%	CFinder	**0.855**	**0.740**	**0.79**	0.884	4	(9)
OSLOM	**0.954**	**0.802**	**0.759**	0.696	12	0.0%	OSLOM	**0.814**	**0.704**	**0.55**	**0.847**	2	1.90%
CONGA	0.823	0.321	0.423	0.451	11	60.0%	CONGA	0.688	0.651	0.49	0.779	4	4.16%
$COPRA_2$*	0.933	0.788	0.705	0.693	10.8	0.52%	$COPRA_2$*	0.687	0.637	0.385	0.825	3	1.05%
$COPRA_3$*	0.944	0.747	0.712	0.668	11.2	2.52%	$COPRA_3$*	0.702	0.649	0.416	0.827	2.8	6.47%
SLPA*	0.748	0.684	0.612	**0.715**	10.30	1.69%	SLPA*	0.755	0.648	0.497	0.83	3.40	12.5%
CDLPOV f_d*	0.854	0.865	**0.751**	0.699	11	0.0%	CDLPOV f_d*	0.784	0.654	0.495	0.844	3	1.90%
CDLPOV f_{cc}*	0.854	0.865	**0.751**	0.699	11	0.0%	CDLPOV f_{cc}*	**0.788**	**0.667**	**0.503**	**0.834**	2	0.0%
NS							**Jazz**						
CDLPOV f_d*				0.977	293	0.0%	CDLPOV f_d*				0.64	2	0.50%
CDLPOV f_{cc}*				0.977	293	0.0%	CDLPOV f_{cc}*				0.64	2	0.50%

Tableau 5. *(*) algorithmes basés sur la propagation de labels*

Nous montrons les résultats de nos méthodes donnant les scores des mesures non supervisées les plus élévés dans la table 5. Nos proposition algorithmiques donnent de relativement bons résultats en terme de qualité. Nous obtenons de meilleurs résultats que COPRA et une meilleure stabilisation. Même si les algorithmes à base de propagation de labels produisent en moyenne plus de communautés, CDLPOV avec f_d et f_{cc} en produit moins. Nous expliquons ce fait par la présence de la matrice de co-fréquence qui stabilise la propagation de labels.

5. Conclusion et perspectives

Nous avons proposé deux méthodes appliquées à une méthode de propagation de label par coeurs pour la détection de communautés chevauchantes. Chacune de ces méthodes utilise la matrice de co-fréquence et les caractéristiques sociales et topologiques des communautés pour savoir si certains noeuds peuvent appartenir à plusieurs communautés. Les utilisateurs ont le choix entre laisser l'algorithme assigner de possibles candidats au chevauchement à un nombre spécifique de communautés auxquelles ces noeuds appartiendraient, ou laisser les fonctions d'appartenance tester toutes les combinaisons pour des assignations automatiques. Les fonctions d'appartenances sont basées sur la densité (f_d), le CC (f_{cc}) et sur les communautés détectées. Les résultats selon les différentes fonctions sont assez similaires en terme de qualité. Néanmoins, le taux d'assignation de noeuds chevauchants à un plus grand nombre de communautés est plus important avec f_d qu'avec f_{cc}. Concernant le temps d'exécution, plus la densité du graphe étudié est importante, plus le nombre de candidats sera important, augmentant ainsi le nombre de combinaisons à tester pour nos fonctions, et par conséquent le temps augmentera. Nous avons vu que calculer les 100 LPA pour alimenter la matrice de fréquence est assez rapide, (1 seconde pour Zachary et 6 secondes pour NS). Néanmois, le temps de calcul de f_{cc} est plus important que celui de f_d. Cela vient du calcul du nombre de triangles au sein des communautés pour calculer le CCL. Le temps de calcul des deux fonctions augmente parrallèlement à α. Mais à partir d'un seuil, le temps devient très important. Pour $\alpha \geq 0.5$, nous avons besoin de 30 secondes pour calculer f_d et 60 secondes sur NS. Pour $\alpha \geq 0.8$, nous avons besoin d'une centaine de secondes pour calculer f_d et f_{cc} toujours sur NS. Pour les réseaux portant sur Zachary, les dauphins, SW ou les livres politiques, il faut approximativement 10 secondes pour f_d et près de 20 secondes avec f_{cc}. Nous souhaiterions développer une version parallèle et distribuée pour travailler sur des grands graphes. Nos recherches se basent actuellement sur la retranscription des fonctions d'appartenances sur les réseaux mutiplexes.

6. Bibliographie

Ana L., Jain A. K., « Robust data clustering », *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, vol. 2, IEEE, p. II-128, 2003.

Attal J.-P., Malek M., « A new label propagation with dams », *2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, IEEE, p. 1292-1299, 2015.

Blondel V., Guillaume J., Lambiotte R., Mech E., « Fast unfolding of communities in large networks », *J. Stat. Mech*p. P10008, 2008.

Fortunato S., « Community detection in graphs », *Physics Reports*, vol. 486, n° 3, p. 75-174, 2010.

Girvan M., Newman M. E. J., « Community structure in social and biological networks », *Proceedings of the National Academy of Sciences*, vol. 99, n° 12, p. 7821-7826, 2002.

Gleiser P., Danon L., « Community Structure in Jazz », *Advances in Complex Systems*, vol. 6, p. 565, 2003.

Gregory S., « An algorithm to find overlapping community structure in networks », *Knowledge discovery in databases : PKDD 2007*, Springer, p. 91-102, 2007.

Gregory S., « Finding overlapping communities in networks by label propagation », *New Journal of Physics*, vol. 12, n^o 10, p. 103018, 2010.

Krebs V., « Books about US politics : », , `http://www.orgnet.com/`, 2004.

Lancichinetti A., Radicchi F., Ramasco J. J., Fortunato S., « Finding statistically significant communities in networks », *PloS one*, vol. 6, n^o 4, p. e18961, 2011.

Lusseau D., Schneider K., Boisseau O. J., Haase P., Slooten E., Dawson S. M., « The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations », *Behavioral Ecology and Sociobiology*, vol. 54, n^o 4, p. 396-405, 2003.

Newman M. E. J., « Finding community structure in networks using the eigenvectors of matrices », *Physical review E*, 2006. cite arxiv :physics/0605087Comment : 22 pages, 8 figures, minor corrections in this version.

Newman M. E. J., Girvan M., « Finding and evaluating community structure in networks », *Phys. Rev. E*, vol. 69, n^o 2, p. 026113, February, 2004.

Nicosia V., Mangioni G., Carchiolo V., Malgeri M., « Extending the definition of modularity to directed graphs with overlapping communities », *Journal of Statistical Mechanics : Theory and Experiment*, vol. 2009, n^o 03, p. P03024, 2009.

Palla G., Derenyi I., Farkas I., Vicsek T., « Uncovering the overlapping community structure of complex networks in nature and society », *Nature*, vol. 435, n^o 7043, p. 814-818, June, 2005.

Raghavan U. N., Albert R., Kumara S., « Near linear time algorithm to detect community structures in large-scale networks », *Physical Review E*, vol. 76, n^o 3, p. 036106, 2007.

Seifi M., Junier I., Rouquier J.-B., Iskrov S., Guillaume J.-L., « Stable community cores in complex networks », *Complex Networks*, Springer, p. 87-98, 2013.

Watts D., Strogatz S., « Collective dynamics of small-world networks », *Nature*, n^o 393, p. 440-442, 1998.

Xie J., Kelley S., Szymanski B. K., « Overlapping community detection in networks : The state-of-the-art and comparative study », *ACM Computing Surveys (csur)*, vol. 45, n^o 4, p. 43, 2013.

Xie J., Szymanski B. K., Liu X., « Slpa : Uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process », *Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on*, IEEE, p. 344-349, 2011.

Zachary W., « An information flow model for conflict and fission in small groups », *Journal of Anthropological Research*, vol. 33, p. 452-473, 1977.