

UNIVERSITY NAME

DOCTORAL THESIS

Conversations, Groupes et Communautés dans les Flots de Liens

Auteur :
Noé GAUMONT

Directeurs :
Clémence MAGNIEN
Matthieu LATAPY

*A thesis submitted in fulfillment of the requirements
for the degree of Doctor of Philosophy*

in the

Research Group Name
Department or School Name

30 mai 2016

Table des matières

Acknowledgements	ii
1 Introduction	3
1.1 Communauté dans les graphes	3
1.2 Extension temporelle	3
2 Expected Nodes : communautés de liens dans les graphes statiques	5
2.1 Travaux existants	5
2.2 Définition d'Expected Nodes	8
2.3 Comparaison	10
2.3.1 Cas du graphe complet	10
2.3.2 Graphe LFR	11
2.4 Calcul et optimisation	14
2.5 Conclusion	15
3 Extensions temporelle des graphes	17
3.1 Ètat de l'art des formalismes et outils	17
3.2 L'approche flot de liens	17
3.2.1 Définition	17
3.2.2 Sous-flot induit	17
3.2.3 Degré et densité	18
3.3 Manipulation de flots	18
3.3.1 Représentation	18
Liste des notations	19
4 Étude d'une archive de courriels	21
4.1 Prétraitement sur le jeu de données	22
4.2 Caractéristiques élémentaires des discussions	22
4.3 Étude des discussions en tant que sous-flots	24
4.3.1 Application de la Δ -densité	24
4.3.2 Répartition temporelle et structurelle des discussions	28
4.3.3 Flot quotient	29
4.3.4 Conclusion	30
4.4 Détection de structures denses	31
4.4.1 Méthode de détection	32
4.4.2 Comparaison des partitions	33
4.4.3 Conclusion	35
5 Détection de groupes denses (SNAM)	37
5.1 Calcul des groupes candidats	37
5.2 Calcul évaluation	37
5.3 Jeux de données	37
5.4 Application	37

5.5 Conclusion	37
6 Fonction de qualité	39
6.1 Définition	39
6.2 Générateur de flots de liens avec structure communautaire	39
7 Conclusion	41

Les axes des figures sont pour l'instant en anglais.

Tant que la bilbio n'est pas en version finale. Il n'est pas nécessaire de vérifier le formatage.

Chapitre 1

Introduction

Sommaire

1.1	Communauté dans les graphes	3
1.2	Extension temporelle	3

1.1 Communauté dans les graphes

[9]

1.2 Extension temporelle

Chapitre 2

Expected Nodes : communautés de liens dans les graphes statiques

Sommaire

2.1	Travaux existants	5
2.2	Définition d'Expected Nodes	8
2.3	Comparaison	10
2.3.1	Cas du graphe complet	10
2.3.2	Graphe LFR	11
2.4	Calcul et optimisation	14
2.5	Conclusion	15

Les structures communautaires dans les graphes ont été beaucoup étudiées lorsque la structure concerne les nœuds mais également, dans une moindre mesure, pour les liens. Les partitions de nœuds d'un graphe trouvent leurs limites lorsque les communautés se chevauchent. Dans ce cas, il est plus intéressant à ce que chaque nœuds puisse appartenir à plusieurs communautés. On manipule alors une partition chevauchante ou couverture. Pour répondre à ce problème de nombreux algorithmes ont été proposés pour la détection et l'évaluation de couvertures de nœuds. Cependant les couvertures, en tant que généralisation des partitions, sont encore plus difficiles à évaluer que les partitions. Les partitions de liens, quant à elles, restent des objets plus simple à manipuler. De plus, elles permettent de mettre en avant une autre structure ayant également du sens.

Dans un réseau sociale, chaque personne a plusieurs centres d'intérêts : famille, sport, politique... Lorsque deux personnes sont communiquent, la communication a lieu dans un contexte bien particulier. Bien que les personnes ont plusieurs centres d'intérêts, la raison de la communication est unique. Il semble donc qu'une information importante soit intrinsèquement liée au lien. Via la recherche de partitions de liens d'un graphe, c'est cette information que nous cherchons à capturer.

Il apparait donc les partitions de liens sont des objets à part entières pertinent à étudier. Il est alors nécessaire d'adapter les outils d'analyses pour évaluer directement les partitions de liens. Nous développons ici une approche similaire à ce qui est fait pour les partitions de nœuds et la modularité [16]. Le but est de créer une fonction de qualité permettant d'évaluer une partition de liens d'un graphe.

2.1 Travaux existants

Les notations utilisées sont les suivantes. Soit $G = (V, E)$ un graphe non-orienté avec V l'ensemble des nœuds de taille n et $E \subseteq V \times V$ l'ensemble des liens de taille m . Le degré d'un sommet u de G est noté $d_G(u)$. Une partition des liens en k groupes

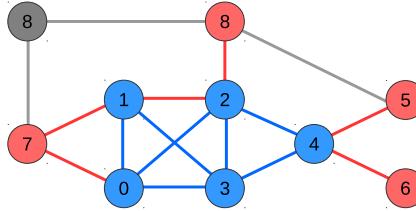


FIGURE 2.1 – Exemple d'un groupe de liens L (en bleu). Les liens rouges sont les liens adjacents L_{out} connectant les nœuds internes V_{in} (en bleu) aux nœuds adjacents V_{out} (en rouge).

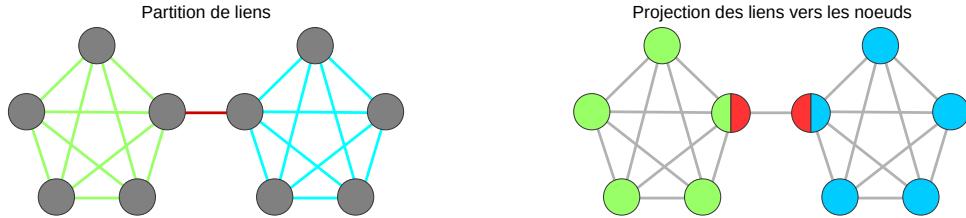


FIGURE 2.2 – Transformation d'une partition de liens à gauche en couverture de nœuds à droite. La couleur représente un groupe.

est notée $\mathcal{L} = (L_1, L_2, \dots, L_k)$ avec $L_i \subseteq E \forall i, L_i \cap L_j = \emptyset \forall i \neq j$ et $\bigcup_i L_i = E$. Pour un groupe de liens $L \in \mathcal{L}$, on pose $V_{in} = \{u \in V, \exists (u, v) \in L\}$ l'ensemble des nœuds internes au groupe L , $V_{out} = \{u \in V \setminus V_{in}, (u, v) \in E \wedge v \in V_{in}\}$ représente les nœuds adjacents au groupe L et enfin $L_{out} = \{(u, v) \in E \setminus L, u \in V_{in} \vee v \in V_{in}\}$ l'ensemble des liens adjacents au groupe L (voir Figure 2.1).

Il existe différent type de méthodes existantes. Il y a les méthodes évaluant une partition de liens via la transformation de la partition en une couverture de nœuds [10, 15, 19]. Il serait tentant de considérer que les partitions de liens et les couvertures de liens sont équivalentes. Ainsi pour évaluer une partition de liens, il suffirait de transformer la partition en couverture. Or, ce changement n'est pas anodin. D'une part, les couvertures de nœuds permettent de modéliser beaucoup plus de situations car il n'y a aucune contrainte sur les couvertures. D'autre part, il n'est pas trivial de transformer une partition de liens en couverture de nœuds, et *vice versa*.

Dans la figure 2.2, est présenté une transformation basique d'une partition de liens en une couverture. Dans cette transformation, un nœuds dans la couverture prends comme communauté l'ensemble des communautés de ses liens. Dans cet exemple, il est évident que la communauté rouge constituée des deux nœuds centraux n'est pas une communauté légitime et qu'il s'agit d'un artefact de la transformation. La transformation d'une partition n'est donc pas un acte neutre. Cet aspect a d'ailleurs été mis en avant par Esquivel *et al* ???. Face à ce problème, nos travaux ainsi que quelques méthodes existantes proposent des méthodes évaluant directement les partitions de liens.

Ahn *et al.* [1] sont parmi les premiers à avoir proposé une méthode détectant les communautés de liens. Leurs méthodes *link clustering* est une méthode hiérarchique d'agglomération. Elle construit un dendrogramme en agglomérant de manière itérative les groupes de liens en fonction de leurs similarités calculée par l'indice de Jaccard. Afin de décider de la coupe du dendrogramme et de la partition résultante, la fonction *partition density* est utilisée. Pour une partition de liens donnée \mathcal{L} , la *partition*

density est définie de la manière suivante :

$$D(\mathcal{L}) = \frac{\sum_{L \in \mathcal{L}} |L| D(L)}{m} \quad D(L) = \frac{|L| - \min_D(|V_{in}|)}{\max_D(|V_{in}|) - \min_D(|V_{in}|)}, \quad (2.1)$$

Il s'agit de la moyenne pondérée de la qualité de chaque groupe. La qualité d'un groupe est le nombre de liens du groupe normalisé par le nombre de liens minimum et maximum pour un groupe de $|V_{in}|$ nœuds. Le nombre minimum de liens est obtenu par un arbre : $\min_D(N) = N - 1$. Le nombre maximum est obtenu par une clique : $\max_D(N) = \frac{N(N - 1)}{2}$.

Après simplification, on obtiens la formule suivante :

$$D(L) = 2 \frac{|L| - (|V_{in}| - 1)}{(|V_{in}| - 1)(|V_{in}| - 2)}. \quad (2.2)$$

D'autres chercheurs [14, 18] ont par la suite utilisé la *partition density* comme fonction à optimiser dans un algorithme génétique. Leurs solutions semblent pour l'instant difficile car leurs algorithme reposent sur de nombreux critères et limité à de petit graphes.

Par ailleurs, la *partition density* ne peut pas être directement appliquée aux graphes pondérés. Une première proposition a été faite par Kim [11].

Evans *et al.* [8] propose trois fonctions de qualité pour évaluer les partitions de liens. Leurs fonctions de qualité sont basées sur trois marches aléatoires qui se déroulent sur les liens du graphe. L'approche est similaire à la modularité car la modularité peut également être définie à l'aide de marche aléatoire sur les nœuds du graphe [5]. Leurs trois fonctions de qualités peuvent être calculées et optimisées sur le graphe mais les auteurs ont montré que l'on pouvait, de manière complètement équivalente, utiliser la modularité sur des line graphe pondérés (LG_1 , LG_2 , LG_3). Ainsi, il suffit de construire le line graphe approprié puis d'utiliser un algorithme existant d'optimisation de la modularité tel que l'algorithme de *Louvain* [3]. Un line graphe est un graphe où chaque lien du graphe initial est transformé en un nœuds dans le line graphe. Deux nœuds du line graphe sont reliés si les liens correspondant ont au moins un nœuds en commun.

Pour construire les lines graphes LG_1 , LG_2 et LG_3 , nous définissons $B \in \mathcal{M}_{n,m}$ la matrice d'incidence du graphe G : un élément $B_{i\alpha}$ de cette matrice $|V| \times |E|$ est égale à 1 si le lien α est relié au nœuds i et 0 sinon. Les matrices LG_1 , LG_2 et LG_3 sont alors définies de la manières suivantes :

	$x = 1$	$x = 2$	$x = 3$
$LG_x(\alpha, \beta)$	$B_{i\alpha}B_{j\beta}(1 - \delta_{\alpha\beta})$	$\sum_{i \in V, d_G(i) > 1} \frac{B_{i\alpha}B_{i\beta}}{d(i) - 1}$	$\sum_{i,j \in V, d(i)d(j) > 0} \frac{B_{i\alpha}A_{ij}B_{j\beta}}{d(i)d(j)}$

Soit $k_x(\alpha) = \sum_{\beta} LG_x(\alpha, \beta)$ le degré pondéré dans le line graphe LG_x du nœuds représentant le lien α et $W_x = \sum_{\alpha, \beta \in |E|} LG_x(\alpha, \beta)$ la somme des poids des liens. Pour $x \in \{1, 2, 3\}$, la fonction de qualité $Evans_x$ est définie de la manières suivante :

$$Evans_x(\mathcal{L}) = \frac{1}{W_x} \sum_{L_i \in \mathcal{L}} \sum_{e_1, e_2 \in L_i^2} LG_x(e_1, e_2) - \frac{k_x(e_1)k_x(e_2)}{W}. \quad (2.3)$$

Kim *et al.* [12] ont exploré une extension du concept *Minimum Length Description* (MDL) introduit par Rosvall *et al.* [17] qui méthode provenant de la théorie de

l'information. Cette extension de la *MDL* évalue directement une partition de liens, contrairement à l'extension proposée par Esquivel *et al.* [7]. Un avantage de leur méthode est de pouvoir comparer l'avantage d'utiliser une partition de liens ou une partition de nœuds avec leur *MDL* respective. Cependant, leur méthode semble favoriser les communautés de liens que dans des cas très précis.

2.2 Définition d'Expected Nodes

Une idée souvent utilisée lors de la détection de communautés de nœuds est qu'une communauté devrait avoir beaucoup de connexion en interne. Pour appliquer ce genre de définition intuitive, il est nécessaire de définir à quoi comparer le nombre de connexion interne. Le choix qui est fait avec la modularité et d'autres méthodes est de définir un modèle nul aléatoire où il n'existe pas de structure communautaire. Le but est de construire un graphe aléatoire qui partage un certains nombre de caractéristiques avec le graphe initial mais dont la structure a été détruite lors du mélange.

Il existe de nombreux modèles et celui utilisé dans la modularité est le modèle de configuration [2]. Dans ce modèle, le nombreux de nœuds et leur degré sont fixes mais la répartition des liens est aléatoire. Ainsi pour chaque nœuds, ses voisins sont tirés de manière aléatoire avec une probabilité proportionnelle à leur degré. Comme les liens sont mélangés dans ce modèle, on suppose qu'il n'existe plus de structure communautaire dans le graphe.

Pour notre fonction de qualité *Expected Nodes*, nous utilisons également le modèle de configuration. Avant d'aller plus loin et de définir formellement *Expected Nodes*, il est utile d'avoir une définition informelle de la fonction de qualité.

Le but est d'évaluer un groupe de liens. Afin qu'un groupe de liens soit évalué comme une bonne communauté, les liens devraient induire un nombre relativement faible de nœuds internes. En effet, plus le nombre de nœuds internes est faible, plus le groupe de liens ressemble à une clique. De manière similaire à la modularité, nous utilisons le configuration modèle pour calculer le nombre de nœuds interne espéré dans un modèle nul. Si le groupe de liens a moins de nœuds internes qu'espéré alors ça indique que le groupe de liens est plus dense et qu'il devrait donc avoir une évaluation élevée.

Il est donc nécessaire de calculer l'espérance du nombre de nœuds interne, μ_G , d'un groupe de liens, L , dans le modèle nul. Un nœuds de G est interne si au moins un de ses demi-liens appartient à L . Ainsi pour calculer μ_G , il faut tirer aléatoirement et sans remise $2|L|$ demi-liens parmi les $2|E|$ demi-liens du graphe aléatoire avec la même distribution de degrés. Soit B_u la variable aléatoire correspondant au nombre de fois où le sommet u est tiré. Cette variable suit une loi hypergéométrique $B_u \sim \mathcal{G}(2|E|, d_G(u), 2m)$. Avec cette notation, on définit μ_G de la manière suivante :

$$\mu_G(|L|) = \sum_{u \in V} \mathbb{P}(B_u \geq 1) = \sum_{u \in V} 1 - \frac{\binom{2|E|-d(u)}{2|L|}}{\binom{2|E|}{2|L|}}. \quad (2.4)$$

Voici quelques propriétés de la fonction $\mu_G(|L|)$:

- La fonction μ_G dépend uniquement de la séquence de degrés $\{d_G(v)\}_{v \in V}$. On peut montrer que cette fonction est Schur-concave, ainsi plus les degrés sont uniformément répartis plus il sera surprenant d'observer un groupe de liens correspondant à peu de nœuds.

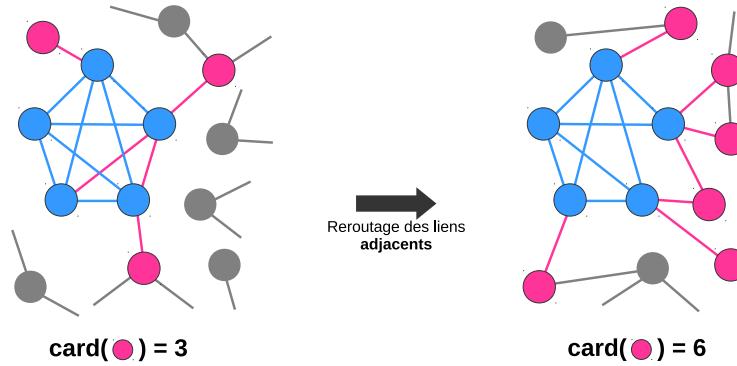


FIGURE 2.3 – Groupe de liens L en bleu et ces liens adjacents en rouges dans le graphe initial à gauche. À droite, une réalisation du modèle de configuration où L a été figé.

- Pour une distribution de degrés donnée, la fonction $\mu_G(|L|)$ est une fonction croissante de $|E|$.
 - Si $L = E$, alors le nombre de nœuds attendus est bien $|V|$.
 - On a $\mu_G(1) \leq 2$, en effet le modèle nul n'interdit pas la présence de boucles.
- Avec μ_G , nous pouvons définir la qualité *interne*, Q_{in} d'un groupe de liens L :

$$Q_{in}(L) = \frac{\mu_G(|L|) - |V_{in}(L)|}{\mu_G(|L|)}. \quad (2.5)$$

Avec cette formulation, pour un groupe de taille $|L|$, plus le nombre de nœuds interne est faible, plus Q_{in} sera élevée.

Q_{in} permet d'évaluer la qualité interne d'un groupe mais il faut aussi tenir compte du voisinage. C'est pourquoi, nous définissons également une qualité externe. Le but est d'évaluer comment sont répartis les liens et nœuds adjacents. Pour ce faire, nous allons également comparer le nombre de nœuds adjacents observés aux nombre espéré dans le configuration modèle. Cependant à l'inverse de la qualité interne, la qualité externe est mauvaise si jamais le nombre de nœuds adjacent est plus faible qu'espéré. En effet, si il y a beaucoup de liens adjacents pour peu de nœuds adjacents, alors cela indique que le voisinage du groupe est dense et devrait être inclus dans le groupe. Le cas idéal est quand chaque lien adjacent est relié à un nœuds différent.

Soit $\bar{d}(L, u) = \sum_{v \in V} \mathbf{1}_{(u,v) \in E \setminus L}$ le degré de u limité aux liens adjacents et $\bar{d}(L) = \sum_{u \in V_{in}(L)} \bar{d}(L, u)$. L'espérance du nombre de nœuds adjacents est calculé comme le nombre de nœuds qui sont tirés lorsque $\bar{d}(L)$ demi-liens sont choisis aléatoirement et sans remise dans modèle de configuration où les liens de L ont été préalablement retirés. Ce graphe aléatoire a la distribution de degré suivante : $\{d_{G \setminus L}(u)\}_{u \in V}$ où $G \setminus L = (V, E \setminus L)$. Dans ce cas, on ne tire pas aléatoirement un lien mais uniquement un demis-liens car l'autre demi-lien est un des demi-liens reliés aux nœuds internes. L'espérance du nombre de nœuds adjacents se définit de la manière suivante :

$$\mathbb{E}[\bar{d}(L)] = \mu_{G \setminus L}(d(\bar{L})/2). \quad (2.6)$$

Une illustration de ce processus est présentée dans la figure 2.3. Sur cette illustration, le groupe L a un très mauvais voisinages et cela se reflète par un nombre de nœuds adjacents observés plus faible qu'espéré.

Comme il est intéressant de pénaliser les groupes ayant de mauvais voisinage mais qu'un bon voisinage n'est pas suffisant pour définir une bonne communauté, nous bornons à 0 la qualité externe :

$$Q_{ext}(L) = \min \left(0, \frac{|V_{out}(L)| - \mu_{G \setminus L}(\bar{d}(L)/2)}{\mu_{G \setminus L}(\bar{d}(L)/2)} \right). \quad (2.7)$$

Enfin, nous définissons *Expected Nodes* pour un groupe L :

$$Q(L) = 2 \frac{|L|Q_{in}(L) + |L_{out}|Q_{ext}(L)}{|L| + |L_{out}|}. \quad (2.8)$$

Nous utilisons la moyenne pondérée entre la qualité interne et externe car la qualité interne est dû aux liens de L et la qualité externe par les liens adjacents. Nous détaillons certaines propriétés des formules 2.7 et 2.8 découlant des propriétés de μ_G :

- En s'intéressant aux nœuds adjacents V_{out} , on pénalise la présence de nœuds adjacents fortement connectés avec les nœuds incidents à L .
- Ainsi la qualité d'un lien isolé dépend du nombre de triangles dans laquelle il se trouve. Un lien séparant deux groupes de nœuds disjoints peut avoir une qualité positive.
- La qualité du groupe contenant tout les liens est nulle.

Nous définissons *Expected Nodes* pour une partition de liens \mathcal{L} comme la moyenne pondérée de la qualité de chaque groupe :

$$Q_G(\mathcal{L}) = \frac{\sum_{L \in \mathcal{L}} |L|Q(L)}{|E|}. \quad (2.9)$$

2.3 Comparaison

Nous évaluons maintenant *Expected Nodes* en utilisant deux jeux de test. Sur ces jeux de tests, nous appliquons également des fonctions de qualités reconnues : *partition density* [1] et les fonctions de qualité proposées par Evans *et al.* [8] que nous nommons *Evans1*, *Evans2* et *Evans3*. Pour chaque graphe de test, nous créons empiriquement plusieurs partitions de liens et nous évaluons chaque partition avec toutes les fonctions de qualité.

2.3.1 Cas du graphe complet

Le premier jeu de test est assez simple puisqu'il s'agit d'un graphe complet. Le but est de vérifier que *Expected Nodes* n'a pas un comportement dégénéré. Nous étudions un graphe complet de 100 nœuds¹. Sur ce graphe, nous définissons plusieurs partitions. La première est une partition triviale où tous les liens sont dans un unique groupe. Nous définissons deux familles de partitions : une séparant les liens en deux groupes et une séparant les liens en 3 groupes. Soit V' un ensemble de p nœuds où p est un paramètre $p < |V|$. Les deux familles de partitions placent les liens de $V' \times V'$ dans un groupe. Pour la partition en 2 groupes, tous les autres liens sont mis dans un second groupe. Pour la partition en 3 groupes, les liens de $V \times V \setminus V'$ sont dans un second groupe et le reste dans un troisième. Ces répartitions sont illustrées dans la figure 2.4.

1. Nous avons obtenus des résultats similaires pour un graphe de 500 nœuds.

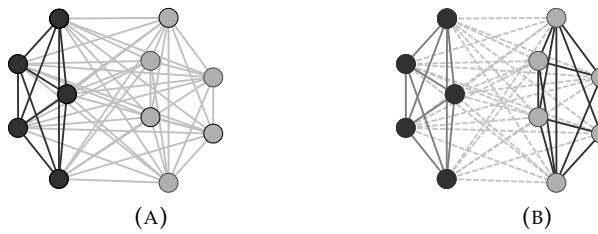


FIGURE 2.4 – Deux partitions de liens pour un graphe complet à 10 nœuds avec $p = 5$: (a) partition en deux groupes et (b) partition en trois groupes. Les nœuds noirs sont les nœuds appartenant à V' et la couleur d'un lien correspond à son groupe.

Comme le graphe est un graphe complet, la meilleure solution est d'avoir un seul groupe contenant l'ensemble des liens, *i.e.* la partition triviale devrait avoir une meilleure évaluation que les autres partitions. La figure 2.5 présentent les résultats. Pour chaque valeur de p et chaque fonction de qualité, nous calculons les évaluations des partitions en deux et en trois groupes ainsi que l'évaluation de la partition triviale. L'évaluation de la partition trivial n'est pas dépendante de p et est calculée qu'une seule fois. De manière assez surprenante, les fonctions *Evans1* et *Evans2* ne passent pas ce test car elles évaluent la partition en deux ou trois groupes comme meilleur que la partition trivial. Selon la *partition density*, *Expected Nodes* et E_3 , la partition trivial est la meilleure des partitions. La fonction *Evans3* diffère légèrement car elle a une amplitude plus faible ($\approx 10^{-3}$).

2.3.2 Graphe LFR

Nous utilisons maintenant un jeu de test plus évolué. Il n'existe pas à notre connaissance de générateur de graphe aléatoire avec une structure communautaire sur les liens. C'est pourquoi, nous utilisons le générateur proposé par Lancichinetti *et al.* [13]. Ce générateur aléatoire permet de générer des graphes ayant une structure communautaire chevauchante sur les nœuds. Comme nous voulons évaluer une partition de liens, il est nécessaire de transformer cette vérité de terrain. Nous introduisons deux transformations de la couverture des nœuds en deux partitions de liens, TA et TB , voir figure 2.6.

Soit $u, v \in V$, $C_{u,v}$ désigne l'intersection des communautés de u et v dans la couverture et $U_{u,v}$ désigne leur union. Nous définissons le groupe d'un lien $(u, v) \in E$ dans TA et TB de la manière suivante :

- intra-communauté** si $|C_{u,v}| = 1$ alors (u, v) est dans la communauté $C_{u,v}$;
- inter-communauté** si $|C_{u,v}| = 0$ alors dans TA , (u, v) appartient à sa propre communauté. Dans TB , le lien appartient à la communauté $U_{u,v}$, qui contient l'ensemble des liens (u', v') tel que $U_{u',v'} = U_{u,v}$;
- chevauchement** si $|C_{u,v}| > 1$ alors le lien (u, v) appartient aléatoirement à une des communautés appartenant à $C_{u,v}$.

Pour générer le graphe, nous avons appliqué un jeu de paramètre classiquement utilisé dans la littérature [9]. Ainsi, nous avons généré des graphes de 500 nœuds ayant un degré moyen de 25, un degré max de 50 et 10 nœuds appartenant à deux communautés et des communautés ayant une taille comprise entre 20 et 100. Le degré est tiré selon une loi exponentielle de paramètre -2 et la taille des communautés a -1 comme paramètre. Enfin, 90% des liens se sont à l'intérieur d'une communauté et les 10% restants sont répartis de manière aléatoire. Avec ces paramètres, il

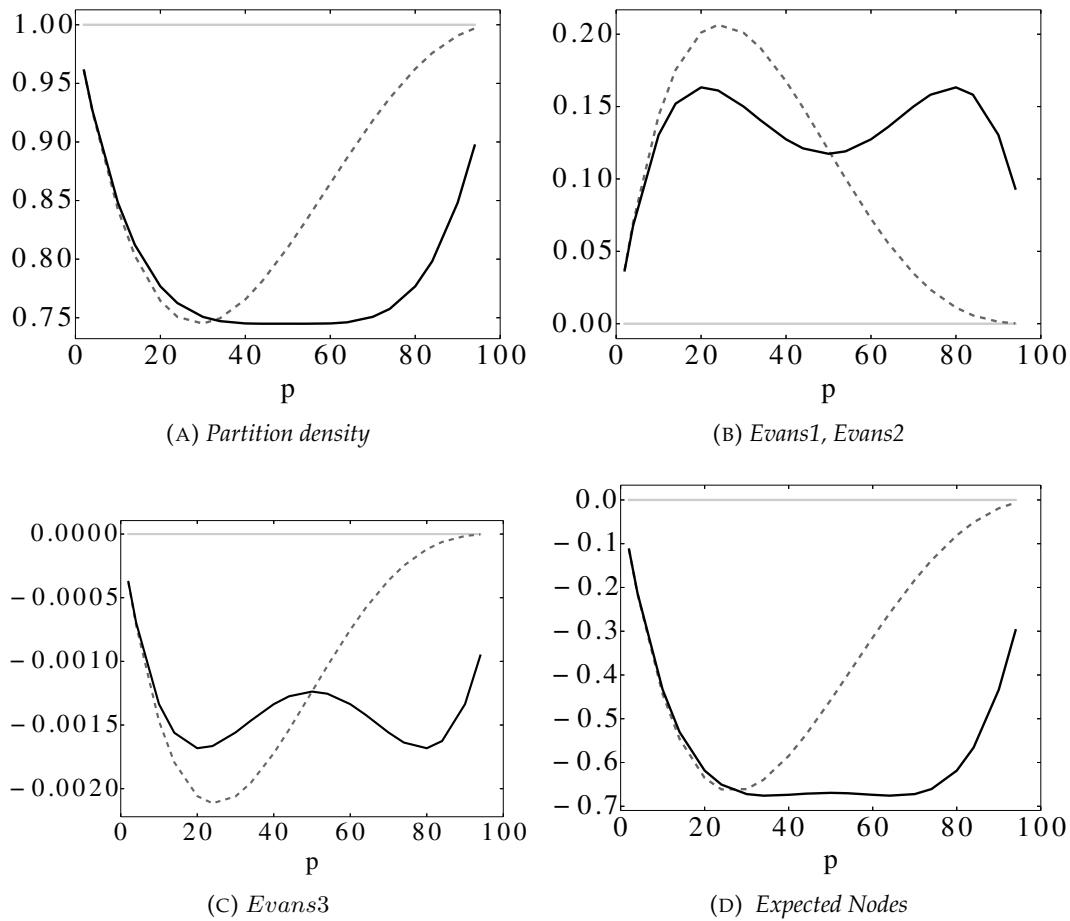


FIGURE 2.5 – Évaluation des 5 fonctions de qualité sur un graphe complet de 100 nœuds pour trois type de partitions. Les partitions testées sont présentée dans la section 2.3.1. Par définition, les résultats pour *Evans1* et *Evans2* sont identiques. Les lignes en gris, noir et pointillé représentent respectivement la partition triviale, les partition en deux groupes et les partitions en trois groupes.

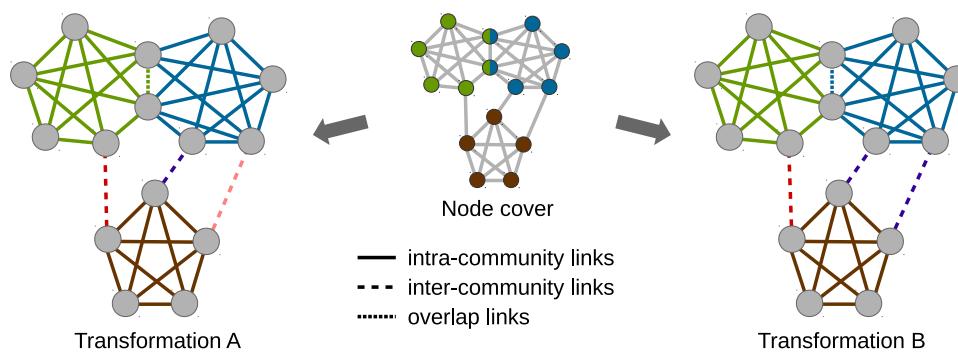


FIGURE 2.6 – Construction de TA et TB depuis une couverture de nœuds. La couleur des liens indique leur groupe.

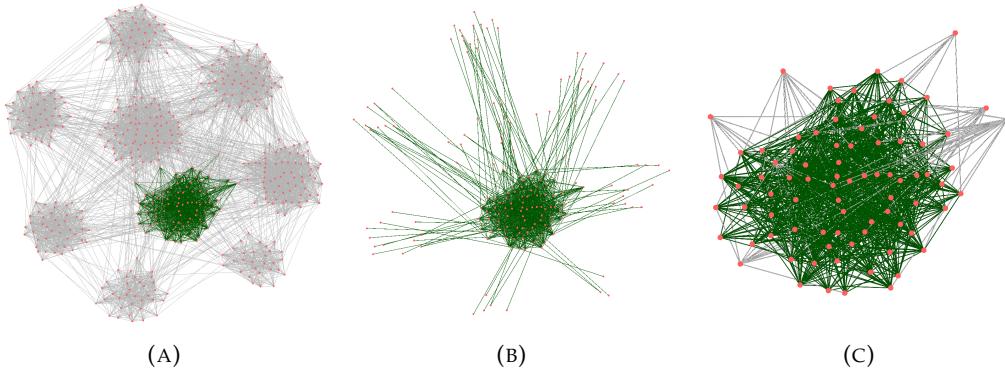


FIGURE 2.7 – Exemple de graphe généré par le LFR en (A) avec une communauté de la vérité de terrain mise en avant en vert. En (B) zoom sur une communauté détectée par $E2$ dont les liens sont en verts. En (C) zoom sur une communauté détectée par LC dont les liens sont en verts.

y a en moyenne 5620 liens intra-communautés, 625 liens inter-communauté et seulement 5 liens chevauchant.

Pour chaque graphe généré, nous testons les partitions TA et TB mais aussi la partition LC trouvée par *link clustering* [1] et la partition $E2$ trouvée par la seconde méthode de Evans *et al.* [8]². Ces deux algorithmes optimisent respectivement *Partition Density* et *Evans2*. Ces partitions sont ensuite évaluées par les fonctions de qualité *Partition Density*, *Evans2* et *Exoected Node*. Une illustration d'un graphe généré et des exemples de groupes capturés par LC et $E2$ sont présentés dans la figure 2.7.

Dans LC , TA , TB et $E2$, il y a 720, 650, 70 et 11 groupes en moyenne. Afin d'observer la ressemblance de ces partitions, nous avons utilisé la NMI [4]. Il apparaît que les partitions TA et TB sont les plus proches. Ensuite, nous remarquons que la partition $E2$ diffère de TA et de TB uniquement sur les liens inter-communautés. En effet si ils ne sont pas pris en compte lors de la comparaisons, alors $E2$, TA et TB sont équivalentes. Il semble en effet que les liens inter-communautés soient arbitrairement distribués entre les plus grosses communauté adjacentes, ce qui est visible dans la figure 2.7b. Enfin, la partition LC , bien que proche de TA et TB , est un peu plus différente. Ces 720 groupes semblent plus petit mais aussi plus denses que ceux de TA ou TB . En particulier, les liens intra-communautés peuvent être séparés dans plusieurs groupes, comme dans la figure 2.7c. Les quatres partitions sont donc différentes et mettent en avant différentes caractéristiques.

Nous procédons maintenant à l'évaluation de ces partitions par les différentes fonctions de qualités. Comme le processus de génération de graphe est aléatoire, les évaluations présentées dans la figure 2.8 représentent 30 générations. On remarque tout d'abord que ni TA ni TB n'a la meilleure évaluation selon *Evans2* (figure 2.8b) ou *Partition density* (figure 2.8a) même si TA et TB représente nos vérités de terrains. Dans le cas de *Partition Density*, TA , TB et LC ont quasiment la même évaluation alors qu'il s'agit de structures assez différentes. Dans le cas de *Evans2*, c'est la partition $E2$ qui obtient la meilleure évaluation. Cela prouve l'efficacité de l'algorithme $E2$ pour optimiser *Evans2* mais remets en cause la pertinence du critère $E2$ pour mettre en avant la vérité de terrain. Enfin les partitions TA et TB ont encore une fois la même évaluation. Notre fonction *Expected Nodes* se comporte différemment des 2 autres. Tout d'abord, c'est la vérité de terrain TA qui obtient la meilleure

2. The results are similar for the algorithms using $E1$ and $E3$.

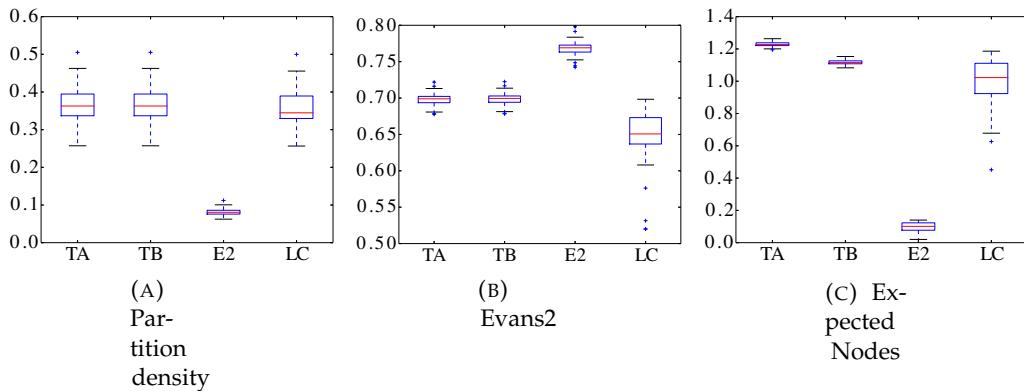


FIGURE 2.8 – Boite à moustache des évaluations des trois fonctions de qualité pour les différentes partitions de liens. La boite représente le premier et troisième quartile ainsi que la médiane. Les moustaches s'étendent sur 1.5 fois l'écart interquartile. Les croix sont les points au-delà des moustaches.

évaluation puis il s'agit de *TB* ou *LC* selon les générations. Notre mesure semble donc bien mettre en avant la vérité de terrain générée. De plus, *Expected Nodes* évalue différemment les partitions *TA* et *TB*. C'est un point important car, dans la partition *TB*, les liens inter-communautés peuvent donner lieu à des groupes non-connexes dans *TB*, voir figure 2.6. Ce phénomène est très pénalisé par *Expected Nodes*. La partition *E2* obtient une mauvaise évaluation également à cause des liens inter-communautés. En effet en les fusionnant à une communauté adjacente, cela augmente fortement le nombre de nœuds interne ce qui fait baisser Q_{in} mais Q_{ext} baisse également.

Pour ces raisons, nous pensons que *Expected Nodes* est une mesure qui permet de mieux évaluer les partitions de liens.

2.4 Calcul et optimisation

Jusqu'à maintenant, nous avons évalué la mesure sans nous attacher ni à son calcul ni à son optimisation. Nous discutons maintenant de la complexité de calcul d'*Expected Nodes* pour une partition donnée. Le calcul de la qualité interne 2.5 nécessite d'évaluer la probabilité qu'un nœuds soit tiré. Or, cette probabilité ne dépend que du degré du nœuds et du nombre de liens dans le groupe. Donc, tous les nœuds ayant le même degré donnent lieu à la même probabilité. Ce calcul de probabilité

nécessite d'évaluer pour un nœuds u : $1 - \frac{\binom{2|E|-d(u)}{2|L|}}{\binom{2|E|}{2|L|}}$, ce qui peut être assez coûteux.

Ce calcul correspond à une loi hypergéométrique. Or sous certaines conditions, une loi hypergéométrique peut être approché par une loi binomial ce qui simplifie le calcul à : $1 - (1 - \frac{|L|}{|E|})^{|L|}$ ce qui est plus rapide. Lors de nos tests, le biais induit par cette approximation reste assez faible, de l'ordre de 0.01% en moyenne. Ce changement de calcul est équivalent à considérer un tirage avec remise au lieu de tirage sans remise.

Si l'on considère que l'évaluation de la probabilité peut se faire en $O(1)$ alors, pour un groupe donné, il est possible de calculer sa qualité en $O(|\{d_G(v)\}_{v \in V}|)$. Cette

propriété est très pratique lorsque beaucoup de noeuds ont le même degré. Enfin comme la qualité d'un groupe ne dépend que de sa taille, on peut calculer la qualité d'une partition \mathcal{L} en $O(\{|L_i|\}_{L_i \in \mathcal{L}})$.

Il est donc assez rapide de calculer Q_{in} . En revanche, la situation est complètement différente pour Q_{ext} . Le processus de calcul est similaire. On peut de nouveau appliquer l'approximation de la loi hypergéométrique par une loi binomiale mais deux noeuds de même degré ne vont plus forcément avoir la même probabilité d'être tiré. En effet, Q_{ext} n'utilise pas le graphe initial mais le graphe $G \setminus L_i$. Pour chaque noeuds, il faut évaluer son degré dans ce nouveau graphe. Le calcul de Q_{ext} pour un groupe se fait donc en $O(n)$. Pour l'évaluation d'une partition, il n'est pas non plus possible de considérer comme équivalent des groupes de même tailles. Le coût pour évaluer une partition est donc en $O(|\mathcal{L}|n)$. Le code pour évaluer une partition de liens d'un graphe selon notre mesure *Expected Nodes* mais aussi *Partition Density*, *Evans1*, *Evans2* et *Evans3* est disponible en ligne : <https://github.com/ksadorf/ExpectedNodes>.

Malgré ce coût élevé, nous avons développé un premier algorithme d'optimisation glouton de *Expected Node*. Le principe de fonctionnement est le suivant. Chaque lien est initialement dans son propre groupe. Puis à chaque itération, on considère de type de modification de la solution courante. Soit la meilleure fusion de deux communautés soit le meilleur changement de communauté d'un seul lien. On fusionne ou change de communauté un lien si cela améliore la qualité de la partition. Les fusions considérées sont les fusions entre des communautés adjacentes. Les changement de liens se font également que pour une communauté adjacente. On continue de modifier la solution courante tant qu'elle est améliorable par un de ces mouvements. Malheureusement cette approche souffre de deux problème majeurs. Tout d'abord le calcul du gain est couteux et rends impossible l'étude de grand jeu de donnée. Ensuite dans nos tests dans les graphes générés par LFR, il semble que cette méthode reste bloquée sur des optimum locaux bien plus faible que la vérité de terrain. Il faudrait donc tester d'autres heuristiques d'optimisation mais aussi travailler sur une méthode de calcul du gain plus optimisé.

Cet algorithme naïf nous a tout de même permis de vérifier que les vérités sont quasiment des optimum locaux vis-à-vis de nos modifications. En effet en utilisant les partitions TA et TB comme point de départ de l'algorithme, il n'est quasiment pas possible d'améliorer la qualité de la partition.

2.5 Conclusion

Nous considérons de nouveaux critères pour l'évaluation des partitions de liens tenant compte de la répartition des noeuds internes et externes d'un groupe. À partir de ces critères, nous définissons une mesure de qualité, *Expected Nodes*, basée sur la différence entre le nombre de noeuds induit par un groupe de lien et le nombre de noeuds attendu dans un modèle nul. Pour montrer la pertinence de cette nouvelle fonction de qualité, nous évaluons quatre fonctions de qualité de la littérature. Nous montrons que sur nos jeux de tests *Expected Nodes* semble être la plus à même de capturer la vérité de terrain. Le premier algorithme agglomératif d'optimisation glouton ne permet pas pour l'instant d'obtenir des partitions avec des scores élevés.

Chapitre 3

Extensions temporelle des graphes

Sommaire

3.1	Etat de l'art des formalismes et outils	17
3.2	L'approche flot de liens	17
3.2.1	Définition	17
3.2.2	Sous-flot induit	17
3.2.3	Degré et densité	18
3.3	Manipulation de flots	18
3.3.1	Représentation	18

3.1 État de l'art des formalismes et outils

3.2 L'approche flot de liens

3.2.1 Définition

Un flot de liens est défini comme un triplet : $\mathcal{L} = (T, V, E)$, où $T = [\alpha, \omega]$ est un intervalle de temps, V un ensemble de n noeuds et $E \subseteq T \times T \times V \times V$ un ensemble de m liens. Les liens de E sont des quadruplets (b, e, u, v) , signifiant que la paire de noeuds (u, v) est connectée sur l'intervalle $[b, e] \subseteq [\alpha, \omega]$. Nous considérons un flot non orienté, *i.e.* $(b, e, u, v) = (b, e, v, u)$, et sans boucle, *i.e.* $u \neq v$.

Nous dénotons la durée du flot par $\bar{L} = \omega - \alpha$. $\beta_E = \min_{(b,e,u,v) \in E}(b)$ et $\psi_E = \max_{(b,e,u,v) \in E}(e)$ sont respectivement l'apparition du premier lien et la disparition du dernier lien.

Un flot de liens est simple si pour tout $(b, e, u, v) \in E$ et $(b', e', u, v) \in E$, $[b, e] \cap [b', e'] = \emptyset$. La simplification d'un flot de liens $\sigma = L$.

$V(E') = u_{(b,e,u,v) \text{ in } E'}$ sommets induits par un ensemble de liens.

$\xi(L, \Delta)$ ajoute Δ à chaque lien.

Que des flots avec durées ou bien delta densité ?

3.2.2 Sous-flot induit

Sous flot induit par un ensemble de lien $E' : L(E') = ([\beta_{E'}, \psi_{E'}, V(E'), E']).$

Sous flot induit par un ensemble de paire noeuds $S \in V^2 : L(S). L(S) = ([\beta_{E'}, \psi_{E'}], V', E')$ avec $E' = \{(b, e, u, v) \in E, (u, v) \in S\}$. Par convention, on note $L(v) = L(\{v\}) \times V$.

Sous flot induit par un intervalle de temps $T' = [\alpha', \omega']$, $T' \subseteq T : L_{\alpha'.. \omega'} . L_{\alpha'.. \omega'} = ([\alpha', \omega'], V(E'), E')$ avec $E' = \{(b', e', u, v), \exists (b, e, u, v) \in E, b' = \max(b, \alpha'), e' = \min(e, \omega')\}$. Par convention, on note $L_{t..t} = L_t$

Il est aussi possible de combiner ces notions. Par exemple avec $V' \subset V$, $L_{\alpha'..,\omega'}(V'^2)$ est le sous flot correspondant au lien entre les nœuds de V' sur l'intervalle $[\alpha', \omega']$.

3.2.3 Degré et densité

Beaucoup de notions sont développées autour de cet objet. Degré temporelle d'un nœud u :

$$d_t(u) = |L_t(v)| = |\{(b, e, u, v) \in E, b \leq t \leq e\}| \quad (3.1)$$

Par extension pour un ensemble de sommets V' :

$$d_t(V') = |L_t(V'^2)| = |\{(b, e, u, v) \in E, u, v \in V', b \leq t \leq e\}| \quad (3.2)$$

Degré interne maintenant ?

$$d_t(E') = |\{(b, e, u, v) \in E', b \leq t \leq e\}| \quad (3.3)$$

Il est possible d'intégrer sur l'ensemble du temps

$$D_{\alpha..\omega}(v) = \int_{\alpha}^{\omega} d(v, t) dt = D(v) \quad (3.4)$$

Par convention, on notera le degré moyen de v : $d(v) = \langle d(v, t) \rangle = \frac{D(v)}{\omega - \alpha}$. Il en va de même pour les autres degrés.

C'est le cas de la densité :

$$\delta(L) = \frac{2 \sum_{l \in E}}{n(n-1)(\bar{L})} \quad (3.5)$$

3.3 Manipulation de flots

Existence de la lib

3.3.1 Représentation

Liste des notations

Symbol	description
L	Flot de liens
T	intervalle de temps
V	ensemble de nœuds
E	ensemble de liens : (b, e, u, v)
n	nombre de nœuds
$ L , E $	nombre de liens dans le flot
β_E	temps d'apparition du premier lien
ψ_E	temps de disparition du dernier lien
$\xi(L, \Delta)$	Flot de liens où chaque lien dure Δ
$L(V'^2)$	sous flot induits par les nœuds de V'
$L_{t..t'}$	sous flot induits par l'intervalle $[t, t']$
$d_t(v)$	degré de v à l'instant t
$d(v)$	degré moyen v sur T
$\delta(L)$	densité du flot
$\delta_\Delta(L)$	densité du flot ou chaque lien dure Δ

Chapitre 4

Étude d'une archive de courriels

Sommaire

4.1	Prétraitement sur le jeu de données	22
4.2	Caractéristiques élémentaires des discussions	22
4.3	Étude des discussions en tant que sous-flots	24
4.3.1	Application de la Δ -densité	24
4.3.2	Répartition temporelle et structurelle des discussions	28
4.3.3	Flot quotient	29
4.3.4	Conclusion	30
4.4	Détection de structures denses	31
4.4.1	Méthode de détection	32
4.4.2	Comparaison des partitions	33
4.4.3	Conclusion	35

L'étude de la structure des réseaux est un sujet qui est étudié depuis assez longtemps [REF](#). Ces études ont, dans un premier temps, permis de trouver comment caractériser une structure puis, dans un second temps, de proposer des méthodes de détections de ces structures. La littérature sur l'étude des flots de liens est encore récente et il n'existe que peu d'études [REF](#) sur les spécificités des structures dans les flots de liens.

Intro sur-
ement trop
général qui
sera bougée
ailleurs.

Nous nous intéressons ici à une archive de courriels publiquement disponibles¹. Cette archive contient l'ensemble des courriels échangés par différent utilisateurs pour résoudre un problème survenu lors de l'utilisation de Debian. Typiquement, une personne ayant un problème lors de l'installation envoie un courriel à la liste afin de demander de l'aide. Toute personne inscrite sur la liste reçoit ce courriel et peut y répondre ce qui donne lieu à une discussion visible par tous. Ces discussions ont déjà été étudiées dans le passé [6] mais cela a été fait en utilisant des méthodes statiques uniquement.

Or, ces données se représentent naturellement sous forme de flot de liens en associant chaque personne à un nœud et chaque courriel entre deux personnes à un lien dans le flot à l'instant où le courriel a été envoyé. L'avantage de ces données de communications est que nous connaissons la discussion (*thread*) dans laquelle a lieu chaque message. Une discussion est un ensemble de courriels dont tous les messages répondent à un message précédent de la discussion excepté pour le premier qui a initié la discussion et que nous appelons *racine*. Ainsi, nous étudions la structure des discussions dans le flot liens représentant les courriels envoyés sur la liste.

Utiliser le formalisme de flot de liens est particulièrement intéressant car cette liste de diffusion existe depuis 1994. L'aspect temporel des discussions est donc important.

1. <https://lists.debian.org/debian-user/>

4.1 Prétraitement sur le jeu de données

Bien qu'accessible sur internet, ce jeu de données nécessite un ensemble de traitements avant de pouvoir exploiter les 724985 courriels que contenait l'archive en janvier 2015. Tout d'abords, les données ne sont pas sous la forme d'un flot de liens avec la structure des conversations. Les données sont accessibles via le site internet et ne sont pas structurées. Pour avoir ces informations sous la forme d'un flots de liens, un script d'extraction a été développé [URL](#). Lors de l'extraction, 2269 courriels n'ont pas pu être pris en compte car certaines informations étaient manquantes ou mal formées.

Une fois les informations récupérées, il faut les transformer en un flot de liens cohérent. Pour chaque message m , nous extrayons son auteur $a(m)$, l'instant $t(m)$ auquel le message a été posté², le message auquel il répond $p(m)$ trouvé via le champ IN-REPLY-TO, son destinataire $a(p(m))$ et la discussion $D(m)$ dans laquelle il apparaît. Comme les messages *racines* ne répondent à aucun autre, nous imposons $p(m) = m$. L'ensemble de liens du flot est donc $\{(t(m), a(m), a(p(m)))\}_m$. Nous ne prenons pas en compte la direction des liens.

Une fois le flot créé, il est encore nécessaire de vérifier sa cohérence. Un message peut être filtré pour différentes raisons : le courriel apparaît avant le message auquel il est censé répondre, le message auquel il répond n'est pas présent dans l'archive, l'auteur et le destinataire sont la même personne. Cette dernière condition permet notamment d'éviter la présence de boucles dans le flot. Cela concerne principalement les *racines*. Il s'agit de vérifications simple auquel il faut ajouter les vérifications sur la cohérence de la structure des discussions. Ainsi, une discussion est entièrement retirée du jeu de données s'il manque la *racine*, si un de ses messages a été retiré à l'étape précédente. Après ces vérifications, environ 7% des discussions sont retirées. À cela, il faut également tenir compte de notre temps d'observation qui est partiel. En effet, une discussion dont le dernier message a lieu 1 semaine avant la fin de la capture peut ne pas être terminée. De même, une discussion durant très longtemps n'ont que peu de probabilité d'être capturée en entier. Pour corriger ces biais, nous filtrons également les discussions ayant débutées trop récemment ou durant trop longtemps. La limite pour considérer une discussion trop récente ou trop longue a été fixé à 4 ans (1.26×10^8 s) car nous avons constaté qu'uniquement quelques discussions dépassé ce seuil sur la distribution de durées des discussions dans la figure 4.1.

Une fois tout ces messages filtrés, nous obtenons un flot de liens avec 316 569 liens entre 34 648 personnes pendant presque 19 ans et 116 999 discussions. Mis à part les 237 664 messages de début de discussion, ce sont 168 482 courriels qui ont été filtrés soit environ 23%.

4.2 Caractéristiques élémentaires des discussions

Les caractéristiques les plus élémentaires des discussions sont le nombre de courriels, le nombres de personnes, le nombre de paires de personnes distinctes en interaction directes et leur durée. Dans la figure 4.1, sont présentées les distributions cumulatives inverses de ces quantités et on remarque qu'elles sont toutes hétérogènes. On remarque que les données filtrées ne diffèrent pas qualitativement des données brutes.

2. Cet instant est convertit en *timestamp* en tenant compte des fuseaux horaires.

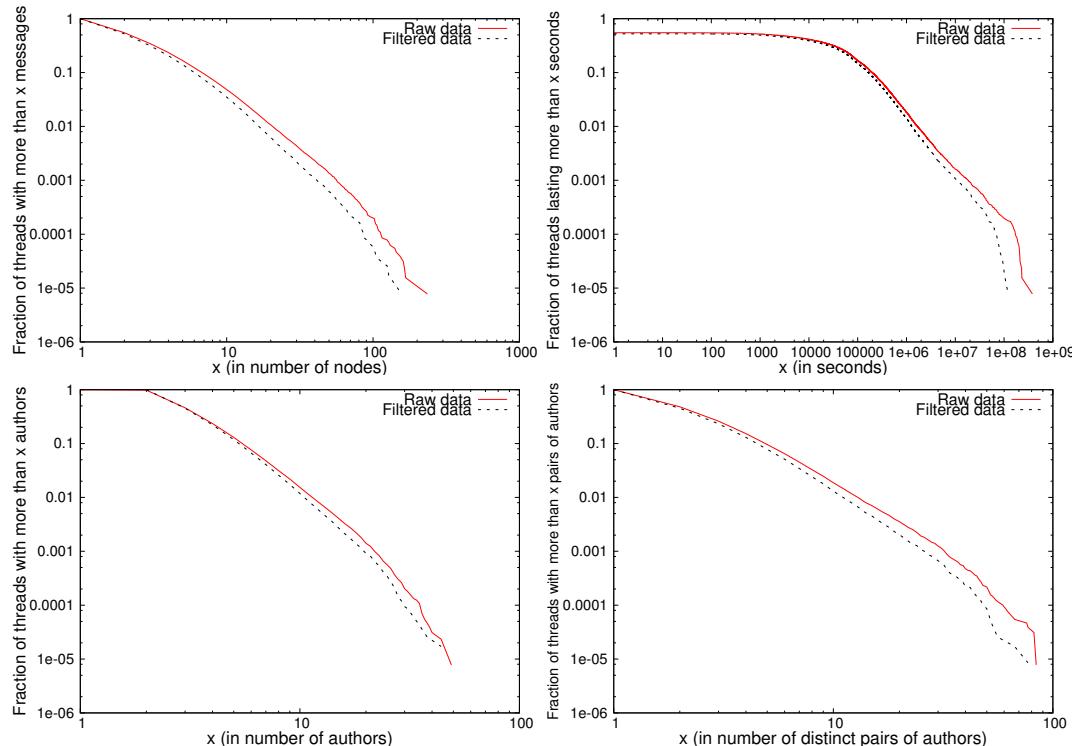


FIGURE 4.1 – Distribution cumulative inverse de différentes caractéristiques pour les données brutes (ligne pleine) et filtrées (ligne pointillée). En haut à gauche : nombre de courriels dans une discussion ; en haut à droite : durée d'une discussion ; en bas à gauche : nombre de personnes dans une discussion ; en bas à droite : nombre de paires d'auteurs distinct dans une .

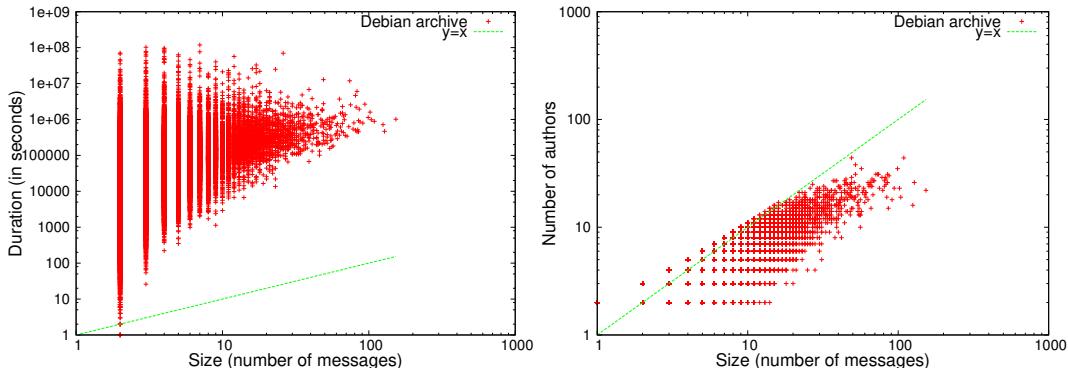


FIGURE 4.2 – Gauche : Corrélations entre le nombre de courriels et la durée d'une discussion. Droite : Corrélation entre le nombre de courriels et le nombre d'auteurs dans une discussion.

La distribution des durées des discussions montre que la majorité des discussions dure environ une journée ou moins (10^5 secondes équivaut à moins de 28 heures). Par ailleurs, on remarque qu'il n'existe que quelques discussions durant plus d'un an.

Ces premières observations sont nécessaires mais pas suffisantes pour comprendre les caractéristiques d'une discussion. Nous avons également étudié la corrélation entre ces différentes notions et une partie d'entre elles sont présentées dans la figure 4.2.

La corrélation entre la durée et le nombre de courriels, dans la figure 4.2 partie gauche, met en évidence que plus une discussion est grande en nombre de courriels plus elle dure longtemps, ce qui est attendu. Par contre, on observe que les petites discussions ont des durées très variables. Dans la partie droite de la figure 4.2 présentant la corrélation entre le nombre de courriels et d'auteurs, on observe un autre fait attendu [6] qui est qu'une discussion est constituée, en général, de plus de messages que de participants. Ainsi lors d'une discussion, c'est un petit nombre de personnes qui échangent potentiellement beaucoup de messages.

Enfin, il est intéressant d'observer la dynamique des échanges entre deux personnes. Soit $\tau(u, v) = (t_{i+1} - t_i)_{i=0..k+1}$ la séquence des temps inter-contacts des k liens entre les nœuds u et v , où t_0 est le temps entre α et le premier lien et t_{k+1} est le temps entre le dernier lien et Ω . Il s'agit du temps écoulé avant que deux personnes se contactent à nouveau, indépendamment peu importe la conversation. Dans la figure 4.3 est représentée la distribution cumulative inverse du temps inter-contacts. 21% des temps inter-contacts sont inférieurs à 30 jours (2.6×10^6 s). Ce chiffre bien que relativement faible est tout de même important car il s'agit de discussions ouvertes où tout le monde peut participer. En particulier, une personne peut envoyer une demande d'aide à un moment donné et ne plus jamais échanger avec les mêmes personnes. Or, on observe que 21% des contacts sont renouvelés en moins de 30 jours. La participation est donc relativement élevée.

4.3 Étude des discussions en tant que sous-flots

4.3.1 Application de la Δ -densité

Jusqu'à maintenant aucune notion intrinsèquement liée aux flots de liens n'a été utilisée pour caractériser les discussions. Le but est d'évaluer si cette structure de flot peut se rapprocher d'une structure communautaire. Comme dit précédemment, les

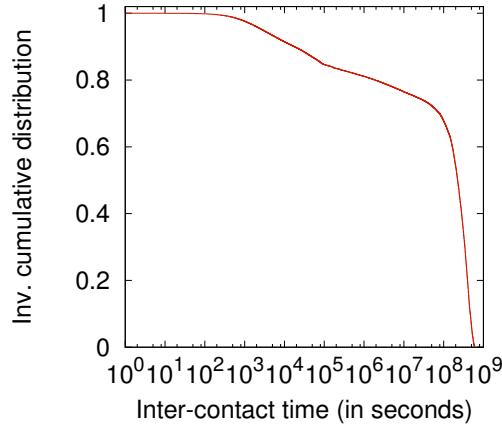


FIGURE 4.3 – Distribution des temps inter-contacts dans le fil de discussions.

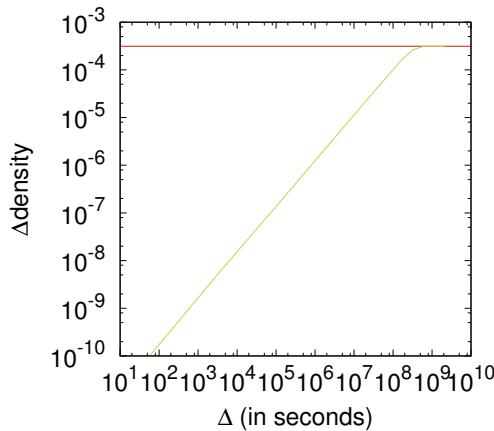


FIGURE 4.4 – Évolution de la Δ -densité (en vert) du flot de liens pour Δ de 60 seconde à 20 ans. En rouge, la densité dans le graphe agrégé.

communautés sont souvent définies comme étant des structures devant être densément connectées. C'est pourquoi nous nous attachons à étudier la densité des discussions.

Comme ces données se modélisent par un flot de liens où les liens n'ont pas de durée, nous étudions la Δ -densité pour différentes valeurs de Δ entre 1 seconde et 20 ans. Tout d'abord dans la figure 4.4 est représentée la Δ -densité globale du le flot. En couvrant un spectre aussi large de Δ , on observe que la Δ -densité est croissante avec Δ mais surtout on observe bien la convergence de Δ -densité vers 3.139×10^{-4} , la densité du graphe agrégé, lorsque Δ est proche de $\omega - \alpha$.

Cependant, la Δ -densité du flot n'apporte que peu d'informations en elle-même. Elle est surtout utile pour comparer les valeurs de Δ -densité des sous-flots que sont les discussions. Ainsi dans la figure 4.5, est présentée la distribution cumulative inverse de la Δ -densité des discussions pour différentes valeurs de Δ . On remarque que les différentes valeurs de Δ ne semblent pas influencer qualitativement la distribution de Δ -densité. Cette courbe met surtout en évidence que les discussions sont des structures beaucoup plus denses que le flot. En effet, la densité médiane des discussions est, selon la valeur de Δ , entre 2.69×10^{-4} et 0.28 alors que le flot a une Δ -densité variant entre 1.05×10^{-10} et 3.42×10^{-5} . La Δ -densité des discussions est donc en moyenne 10^5 fois plus élevée que celle du flot. Bien que notable, ce fait est attendu notamment car le flot dure beaucoup plus longtemps et concerne beaucoup

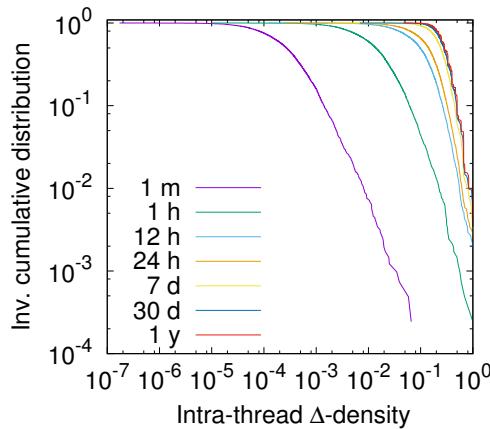


FIGURE 4.5 – Distribution cumulative inverse de la Δ -densité des discussions pour différentes valeurs de Δs .

plus de nœuds que les discussions.

Afin d'aller plus loin dans l'étude de cette structure, il faut revenir à une définition plus précise de ce qu'est une bonne communauté. En soit, une valeur de densité n'est pas suffisante pour définir une structure communautaire. En effet, une discussion ayant une densité de 0.8 peut ne pas être une communauté tandis qu'une autre ayant une densité proche de zéro peut être une communauté. Il faut définir un point de comparaison pour effectivement affirmer qu'une structure est particulièrement dense. La prise en compte de la densité globale est un début mais n'est pas suffisante.

Une autre définition d'une communauté est qu'elle devrait être plus densément connectée à l'intérieur qu'avec les autres communautés adjacentes. Pour un graphe $G = (V, E)$ et une communauté C_i de la partition $C = \{C_j\}_{j=1..k}$ de V en k communautés, cela se traduit par le calcul de la densité entre les communautés, $\delta^{inter}(C_i)$:

$$\delta^{inter}(C_i) = \frac{1}{|C|-1} \sum_{j,i \neq j} \frac{|\{(u, v) \in E \text{ t.q. } u \in C_i \text{ and } v \in C_j\}|}{|C_i| \cdot |C_j|}. \quad (4.1)$$

Il s'agit tout simplement de la probabilité qu'un lien existe entre les nœuds des deux communautés. Encore une fois, cette notion n'a pas de sens direct dans le formalisme de flot de liens et il est nécessaire de l'adapter. Pour ce faire, nous définissons la Δ -densité inter discussions entre deux discussions D_i et D_j : $\delta_{\Delta}^{inter}(D_i, D_j)$. Soit $L_{\Delta} = \xi(L, \Delta)$ et $L_{inter}(D_i, D_j) = (T', V', E')$ avec $V' = V(D_i \cup D_j)$, $T' = [t_{\beta}(D_i \cup D_j), t_{\psi}(D_i \cup D_j)]$, et $E' = E_{\Delta} \setminus (D_i \cup D_j)$. Avec ces définitions, la définition de $\delta_{\Delta}^{inter}(D_i, D_j)$ est la suivante : $\delta_{\Delta}^{inter}(D_i, D_j) = \delta(L_{inter}(D_i, D_j))$. Il s'agit donc de la densité du flot inter discussions. Le flot inter discussion est constitué des liens entre les nœuds induits par D_i et D_j qui n'appartiennent ni à D_i ni à D_j . Dans la figure 4.6, un exemple de flot inter discussion est représenté.

Afin d'obtenir la Δ -densité inter discussions entre D_i et tout les autres discussions, nous utilisons la moyenne des densité inter discussion entre D_i et les autres discussions, soit :

$$\delta_{\Delta}^{inter}(D_i) = \frac{1}{|C|-1} \sum_{j,i \neq j} \delta_{\Delta}^{inter}(D_i, D_j). \quad (4.2)$$

La distribution cumulative inverse de la Δ -densité inter discussions est présentée

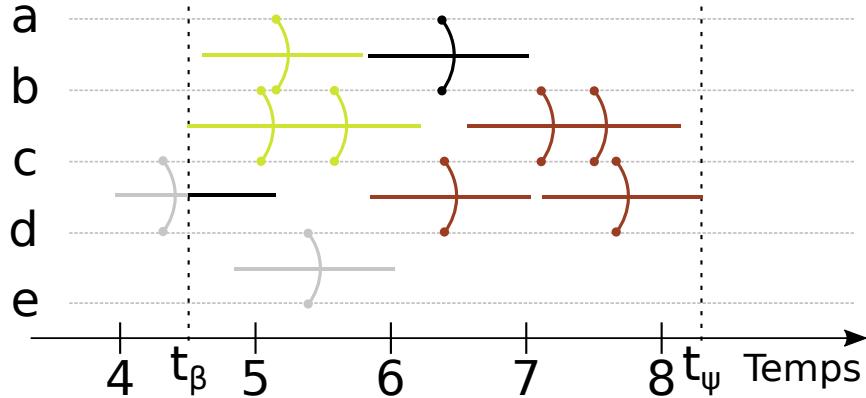


FIGURE 4.6 – Le flot entre les discussions vertes et rouge est constitué des liens ou partie de liens en noir. Les liens en gris ne sont pas pris en compte.

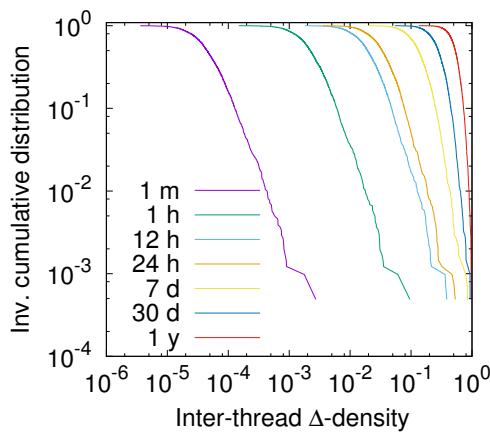


FIGURE 4.7 – Distribution cumulative inverse de la Δ -densité inter discussions pour différentes valeurs de Δ_s .

dans la figure 4.7 pour différentes valeurs de Δ . Bien que similaire, le comportement de la Δ -densité inter discussions diffère qualitativement de celui de la Δ -densité. La Δ -densité inter discussions croît également en fonction de Δ mais il y a toujours une différence notable entre $\Delta = 1 \text{ mois}$ et $\Delta = 1 \text{ an}$ ce qui n'est pas le cas pour la Δ -densité. Cette différence est normale car lors du calcul de Δ -densité le nombre de liens considérés est fixe peut importe Δ alors qu'il croît avec Δ lors du calcul de Δ -densité inter discussions. Cet effet est visible dans la figure 4.6. Le lien (c, d) qui apparaît peut avant t_β n'est pas pris en compte si Δ est proche de 0 alors qu'il est en parti pris en compte lorsqu'un Δ plus grand est considéré, ce qui est le cas dans la figure.

Un autre facteur est aussi la durée considérée, $t'_\psi - t'_\beta$, qui est plus longue que la durée des discussions.

Afin de comparer plus aisément Δ -densité et Δ -densité inter discussions, la corrélation entre ces deux mesures est présentée dans la figure 4.8 pour différentes valeurs de Δ . On remarque que les discussions sont effectivement plus denses intérieurement qu'avec les autres discussions. La différence est de plusieurs ordres de grandeur lorsque Δ est petit et elle diminue lorsque Δ croît. Pour $\Delta = 20 \text{ ans}$ dans la figure 4.8d, la différence n'est plus visible car à cette échelle de temps, l'ancrage temporel des discussions n'est plus décisif. On remarque tout de même que pour

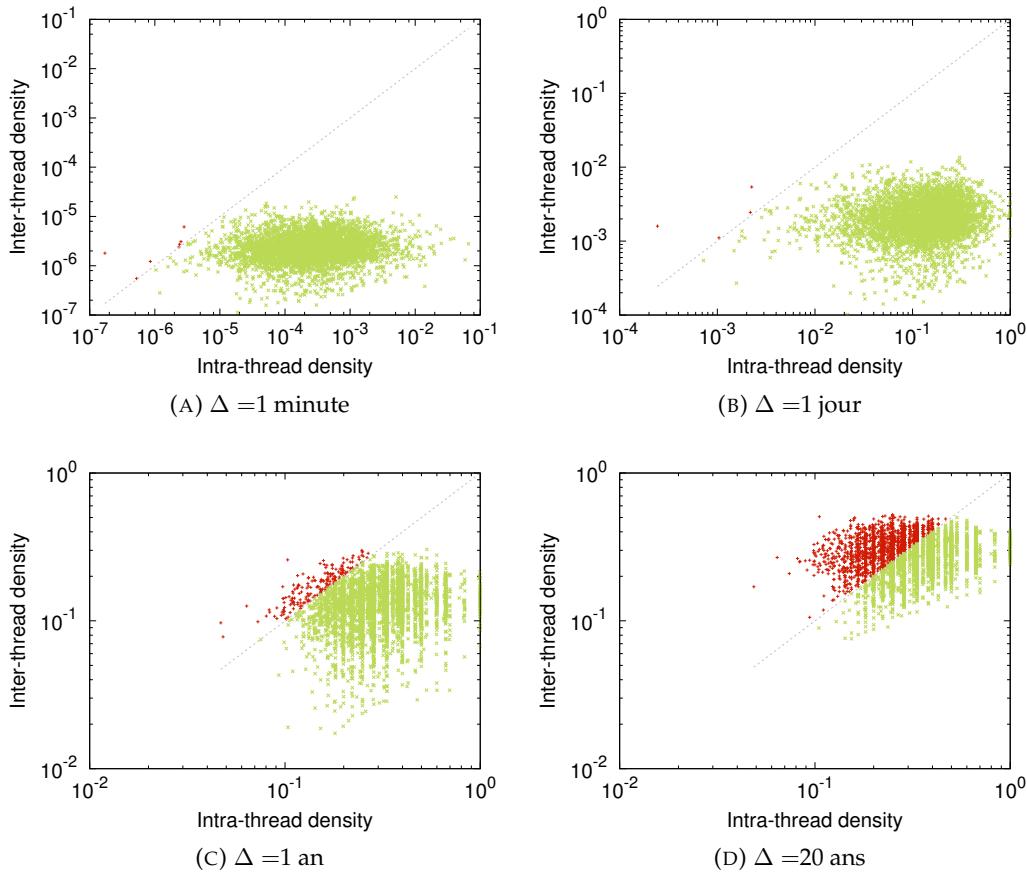


FIGURE 4.8 – Corrélations entre Δ -densité et Δ -densité inter discussions pour différentes valeurs de Δ . Une discussion est en vert (resp. rouge) si elle a une Δ -densité plus (resp. moins) élevée que sa Δ -densité inter discussions.

$\Delta = 1 \text{ an}$, la différence reste notable.

4.3.2 Répartition temporelle et structurelle des discussions

Nous avons étudié la densité des discussions et entre les discussions mais il est également intéressant d'observer comment ces discussions sont réparties topologiquement et temporellement. Pour étudier la répartition des discussions dans le temps, nous construisons un graphe d'intervalle $\text{REF}_X = (V_X, E_X)$ représentant le chevauchement temporel. Chaque discussion du flot devient un nœud de V_X et le lien (i, j) existe dans E_X si les discussions D_i et D_j correspondantes ont eu lieu au même instant, *i.e.* $[\alpha_i, \omega_i] \cap [\alpha_j, \omega_j] \neq \emptyset$. De manière similaire, nous définissons le graphe de chevauchement topologique $Y = (V_Y, E_Y)$. Les nœuds de ce graphe représentent encore une fois les discussions du flot et un lien existe entre deux discussions si au moins une personne a participé aux deux, *i.e.* $V(D_i) \cap V(D_j) \neq \emptyset$.

Ces deux graphes sont constitués de 116 999 nœuds et d'environ 2 millions de liens pour le graphe de chevauchement temporel et d'environ 63 millions de liens pour le graphe de chevauchement topologique. Par construction, ces graphes contiennent beaucoup d'informations sur les relations entre les discussions.

Dans la figure 4.9(gauche), est représentée la corrélation entre le degré d'une discussion dans le graphe de chevauchement temporel X et sa durée. Il y a une

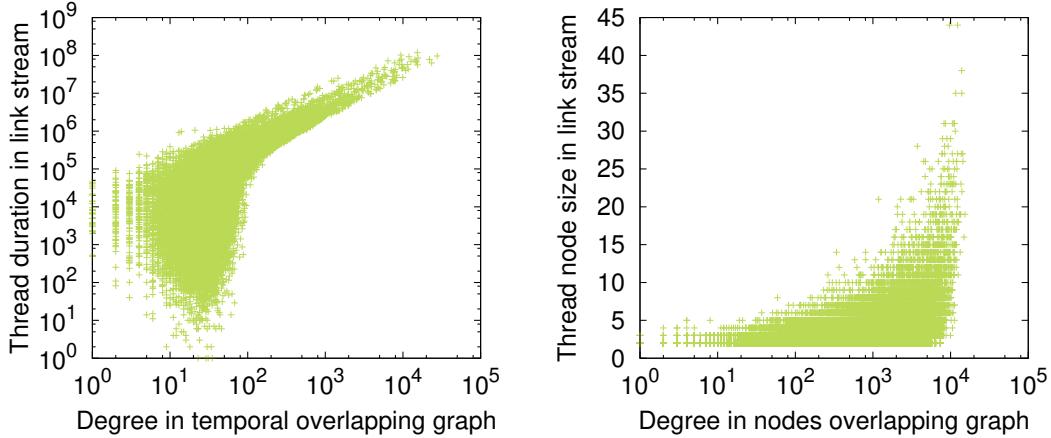


FIGURE 4.9 – Gauche : Corrélation entre le degré des discussions dans le graphe de chevauchement temporel et leur durée. Droite : Corrélation entre le degré des discussions dans le graphe de chevauchement topologique et leur nombre de participants.

corrélation évidente entre ces deux notions lorsque les discussions ont une durée supérieur à 10^5 secondes. Plus une discussion dure longtemps, plus elle a de chance d'avoir lieu en même temps que beaucoup d'autres discussions. On observe également que, même pour les discussions durant moins d'un jour (8.6×10^4 s), il peut y avoir jusqu'à une centaine d'autres discussions actives sur la même période.

La figure 4.9(droite) présente la corrélation entre le degré d'une discussion dans le graphe de chevauchement topologique Y et son nombre de participants. La corrélation est moins nette mais il y a tout de même une tendance. Par contre, on remarque de manière frappante que même une petite discussions peut partager des noeuds avec les énormément d'autres discussions.

4.3.3 Flot quotient

Le graphe quotient^{REF} est une autre notion clef pour étudier les relations entre les communautés d'un graphe $G = (V, E)$. Soit une partition $C = \{C_i\}_{1..k}$ des noeuds de G en k communautés, chaque communauté est représentée dans le graphe quotient \bar{G} par un noeuds dans V . Il y a un lien entre deux communauté C_i et C_j dans E si il existe au moins un lien entre un noeuds de C_i et un noeuds de C_j . Voir une illustration sur la figure 4.10. Il est possible d'ajouter un poids sur les liens de \bar{G} égale au nombre de liens reliant les communautés. Le graphe quotient permet de facilement étudier, dans un graphe, les relations entre les communautés.

Nous étendons ici cette notion de graphe quotient aux flots de liens. Nous définissons le flot quotient, $Q = (T_Q, V_Q, E_Q)$, induit par une partition $P = \{P_i\}_{1..k}$ en k sous-flots de la manière suivante. Chaque sous-flot P_i est représenté par un noeud dans V_Q . Il existe un lien (t, P_i, P_j) dans E_Q si il existe $(t_1, u, v) \in P_i$, $(t_2, u, v') \in P_j$ et $(t_3, u, v'') \in P_i$ avec $t_1 \leq t_2 \leq t_3$. En d'autre termes, il y a un lien dans E_Q si un noeud u a un lien dans P_j qui apparaît entre deux autres de ses liens du groupe P_i .

Le flot quotient induit par les discussions dans le jeu de données contient 12 281 269 liens impliquant 68 524 discussions différentes. Comme le jeu de données contient 116 999 discussions, il y a donc 48 475 discussions sans lien et qui ne seront pas prises en compte par la suite. Ce nombre de discussions non-reliées est élevé comparé à ce qui est obtenu dans un graphe. En effet dans un graphe, un noeud de degré 0 correspond à une communauté qui est une composante connexe

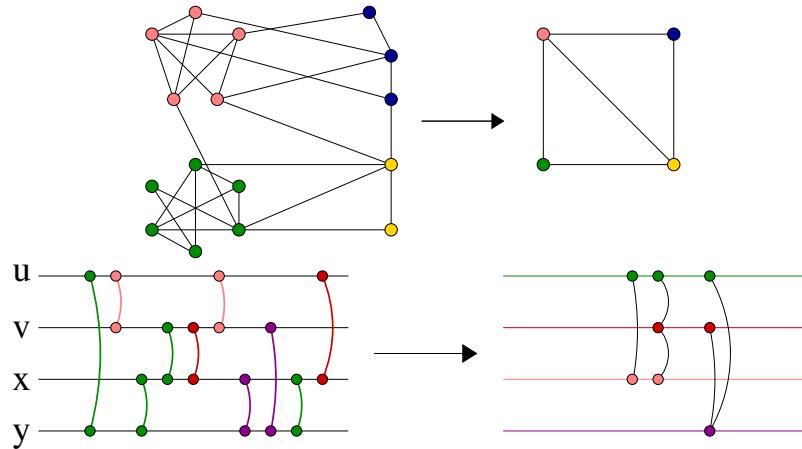


FIGURE 4.10 – Haut : Exemple de graphe ayant une structure communautaire et son graphe quotient associé. Bas : Exemple d'un flot de lien avec une structure ainsi que son flot quotient associé.

(ou un union de composantes connexes). En ajoutant l'information temporelle, les discussions sont séparées par le temps dans flot. C'est pourquoi un grand nombre de discussions n'ont pas de liens dans le flot quotient. Ce phénomène est d'autant plus vrai pour les petites discussions.

Il faut aussi noter qu'il y a environ 20 fois plus de liens dans le flot quotient que dans le flot initial. Cela est normal car un lien dans le flot peut donner lieu à plusieurs liens dans le flot quotient. Ce cas est visible dans la figure 4.10. Le lien (x, y) du groupe violet du flot à gauche donne lieu au lien (*violet*, *rouge*) et au lien (*violet*, *vert*) dans le flot quotient à droite.

La figure 4.11 présente la Δ -densité du flot de liens initial et du flot quotient pour différentes valeurs de Δ . Le flot initial et le flot quotient ont le même comportant de densité mais le flot quotient est moins Δ -dense que le flot initial. Ce résultat diffère par rapport à ce qui est obtenu dans un graphe. Cela est dû au nombre de nœuds qui augmente dans le flot quotient. Mais le flot quotient contient tout de même beaucoup de liens. En effet, le degré moyen dans le flot quotient est moyenne 25 fois plus élevé que dans le flot.

4.3.4 Conclusion

Nous avons utilisé le modèle de flot de liens pour étudier une archive de courriels provenant du projet Debian. Grâce au modèle de flot de liens, nous avons étudié des notions clefs pour mieux comprendre la répartition temporelle et topologique des discussions. Nous avons étudié la notion de Δ -densité sur les discussions en elles mêmes. Puis, nous avons étudié les relations entre les discussions avec la Δ -densité inter discussions, les projections en graphe de chevauchement temporel ou topologique et le flot quotient.

Cette étude repose en grande partie sur la notion de Δ -densité qui nécessite un paramètre fixé arbitrairement. Nous avons à chaque fois testé un ensemble de valeurs de Δ variant d'une seconde jusque parfois 20 ans et, lors de ces tests, aucune valeur Δ caractéristique n'a pu être identifiée. Il semble donc que la Δ -densité soit relativement robuste vis-à-vis de Δ dans ce contexte.

Nous avons tout d'abord observé que les discussions forment une structure plus dense que le flot de liens. De manière encore plus forte, nous avons constaté, grâce à la Δ -densité inter discussion, que les discussions sont plus denses en interne qu'en

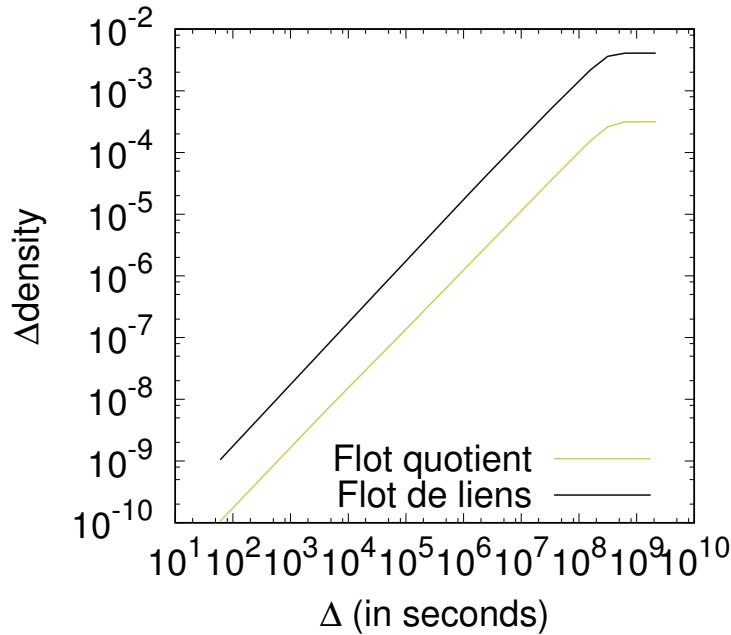


FIGURE 4.11 – Δ -densité du flot de liens et du flot de liens quotient en fonction de Δ pour $\Delta = 1mn, 1h, 12h, 1j, 7j, 30j, 1\text{ an}$ et 20 ans .

externe. C'est une caractéristique importante des communautés que l'on trouve dans les graphes mais qui n'avait pas été observée dans un contexte temporel. À partir de ces observations, nous avons également observé les relations entre les discussions. Via le graphe de chevauchement temporel, nous avons validé le fait que différentes discussions ont lieu en même temps et que par conséquent une agrégation temporelle entraînerait une perte d'information. De même via le graphe de chevauchement topologique, on remarque que la structure est très recouvrante sur les noeuds, rendant ainsi l'utilisation de partitions statiques de noeuds difficilement envisageable pour décrire les discussions.

On a pas étudié de mesure spéciale graphe sur X ou Y.

Faire évaluation des threads comme pertinent.

4.4 Détection de structures denses

À partir du constat que les discussions forment une structure particulière, il est naturel d'essayer de les retrouver automatiquement. Pour y parvenir, il faut un moyen capable de trouver des sous-flots-denses dans le flots. C'est à dire une méthode capturant des groupes de liens qui soient proche temporellement et topologiquement. Il serait tentant de directement d'optimiser la densité dans le flot mais ce n'est pas envisageable car un groupe constitué d'un unique lien a une densité de 1. Il faut donc trouver une autre méthode. C'est pourquoi nous avons construit une autre projection du flot en un graphe statique afin d'y appliquer une méthode de détection de communautés. Le problème est alors de réussir à créer une transformation de telle sorte que les informations temporelles et topologiques ne soient pas complètement détruites.

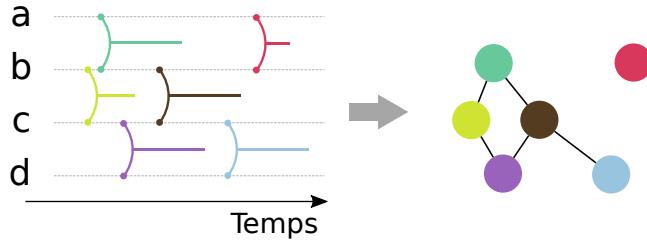


FIGURE 4.12 – Transformation d'un flot de liens avec 4 noeuds (a-d) et 6 liens à gauche en un graphe à droite à 6 noeuds. La couleur d'un noeuds dans le graphe indique le lien du flot qu'il représente.

4.4.1 Méthode de détection

Dans la transformation que nous appliquons, nous créons un graphe non-orienté et non-pondéré $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Chaque lien du flot est représenté par un noeud. Deux liens (b, e, u, v) et (b', e', u', v') sont connectés dans le graphe s'ils partagent un noeud, i.e. $\{u, v\} \cap \{u', v'\} \neq \emptyset$, et si les intervalles s'intersectent, i.e. $[b, e] \cap [b', e'] \neq \emptyset$, voir figure 4.12. Ainsi, un lien dans le graphe représente une connexion structurelle et temporelle entre deux liens du flot de liens. Les groupes denses dans le graphe représentent donc des groupes de liens connectés temporellement et topologiquement dans le flot.

Cette définition n'est valide que pour un flot de liens avec durée. Or, les courriels échangés n'ont pas de durées. Lors du calcul de la densité dans la section 4.3.1, nous avions ajouté une durée arbitraire Δ . Ici, il n'est pas très pertinent d'appliquer la même logique. En effet, si on utilise un Δ faible, alors il n'y aura que très peu de liens dans \mathcal{E} et les noeuds représentant les liens d'une discussions ne seront pas forcément connexes. Il paraît illusoire d'espérer retrouver les discussions dans \mathcal{G} si elles ne sont même pas connexes. C'est pourquoi nous adoptons une autre manière d'ajouter une durée sur les liens.

Pour chaque message m , nous connaissons $p(m)$, le message auquel il répond dans la discussion. Nous définissons un autre flot, \mathfrak{L} , où les liens représentant les messages sont de la forme $(t(p(m)), t(m), a(m), a(p(m)))$. Ainsi, deux messages, m_1 et m_2 , se succédant dans une discussion sont par définition reliés topologiquement car $a(m_1) = a(p(m_2))$. Ces deux messages sont aussi reliés temporellement car nous avons la relation suivante :

$$\begin{aligned}[t(p(m_1)), t(m_1)] \cap [t(p(m_2)), t(m_2)] &= \\ [t(p(m_1)), t(m_1)] \cap [t(m_1), t(m_2)] &= [t(m_1)] \neq \emptyset.\end{aligned}$$

Par construction, une discussion est donc représentée dans \mathcal{G} par un ensemble connexe de noeuds. Un fois \mathcal{G} construit, on peut appliquer un algorithme de détection de communautés.

Avec cette construction, \mathcal{G} contient plus d'1 millions de liens pour les 116 999 discussions présentes. Sur ce graphe, nous avons appliqué l'algorithme de Louvain [3] qui optimise la modularité. D'autres algorithmes peuvent également être appliqués s'ils capturent des groupes de noeuds disjoints et qu'ils passent à l'échelle. Les groupes trouvés par Louvain sont des communautés dans \mathcal{G} . Par conséquent, ils sont censé être densément connectés dans \mathcal{G} . Comme un lien de \mathcal{G} correspond à une connexion temporelle et topologique dans le flot, on peut espérer qu'ils correspondent à des groupes denses dans le flot.

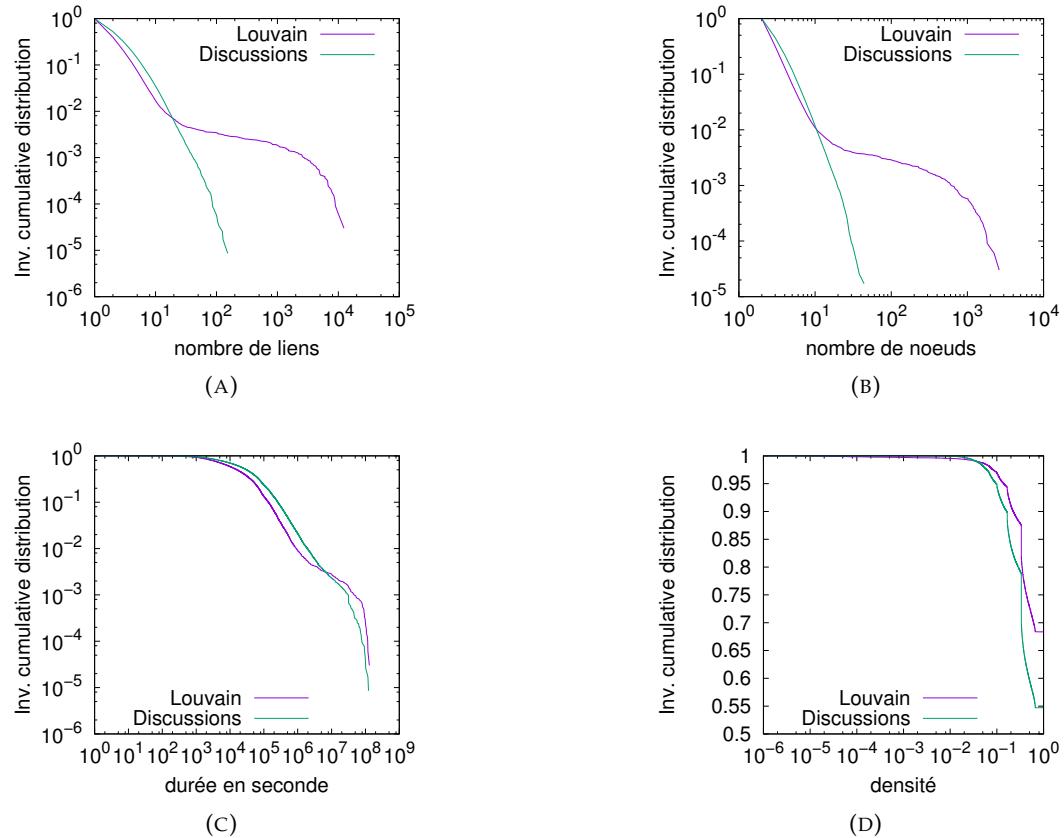


FIGURE 4.13 – Distribution cumulative inverses du nombre de liens (a), du nombre de nœuds (b), de la durée (c) et de la densité (d) pour les groupes trouvés par Louvain et les discussions.

4.4.2 Comparaison des partitions

Avant de comparer la structure des discussions, D , et la partition, \mathfrak{D} , trouvée par la méthode de Louvain sur \mathcal{G} , il est nécessaire de décrire cette dernière. Dans la figure 4.13, les distributions cumulatives inverses du nombre de liens, du nombre de nœuds et de leur durée sont présentées pour les groupes de \mathfrak{D} . Pour rappel, les mêmes données sont représentées pour les discussions. On remarque tout de suite que \mathfrak{D} contient des groupes beaucoup plus gros en nombre de nœuds et de liens alors qu'ils ont les mêmes durées.

Ces deux structures sont donc très différentes mais cela pourrait être dû à l'algorithme de Louvain qui n'est pas adapté pour trouver des groupes denses. C'est pourquoi, nous avons également observé la densité des groupes de D et \mathfrak{D} dans le flot \mathfrak{L} . Le résultat est visible dans la figure 4.13d. Comme les liens de \mathfrak{L} ont une durée, il est possible d'utiliser directement la densité au lieu de la Δ -densité utilisée précédemment. On remarque que les groupes de \mathfrak{D} , bien que plus gros, sont plus denses que les groupes de D . Cependant la distribution cumulative inverse cache les effets de la taille sur la densité. Or à taille égale (entre 2 et 160) liens, on remarque que les groupes trouvés par Louvain sont légèrement plus denses en médiane, 0.34 contre 0.33, et également plus denses en moyenne, 0.46 contre 0.38.

En revanche, les plus gros groupes ($|D_j| > 160$) trouvés par Louvain ont une densité plus faible mais c'est attendu à cause de leur taille.

Si les groupes de \mathfrak{D} sont plus denses, c'est peut-être car ils regroupent plusieurs

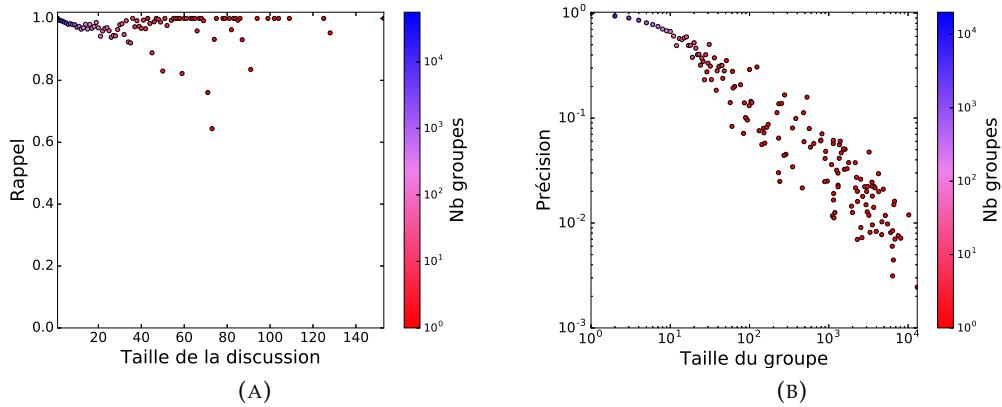


FIGURE 4.14 – Précision (B) et rappel (A) entre les discussions et les groupes trouvés par Louvain en fonction de la taille. Chaque point représente la moyenne pour les groupes ayant la même taille. La couleur du point indique le nombre de groupes ayant la même taille.

discussion de D dans un groupe. Pour comparer deux partitions, l'indice de Jaccard est classiquement utilisé pour calculer la *précision* et le *rappel* qui sont définis de la manière suivante :

$$\text{précision}(D_i) = \max_j \frac{|\mathfrak{D}_j \cap D_i|}{|\mathfrak{D}_j|}, \quad \text{rappel}(D_i) = \max_j \frac{|\mathfrak{D}_j \cap D_i|}{|D_i|}.$$

Dans la figure 4.14, est présenté la *précision* et le *rappel* des discussions en fonctions de leurs tailles. Chaque point représente la moyenne du *rappel* et de *précision* pour les groupes d'une taille donné. On voit qu'il y a un important rappel et ce même pour les grandes discussions, ce qui veut dire qu'en générale une discussion D_i est totalement incluse dans un groupe \mathfrak{D}_j . En revanche, la précision est très faible car un groupe \mathfrak{D}_j contient plusieurs discussions, ce qui est cohérent avec la taille très importante des groupes de \mathfrak{D}_j .

Il semble donc que la partition \mathfrak{D} soit proche de D mais que ces groupes soient plus gros. Pour circonvenir à ce problème, nous appliquons de manière récursive l'algorithme de Louvain sur chaque graphe induit par un groupe \mathfrak{D}_j . Ce processus permet de subdiviser de manière récursive chaque groupe \mathfrak{D}_j . Par construction, un groupe trouvé au niveau h , $\mathfrak{D}_j(h)$, est donc inclus dans un groupe trouvé au niveau $h - 1$, c'est à dire $\mathfrak{D}_j(h) \subset \mathfrak{D}_j(h - 1)$. Le niveau 0 est la première partition trouvée par l'algorithme de Louvain sur le \mathfrak{G} .

Soit $D_i \in D$ et $\mathfrak{D}_{\tilde{j}}(h)$ avec $h \in \mathbb{N}$, le groupe trouvé par la méthode de Louvain au niveau h qui soit le plus proche de D_i au niveau h , c'est-à-dire $|\mathfrak{D}_{\tilde{j}}(h) \cap D_i| = \max_j |\mathfrak{D}_j(h) \cap D_i|$. Avec ces définitions, on observe la relation suivante : $\mathfrak{D}_{\tilde{j}}(h) \cap D_i \subseteq \mathfrak{D}_{\tilde{j}}(h - 1) \cap D_i$. La définition de *rappel* de l'équation 4.4.2 n'est donc pas adaptée pour les niveaux inférieurs et nous l'adaptons de la manière suivante :

$$\text{rappel}(D_i, h) = \max_j \frac{|\mathfrak{D}_j(h) \cap D_i|}{|D_i \cap \mathfrak{D}_{\tilde{j}}(h - 1)|}. \quad (4.3)$$

Ainsi, le *rappel* au niveau h prends en compte le maximum d'élément qu'il est possible de trouver à ce niveau. La définition de *précision* ne pose quant à elle pas

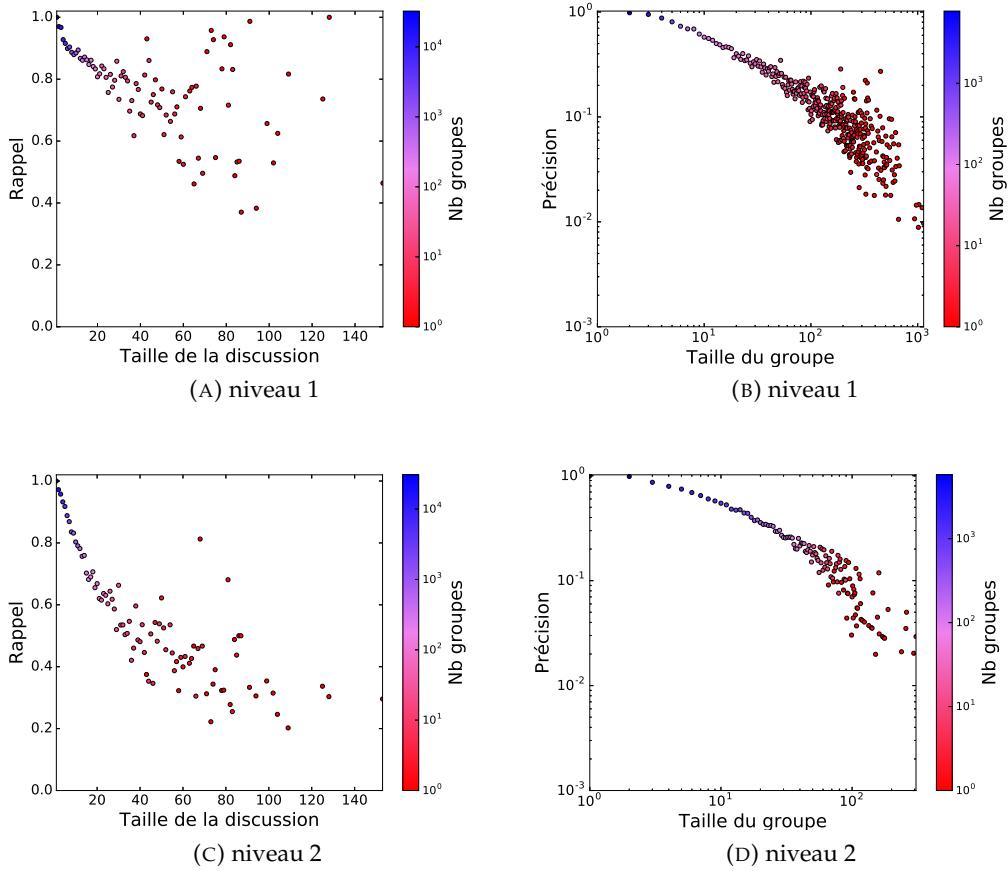


FIGURE 4.15 – Précision (B,D,F) et rappel (A,B,C) entre les discussions et les groupes trouvés par Louvain à différent niveaux récursifs. Chaque point représente la moyenne pour les groupes ayant la même taille. La couleur du point indique le nombre de groupes ayant la même taille.

de problème. Dans la figure 4.15, sont représentés le *rappel* adapté et la *précision* pour le premier et deuxième niveau de récursion de la même manière que pour la figure 4.14. On remarque que, dès le premier niveau, le rappel baisse et que ce phénomène s'amplifie fortement au niveau suivant. Cela implique que les discussions ne sont plus incluses dans un groupe mais au contraire réparties dans plusieurs. La précision quant à elle augmente légèrement mais cela est dû à la baisse de la taille des groupes trouvés. Il semble donc qu'il ne soit pas possible avec cette approche de retrouver automatiquement les discussions.

4.4.3 Conclusion

Nous avons avec cette méthode mis en évidence des groupes denses. Les groupes trouvés sont plus gros et plus denses que la structure des discussions. Cependant, ces observations ne remettent pas en cause les conclusions faites dans la section 4.3 pour plusieurs raisons. Tout d'abord, les flots de lien étudiés ne sont pas exactement les-mêmes ($L \neq \mathcal{L}$). Ce changement de flot est nécessaire pour le fonctionnement de la méthode de détection. Ensuite, les deux structures ne sont pas complètement différentes car les groupes trouvés semblent en fait agréger plusieurs discussions.

Malheureusement, nous n'avons pas réussi avec notre méthode à isoler chaque discussion malgré notre approche récursive. Pourtant, il semble que la structure trouvée ai du sens au vu des valeurs de densité des groupes.

Chapitre 5

Détection de groupes denses (SNAM)

Sommaire

5.1	Calcul des groupes candidats	37
5.2	Calcul évaluation	37
5.3	Jeux de données	37
5.4	Application	37
5.5	Conclusion	37

5.1 Calcul des groupes candidats

5.2 Calcul évaluation

5.3 Jeux de données

5.4 Application

5.5 Conclusion

Chapitre 6

Fonction de qualité

Sommaire

6.1	Définition	39
6.2	Générateur de flots de liens avec structure communautaire	39

6.1 Définition

**6.2 Générateur de flots de liens avec structure communau-
taire**

Chapitre 7

Conclusion

Bibliographie

- [1] Yong-Yeol Ahn, James P. Bagrow, and Sune Lehmann. Link communities reveal multiscale complexity in networks. *Nature*, 466(7307) :761–764, aug 2010.
- [2] Edward A Bender and E.Rodney Canfield. The asymptotic number of labeled graphs with given degree sequences. *Journal of Combinatorial Theory, Series A*, 24(3) :296–307, may 1978.
- [3] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics : Theory and Experiment*, 2008(10) :P10008, oct 2008.
- [4] Leon Danon, Albert Díaz-Guilera, Jordi Duch, and Alex Arenas. Comparing community structure identification. *Journal of Statistical Mechanics : Theory and Experiment*, 2005(09) :P09008–P09008, sep 2005.
- [5] J-C Delvenne, S N Yaliraki, and M Barahona. Stability of graph communities across time scales. *Proceedings of the National Academy of Sciences of the United States of America*, 107(29) :12755–60, jul 2010.
- [6] Remi Dorat, Matthieu Latapy, Bernard Conein, and Nicolas Auray. Multi-level analysis of an interaction network between individuals in a mailing-list. *Ann Telecommun*, 62 :325–349, 2007.
- [7] Alcides Viamontes Esquivel and Martin Rosvall. Compression of Flow Can Reveal Overlapping-Module Organization in Networks. *Physical Review X*, 1 :1–11, 2011.
- [8] T. S. Evans and Renaud Lambiotte. Line graphs, link partitions, and overlapping communities. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 80(1) :016105, jul 2009.
- [9] Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(3–5) :75–174, feb 2010.
- [10] Lan Huang, Guishen Wang, Yan Wang, Enrico Blanzieri, and Chao Su. Link Clustering with Extended Link Similarity and EQ Evaluation Division. *PLoS ONE*, 8(6) :e66005, jun 2013.
- [11] Sungmin Kim. Community Detection in Directed Networks and its Application to Analysis of Social Networks. 2014.
- [12] Youngdo Kim and Hawoong Jeong. Map equation for link communities. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 84(2) :026110, aug 2011.
- [13] Andrea Lancichinetti and Santo Fortunato. Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 80 :016118, 2009.
- [14] Zhenping Li, Xiang-Sun Zhang, Rui-Sheng Wang, Hongwei Liu, and Shihua Zhang. Discovering link communities in complex networks by an integer programming model and a genetic algorithm. *PloS one*, 8 :e83739, 2013.

- [15] Sungsu Lim, Seungwoo Ryu, Sejeong Kwon, Kyomin Jung, and Jae-Gil Lee. LinkSCAN* : Overlapping community detection using the link-space transformation. In *2014 IEEE 30th International Conference on Data Engineering*, pages 292–303. IEEE, mar 2014.
- [16] M. E J Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 69, 2004.
- [17] Martin Rosvall and Carl T Bergstrom. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences of the United States of America*, 105(4) :1118–23, jan 2008.
- [18] Chuan Shi, Yanan Cai, Di Fu, Yuxiao Dong, and Bin Wu. A link clustering based overlapping community detection algorithm. In *Data and Knowledge Engineering*, volume 87, pages 394–404, 2013.
- [19] Zhihao Wu, Youfang Lin, Huaiyu Wan, and Shengfeng Tian. A fast and reasonable method for community detection with adjustable extent of overlapping. In *Proceedings of 2010 IEEE International Conference on Intelligent Systems and Knowledge Engineering, ISKE 2010*, pages 376–379, 2010.