Faculté des Sciences
Département de Mathématique

Recherche de communautés dans les graphes

réalisé par :
Hélène Berger

Promoteur :
Michel Rigo

Mémoire présenté en vue de l'obtention du grade de Master en Sciences Mathématiques à finalité informatique

Année académique : 2018–2019

Remerciements

« Pour ce qui est de l'avenir, il ne s'agit pas de le prévoir mais de le rendre possible. »

Antoine de Saint-Exupéry

Avant toutes choses, je souhaite remercier les personnes qui ont permis de rendre ce travail possible.

Tout d'abord, je souhaite remercier mon promoteur, Michel Rigo, de m'avoir encadré, guidé et corrigé dans la rédaction de ce travail et surtout, d'avoir toujours été disponible pour répondre à mes questions.

Je souhaite également remercier Célia Cisternino ainsi que Marie Lejeune pour leur relecture attentive et leurs précieux conseils mais également pour leur soutient durant toutes ces années.

Je tiens aussi à remercier tous les professeurs et assistants pour leur disponibilité et leur aide durant mes études au B37.

Enfin, je souhaite remercier toutes les personnes qui ont contribué de près ou de loin à mon parcours académique et plus particulièrement mes parents et ma soeur pour leurs encouragements et leur soutient tout au long de mon parcours, et qui m'ont permis d'arriver au terme de mes études.

Table des matières

Introduction

Un grand nombre de situations et problèmes de la vie réelle peuvent être modélisés par des graphes. Par exemple, en biologie, les réseaux métaboliques sont représentés par un graphe où les sommets sont les composés chimiques et deux sommets sont liés par une arête si il existe une réaction qui transforme le premier composé en un substrat qui est le deuxième sommet. En informatique, on peut créer un graphe pour représenter le « World Wide Web ». Dans ce cas-ci, les sommets sont les différentes pages web et il existe une arête entre deux sommets si une des deux pages référence l'autre par un hyperlien. Les réseaux sociaux peuvent également être modélisés par des graphes où les personnes sont représentées par les sommets et les liens d'amitié sont représentées grâce aux arêtes.

De nombreux chercheurs se sont naturellement intéressés à ce type de graphes et récemment, ils ont introduit un nouveau concept qui est la recherche de *communautés* dans les graphes. Une communauté est décrite comme étant un sous-ensemble de sommets fortement connectés entre eux, tandis que les arêtes entre des sommets appartenant à des communautés différentes sont rares. Les notions de « fortement » et « rare » restent subjectives et il n'existe pas encore de définition formelle d'une communauté. Cependant, de nombreux algorithmes, ainsi que des fonctions permettant d'évaluer la qualité d'une partition d'un graphe en sous-ensembles de sommets, ont été développés.

La recherche de communautés revêt un intérêt tout particulier pour les situations décrites ci-dessus. En effet, pour un réseau métabolique, connaître sa topologie fournit de nombreuses informations sur le fonctionnement du système et permet de prédire des réactions ou de stabiliser le système [6]. Concernant la représentation des liens entre les pages web, la recherche de communautés permet de les classer grâce à un index et ainsi mieux orienter l'utilisateur dans ses recherches [2]. Enfin, connaître les communautés présentes dans les réseaux sociaux permet aux applications de ces réseaux de proposer de nouveaux liens de façon intelligente mais a également permis à des enquêteurs d'identifier des individus susceptibles d'être des terroristes [16].

Le but de ce travail est d'étudier et d'appliquer deux algorithmes de détection

de communautés dans les graphes. D'une part, la méthode du *chemin aléatoire* développée par Pascal Pons et Matthieu Latapy [9]. D'autre part, la méthode heuristique introduite par Vincent Blondel, Jean-Loup Guillaume, Renaud Lambiotte et Etienne Lefebvre [3] .

Le premier chapitre de ce mémoire rappelle les notions de base de théorie des graphes et d'algèbre linéaire nécessaires pour la compréhension des chapitres suivants et permet également de fixer les notations utilisées dans ce travail.

Le deuxième chapitre est le développement de la méthode du chemin aléatoire. Tout d'abord, on définit formellement ce qu'est un chemin aléatoire et on décrit les grandes étapes de l'algorithme. L'initialisation de la méthode consiste à considérer chaque sommet comme étant une communauté. Ensuite, à chaque étape on fusionne deux communautés en fonction des distances qui existent entre chaque paire de communautés, puis on met à jour les distances après la fusion. On crée ainsi une suite de partitions, et à la fin de l'algorithme, grâce à une fonction qui permet d'estimer la qualité d'une partition, on choisit la meilleure d'entre elles. Afin d'exécuter efficacement cet algorithme, dans ce chapitre on introduit la notion de distance entre deux communautés, on établit un critère de sélection pour choisir les communautés à fusionner et une méthode de calcul pour mettre à jour les distances de façon efficace et enfin un critère pour quantifier la qualité d'une partition. Finalement, on illustre l'algorithme sur un exemple et on calcule sa complexité théorique.

Le chapitre suivant présente le deuxième algorithme. Il s'agit d'une méthode heuristique qui est basée sur la notion de *modularité*. On explique donc tout d'abord la notion de modularité d'une partition dans un graphe qui permet de mesurer la qualité d'une partition. On décrit ensuite l'algorithme. Celui-ci est constitué de deux grandes étapes. La première consiste à fusionner au fur et à mesure des sommets afin d'augmenter la modularité. Quand cela n'est plus possible, la deuxième étape permet de créer un nouveau graphe plus petit et on recommence ensuite à la première étape jusqu'à ce que plus aucun changement ne soit observé. On illustre également cette méthode sur l'exemple du chapitre précédent.

Enfin, le dernier chapitre détaille l'implémentation des deux algorithmes réalisés pour ce mémoire. Ils sont implémentés dans le langage de programmation C et sont accessibles sur la plateforme « MatheO ». Ces deux programmes sont utilisés pour la réalisation des quelques exemples présentés dans ce mémoire. Ces exemples sont également réalisables à la main, cependant les implémentations sont indispensables pour exécuter les deux algorithmes sur la base de données institutionnelle « ORBi » de l'Université de Liège [1]. En effet, à partir de cette base de données, on va créer un graphe contenant un peu plus de 10.000 sommets qui représentent les chercheurs de l'Université de Liège et il existe une arête entre deux chercheurs si

ils ont écrit au moins un article ensemble. En fonction du nombre d'articles écrits en commun, un poids est également attribué aux arêtes. Ensuite, les algorithmes nous permettent d'identifier les différentes communautés de chercheurs au sein de l'Université. Une analyse du résultat est donnée à la fin de ce chapitre. Remarquons que cette interprétation n'est pas faite en profondeur afin de garder l'anonyma des chercheurs et respecter le règlement général sur la protection des données.

Chapitre 1

Notions de base

Tout d'abord, dans ce chapitre, nous allons rappeler quelques définitions et propriétés de théorie des graphes qui seront nécessaires pour la suite de ce travail. Cela nous permettra également de fixer les notations utilisées dans ce travail. Ces notions sont présentes de manière équivalente dans le cours de théorie des graphes [11]. De plus, nous énoncerons quelques résultats classiques d'algèbre linéaire qui nous serons utiles dans la section 2.3.2. Nous ne justifierons pas ces résultats classiques dans ce travail mais les preuves sont présentes dans le cours [10].

Commençons tout d'abord par les notions de base de théorie des graphes. Soient V un ensemble et E une relation binaire sur $V \times V$. Le graphe $G = (V, E)$ est la donnée du couple (V, E). Les éléments de V sont appelés les *sommets* de G et les éléments de E sont appelés les *arêtes* de G. Nous prendrons comme convention que le graphe G possède n sommets et m arêtes.

Chaque arête du graphe est identifiée par un couple (v_i, v_j) où v_i est l'origine et v_j l'extrémité de l'arête, et dans ce cas on parle de *graphe orienté*. En particulier, si E est une relation symétrique sur V, on dira que G est un *graphe non orienté* et on identifiera les arêtes (v_i, v_j) et (v_j, v_i) par une unique paire $\{v_i, v_j\}$.

On dira que G est un *multi-graphe* si l'ensemble E des arêtes est un multi-ensemble, i.e., un ensemble dans lequel un élément peut être répété plus d'une fois. On définit également G comme étant un multi-graphe *pondéré* si il existe une fonction $f : E \to \Sigma$ où $\Sigma \subseteq [0, +\infty[$. Dans ce cas, si e_i est une arête de E, on appelle $f(e_i)$ le label ou le poids de e_i, $i \in \{1, ..., m\}$. La *matrice d'adjacence* A d'un multi-graphe G est une matrice carrée de dimension n où l'élément A_{ij} est égal au nombre d'arêtes $\{v_i, v_j\}$ présentes dans E, avec $i, j \in \{1, ..., n\}$.

On peut à présent définir la notion de chemin. Un *chemin de longueur* $k \geq 1$ est la donnée d'une suite ordonnée $(e_1, ..., e_k)$ de k arêtes adjacentes, i.e., pour tous

$i \in \{1, ..., k-1\}$, l'extrémité de l'arc e_i correspond à l'origine de l'arc e_{i+1}. Ainsi, on dit qu'un graphe non orienté est *connexe* si, pour tout couple de sommets, il existe un chemin les joignant.

Nous allons à présent énoncer les quelques définitions et propriétés d'algèbre linéaire qui nous seront utiles pour la suite de ce travail.

Définition 1.1. Soit A une matrice carrée de dimension n à coefficients positifs ou nuls. La matrice A est *irréductible* si pour tous $i, j \in \{1, ..., n\}$, il existe un entier $N(i,j) > 0$ tel que $[A^{N(i,j)}]_{ij} > 0$. En particulier, on dit que la matrice A est *primitive* si il existe un entier $N > 0$ tel que pour tous $i, j \in \{1, ..., n\}, [A^N]_{ij} > 0$.

Proposition 1.2. *Un multi-graphe non orienté est connexe si et seulement si sa matrice d'adjacence est irréductible.*

Définition 1.3. S'il existe un entier $N > 0$ tel que $[A^N]_{ii} > 0$, la *période* de l'indice i est le plus grand commun diviseur de l'ensemble des entiers $p > 0$ pour lesquels $[A^p]_{ii} > 0$.

On peut étendre cette définition aux graphes. On définit la période d'un sommet v_i comme étant le plus grand commun diviseur de l'ensemble des entiers $p > 0$ pour lesquels il existe un circuit de longueur p passant par v_i. On peut facilement montrer que deux sommets d'une même composante connexe ont la même période. Ainsi, si la matrice d'adjacence est irréductible, alors tous les sommets ont même période.

Définition 1.4. Une matrice irréductible A à coefficients réels est *acyclique* si tous les indices de A sont de période égale à 1.

On peut également étendre cette définition aux graphes, et on dit qu'un graphe est apériodique si tous les sommets sont de période égale à 1.

Proposition 1.5. *Une matrice irréductible est acyclique si et seulement si elle est primitive.*

Définition 1.6. Soient E_v un espace vectoriel et T un endomorphisme de E_v. Le nombre complexe λ est une *valeur propre* de T s'il existe $v \in E_v \backslash \{0\}$ tel que $Tv = \lambda v$. Le vecteur $v \in E_v$ est appelé *vecteur propre* de T de valeur propre λ.

Théorème 1.7 (Perron)**.** *Soit $A \geq 0$ une matrice carrée primitive de dimension n.*

- *La matrice A possède un vecteur propre $v \in \mathbb{R}^n$ dont les composantes sont toutes strictement positives et correspondant à une valeur propre $\lambda > 0$.*
- *Cette valeur propre λ possède une multiplicité algébrique simple.*

— *Toute autre valeur propre* $\alpha \in \mathbb{C}$ *de* A *est telle que* $|\alpha| < \lambda$.

Proposition 1.8. *Si* S *est une matrice stochastique, i.e., si la somme des éléments de chacune de ses lignes vaut* 1*, alors* 1 *est la valeur propre dominante de* S*. De plus,* $(1, ..., 1)^\sim$ *est un vecteur propre de valeur propre* 1*.*

Proposition 1.9. *Si* A *et* B *sont deux matrices semblables, i.e., s'il existe une matrice inversible* S *telle que* $A = S^{-1}BS$*, alors elles possèdent les mêmes valeurs propres avec les mêmes multiplicités.*

Soit A une matrice complexe. La *matrice adjointe* de A est la matrice $A^* = \overline{A}^\sim$, i.e., la transposée de la matrice A où tous les éléments sont conjugués. On dit que A est *normale* si $A^*A = AA^*$ et on dit qu'elle est *hermitienne* si $A^* = A$.

Proposition 1.10. *Les vecteurs propres d'une matrice normale* N *associés à des valeurs propres différentes sont orthogonaux.*

Proposition 1.11. *Si* H *est une matrice hermitienne, alors les valeurs propres de* H *sont réelles.*

Grâce à ces notions de base et ces quelques propriétés, nous allons pouvoir développer deux méthodes qui nous permettront d'extraire des communautés dans les graphes.

Chapitre 2

Méthode du chemin aléatoire

Une première méthode pour pouvoir trouver des communautés dans les graphes est une méthode basée sur les chemins aléatoires. Cette méthode a été développée par Pascal Pons et Matthieu Latapy [9].

Dans ce chapitre, nous allons considérer que $G = (V, E)$ est un graphe non orienté connexe tel que $|V| = n$ et $|E| = m$. Si la connexité n'est pas vérifiée, il suffit de considérer chaque composante connexe séparément. Nous supposerons également que chaque sommet est lié avec lui-même grâce à une boucle. Ainsi le graphe sera apériodique. Dans l'article [9], les auteurs considèrent également que le graphe est non pondéré. Nous généraliserons cette méthode aux graphes pondérés dans la section 2.10.

2.1 Préliminaires

Afin de pouvoir décrire la méthode du chemin aléatoire, nous avons besoin de la définition d'un tel chemin. Pour cela, définissons tout d'abord ce qu'est une chaîne de Markov. Cette définition est présente de manière similaire dans le cours [12].

Définition 2.1.1. Soit E un ensemble au plus dénombrable. Une suite $(X_n)_{n\geq 0}$ de variables aléatoires à valeurs dans E est une *chaîne de Markov* si et seulement si pour tout $k \in \mathbb{N}$ et pour tous $x_0, ..., x_{k+1}$ dans E tels que $\mathbb{P}(X_{k+1} = x_{k+1}, ..., X_0 = x_0) > 0$, on a

$$\mathbb{P}(X_{k+1} = x_{k+1} | X_k = x_k, ..., X_0 = x_0) = \mathbb{P}(X_{k+1} = x_{k+1} | X_k = x_k).$$

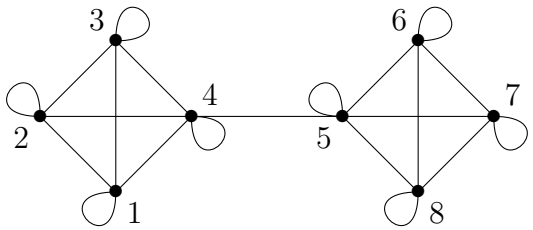
Autrement dit, une chaîne de Markov est un processus permettant de prédire, en fonction de l'état actuel, dans quel état on va arriver et où la probabilité de passer d'un état à un autre dépend uniquement de l'état actuel et est indépendant des états

parcourus précédemment. Nous pouvons dès à présent définir ce qu'est un chemin aléatoire.

Définition 2.1.2. Soit $G = (V, E)$ un multi-graphe. Un *chemin aléatoire discret* sur G est une suite de sommets $v_0, ..., v_k$ appartenant à V (et pas forcément distincts) tels que l'arête $\{v_i, v_{i+1}\}$ est choisie aléatoirement et uniformément parmi toutes les arêtes d'origine v_i, pour tout $i \in \{0, ..., k-1\}$.

Remarquons que la suite de sommets définissant un chemin aléatoire est une chaîne de Markov où, à chaque étape, la probabilité de passer du sommet v_i au sommet v_j est donné par $P_{ij} = \frac{A_{ij}}{d(i)}$, où A est la matrice d'adjacence du graphe G et $d(i)$ est le degré du sommet v_i, i.e., $d(i) = \sum_{j=1}^{n} A_{ij}$. Ceci nous permet de définir une matrice P que l'on appelle *matrice de transition*. Nous utiliserons la notation $P_{i\bullet}$ (respectivement $P_{\bullet j}$) lorsque l'on considère la $i$ème ligne (respectivement $j$ème colonne) de cette matrice. Ainsi, la probabilité de passer d'un sommet v_i à un sommet v_j en t étapes, i.e., en utilisant un chemin de longueur t, est donnée par $[P^t]_{ij}$. Remarquons également que si l'on pose D comme étant la matrice diagonale des degrés, i.e., $D_{ii} = d(i)$ pour tout $i \in \{1, ..., n\}$, et $D_{ij} = 0$ pour tous $i \neq j$, alors, vu que $d(i) \neq 0$ pour tout i, la matrice D est inversible et on a $P = D^{-1}A$.

Exemple 2.1.3. Illustrons ces quelques notions sur un exemple. Soit G le graphe suivant :



Graphe 2.1

Si on regarde le degré de chaque sommet, on a

$$\begin{cases} d(i) = 5 & \text{si } i = 4 \text{ ou } i = 5, \\ d(i) = 4 & \text{sinon.} \end{cases}$$

La matrice d'adjacence et la matrice de transition du graphe G sont

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}, P = \begin{pmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 \\ \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \\ 0 & 0 & 0 & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ 0 & 0 & 0 & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ 0 & 0 & 0 & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \end{pmatrix}.$$

Grâce à ces notions, nous obtenons les deux propriétés suivantes concernant la matrice de transition P.

Proposition 2.1.4. *Les probabilités de passer d'un sommet v_i à un sommet v_j et de passer du sommet v_j au sommet v_i en utilisant un chemin aléatoire de longueur t ont un ratio qui dépend uniquement des degrés $d(i)$ et $d(j)$, pour tous $v_i, v_j \in V$. On a, pour tous $v_i, v_j \in V$,*

$$d(i)[P^t]_{ij} = d(j)[P^t]_{ji}.$$

Exemple 2.1.5. Illustrons cette propriété sur le Graphe 2.1 et regardons les probabilités de passer respectivement du sommet 3 au sommet 4 et de passer du sommet 4 au sommet 3 en utilisant un chemin aléatoire de longueur 2. Pour cela, on a besoin de l'élément situé à la troisième ligne et quatrième colonne ainsi que de celui situé à la quatrième ligne et troisième colonne de la matrice P^2. On a

$$[P^2]_{34} = \begin{pmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{5} & \frac{1}{5} & 0 & 0 & 0 \end{pmatrix}^{\sim} = \frac{3}{16} + \frac{1}{20}$$

et

$$[P^2]_{43} = \begin{pmatrix} \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{5} & 0 & 0 & 0 & 0 \end{pmatrix}^{\sim} = \frac{3}{20} + \frac{1}{25}.$$

Or $d(3) = 4$ et $d(4) = 5$, donc

$$d(3)[P^2]_{34} = 4\left(\frac{3}{16} + \frac{1}{20}\right) = \frac{3}{4} + \frac{1}{5}$$

et

$$d(4)[P^2]_{43} = 5\Big(\frac{3}{20} + \frac{1}{25}\Big) = \frac{3}{4} + \frac{1}{5}.$$

D'où, on obtient $d(3)[P^2]_{34} = d(4)[P^2]_{43}$. A présent, démontrons cette proposition en toute généralité.

Démonstration. On a, vu que $P = D^{-1}A$,

$$DP^tD^{-1} = D(D^{-1}A)^tD^{-1} = (AD^{-1})^t.$$

De plus, si S est une matrice symétrique, alors $S = S^{\sim}$ et S^{-1} est également une matrice symétrique. Or, D et A sont deux matrices symétriques, donc

$$DP^tD^{-1} = (AD^{-1})^t = (A^{\sim}(D^{-1})^{\sim})^t = ((D^{-1}A)^{\sim})^t = (P^{\sim})^t.$$

Ainsi, on obtient

$$DP^t = (P^{\sim})^tD.$$

Donc, pour tous $i, j \in \{1, ..., n\}$, on a

$$\begin{aligned} &[DP^t]_{ij} = [(P^{\sim})^tD]_{ij} \\ \Leftrightarrow\ &d(i)[P^t]_{ij} = [(P^t)^{\sim}]_{ij}d(j) \\ \Leftrightarrow\ &d(i)[P^t]_{ij} = [P^t]_{ji}d(j). \end{aligned}$$

Ainsi, $d(i)[P^t]_{ij} = d(j)[P^t]_{ji}$.

□

Proposition 2.1.6 (Distribution stationnaire)**.** *Lorsque la longueur t d'un chemin aléatoire démarrant au sommet v_i tend vers l'infini, la probabilité d'arriver à un sommet v_j dépend uniquement du degré du sommet v_j et ne dépend pas du sommet d'origine. Formellement,*

$$\lim_{t\to+\infty}[P^t]_{ij} = \frac{d(j)}{\sum_{k=1}^{n} d(k)}.$$

Nous démontrerons cette propriété lorsque nous démontrerons le Théorème 2.3.5. Cependant, nous pouvons l'illustrer sur notre exemple.

Exemple 2.1.7. En utilisant le logiciel Mathematica, on peut effectuer une simulation numérique afin de vérifier que cette propriété s'applique bien au Graphe 2.1.

$$\boldsymbol{A = \{\{1,1,1,1,0,0,0,0\},\{1,1,1,1,0,0,0,0\},\{1,1,1,1,0,0,0,0\},}$$

$$\boldsymbol{\{1,1,1,1,1,0,0,0\},\{0,0,0,1,1,1,1,1\},\{0,0,0,0,1,1,1,1\},}$$

$$\boldsymbol{\{0,0,0,0,1,1,1,1\},\{0,0,0,0,1,1,1,1\}\};}$$

$d = A.\text{Table}[1, \{i, 1, 8\}];$

$P = \text{Table}[A[[i]]/d[[i]], \{i, 1, 8\}];$

$\text{Limit}[\text{MatrixPower}[P, t], t \to \text{Infinity}]$

$\left\{\left\{\frac{2}{17}, \frac{2}{17}, \frac{2}{17}, \frac{5}{34}, \frac{5}{34}, \frac{2}{17}, \frac{2}{17}, \frac{2}{17}\right\}, \left\{\frac{2}{17}, \frac{2}{17}, \frac{2}{17}, \frac{5}{34}, \frac{5}{34}, \frac{2}{17}, \frac{2}{17}, \frac{2}{17}\right\},\right.$
$\left\{\frac{2}{17}, \frac{2}{17}, \frac{2}{17}, \frac{5}{34}, \frac{5}{34}, \frac{2}{17}, \frac{2}{17}, \frac{2}{17}\right\}, \left\{\frac{2}{17}, \frac{2}{17}, \frac{2}{17}, \frac{5}{34}, \frac{5}{34}, \frac{2}{17}, \frac{2}{17}, \frac{2}{17}\right\},$
$\left\{\frac{2}{17}, \frac{2}{17}, \frac{2}{17}, \frac{5}{34}, \frac{5}{34}, \frac{2}{17}, \frac{2}{17}, \frac{2}{17}\right\}, \left\{\frac{2}{17}, \frac{2}{17}, \frac{2}{17}, \frac{5}{34}, \frac{5}{34}, \frac{2}{17}, \frac{2}{17}, \frac{2}{17}\right\},$
$\left.\left\{\frac{2}{17}, \frac{2}{17}, \frac{2}{17}, \frac{5}{34}, \frac{5}{34}, \frac{2}{17}, \frac{2}{17}, \frac{2}{17}\right\}, \left\{\frac{2}{17}, \frac{2}{17}, \frac{2}{17}, \frac{5}{34}, \frac{5}{34}, \frac{2}{17}, \frac{2}{17}, \frac{2}{17}\right\}\right\}$

$\text{Table}[d[[i]]/d.\text{Table}[1, \{k, 1, 8\}], \{i, 1, 8\}]$

$\left\{\frac{2}{17}, \frac{2}{17}, \frac{2}{17}, \frac{5}{34}, \frac{5}{34}, \frac{2}{17}, \frac{2}{17}, \frac{2}{17}\right\}$

On a donc, pour un i fixé,

$$\lim_{t \to +\infty} [P^t]_{ij} = \left[\frac{2}{17}, \frac{2}{17}, \frac{2}{17}, \frac{5}{34}, \frac{5}{34}, \frac{2}{17}, \frac{2}{17}, \frac{2}{17}\right]_j = \frac{d(j)}{\sum_{k=1}^{8} d(k)}.$$

2.2 Algorithme

Le but de l'algorithme décrit ci-dessous, est d'extraire des communautés présentes dans le graphe $G = (V, E)$. La première étape de l'algorithme est de considérer chaque sommet $v \in V$ comme étant une communauté. On dispose donc d'une première partition triviale du graphe $\mathcal{P}_1 = \{\{v\}, v \in V\}$ en n communautés. Ensuite, on calcule la distance entre chacune de ces communautés, ce qui revient à calculer la distance entre toutes les paires de sommets du graphe. La notion de distance qui sera utilisée pour cette étape sera décrite dans la section suivante. Ensuite, nous allons modifier la partition $\mathcal{P}_1$. Pour ce faire, à chaque étape $k \geq 1$,

1. on choisit, grâce à un critère de sélection, deux communautés $\mathcal{C}_1$ et $\mathcal{C}_2$ de $\mathcal{P}_k$,
2. on les fusionne en une nouvelle communauté $\mathcal{C}_3 = \mathcal{C}_1 \cup \mathcal{C}_2$, et on crée ainsi une nouvelle partition $\mathcal{P}_{k+1} = (\mathcal{P}_k \backslash \{\mathcal{C}_1, \mathcal{C}_2\}) \cup \{\mathcal{C}_3\}$,
3. on met à jour la distance entre les communautés.

On itère ces étapes $n-1$ fois et on obtient $\mathcal{P}_n = \{V\}$. Le choix des deux communautés à fusionner se fait grâce à un critère de sélection basé sur la distance entre deux communautés. Nous détaillerons ce critère dans la section 2.4. Nous verrons également comment la distance entre les communautés peut être modifiée efficacement. Lorsque l'on effectue cet algorithme, on crée parallèlement un dendrogramme. Il s'agit d'un arbre pour lequel chaque feuille correspond à un sommet du graphe de départ, et chaque noeud interne est la fusion de ses deux fils.

Lorsque cet algorithme est terminé, il reste à choisir la partition $\mathcal{P}_k$ qui représente le mieux la séparation des communautés. Nous détaillerons ce choix dans la section 2.6.

2.3 Construction d'une distance appropriée

2.3.1 Définitions

Nous allons à présent introduire une distance entre deux sommets. Cette distance doit pouvoir faire ressortir les communautés. Autrement dit, la distance entre deux sommets doit être « petite » (resp. « grande ») si les deux sommets appartiennent à la même communauté (resp. à des communautés différentes). On va pouvoir satisfaire ces contraintes grâce aux chemins aléatoires et à la matrice de transition P que nous avons introduits précédemment.

Considérons donc un chemin aléatoire de longueur t. La longueur de ce chemin doit être assez grande pour que l'on puisse avoir assez d'information sur la structure du graphe mais ne doit pas être trop grande afin d'éviter la distribution stationnaire donnée dans la Proposition 2.1.6. Nous pouvons donc définir une distance entre deux sommets de la façon suivante :

Définition 2.3.1. Soient $i, j \in \{1, ..., n\}$. La *distance* entre le sommet v_i et le sommet v_j est donnée par

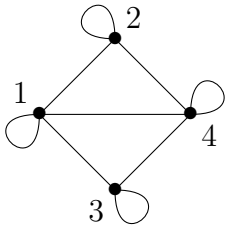
$$r_t(i,j) = \sqrt{\sum_{k=1}^{n} \frac{([P^t]_{ik} - [P^t]_{jk})^2}{d(k)}}.$$

Notons qu'il ne s'agit pas, au sens strict, d'une distance métrique. En effet, la distance entre deux points est bien positive, elle est symétrique et l'inégalité trian-

gulaire est bien vérifiée comme détaillé ci-dessous :

$$\begin{aligned} r_t(i,j) &= \sqrt{\sum_{k=1}^{n} \frac{([P^t]_{ik} - [P^t]_{jk})^2}{d(k)}} = \sqrt{\sum_{k=1}^{n} \frac{([P^t]_{ik} - [P^t]_{lk} + [P^t]_{lk} - [P^t]_{jk})^2}{d(k)}} \\ &\leq \sqrt{\sum_{k=1}^{n} \frac{([P^t]_{ik} - [P^t]_{lk})^2 + ([P^t]_{lk} - [P^t]_{jk})^2}{d(k)}} \\ &\leq r_t(i,l) + r_t(l,j) \qquad\qquad \forall i,j,l \in \{1,...,n\}. \end{aligned}$$

De plus, si le sommet v_i et le sommet v_j sont égaux, alors la distance entre ces deux sommets est nulle. Cependant, la réciproque est fausse. Voici un contre-exemple.



Les degrés des sommets 1 et 4 sont égaux à 4 et ceux des deux autres sommets sont égaux à 3. Les matrices d'adjacence et de transition de ce graphe sont les suivantes :

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}, P = \begin{pmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{3} & \frac{1}{3} & 0 & \frac{1}{3} \\ \frac{1}{3} & 0 & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \end{pmatrix}.$$

D'où, si on calcule la distance entre le sommet 1 et 4, en prenant $t = 1$, on obtient

$$r_1(1,4) = \sqrt{\sum_{k=1}^{4} \frac{([P^1]_{1k} - [P^1]_{4k})^2}{d(k)}} = \sqrt{\sum_{k=1}^{4} \frac{(\frac{1}{4} - \frac{1}{4})^2}{d(k)}} = 0.$$

Or les sommets 1 et 4 sont distincts. Cette situation se produit quand deux sommets sont adjacents et qu'ils possèdent les mêmes voisins. Ainsi, les deux lignes de la matrice d'adjacence correspondant à ces sommets sont identiques, donc elles seront également égales dans la matrice de transition et dans les puissances de celle-ci. La distance sera donc nulle entre ces deux sommets particuliers. On n'a donc pas une distance métrique à proprement parler, mais cela a du sens d'utiliser le terme « distance ». En effet, même si le cas particulier se produit, les sommets sont adjacents et possèdent tous leurs voisins en commun donc on peut considérer qu'ils sont très proches l'un de l'autre et qu'ils seront probablement dans la même communauté.

Par la suite, afin de simplifier les notations, nous noterons $r(i,j)$ pour la distance entre les sommets v_i et v_j sans spécifier que cette distance dépend du paramètre t. Remarquons également que l'on peut réécrire de façon équivalente la distance entre le sommet v_i et le sommet v_j de la manière suivante

$$r(i,j) = \|D^{-\frac{1}{2}}[P^t]_{i\bullet} - D^{-\frac{1}{2}}[P^t]_{j\bullet}\|$$

où $\|.\|$ est la norme euclidienne sur $\mathbb{R}^n$.

A présent, essayons de généraliser cette distance aux communautés, qui sont composées d'un sous-ensemble de sommets. Considérons tout d'abord un chemin aléatoire dont le sommet origine se trouve dans la communauté $\mathcal{C}_0$. Ce sommet est choisi aléatoirement et uniformément parmi tous les sommets de $\mathcal{C}_0$. On définit la probabilité d'aller de la communauté $\mathcal{C}_0$ à un sommet v_j en utilisant un chemin de longueur t comme suit :

$$[P^t]_{\mathcal{C}_0 j} = \frac{1}{|\mathcal{C}_0|}\sum_{i\in\mathcal{C}_0}[P^t]_{ij}. \tag{2.1}$$

Ceci nous permet de généraliser la distance entre deux sommets à la distance entre deux communautés.

Définition 2.3.2. Soient $\mathcal{C}_1, \mathcal{C}_2$ deux communautés. La distance entre ces deux communautés est égale à

$$r(\mathcal{C}_1,\mathcal{C}_2) = \sqrt{\sum_{k=1}^{n}\frac{([P^t]_{\mathcal{C}_1 k} - [P^t]_{\mathcal{C}_2 k})^2}{d(k)}}.$$

Remarquons que si les deux communautés dans la définition se restreignent chacune à un seul sommet, nous obtenons bien la définition de distance entre deux sommets. A nouveau, on n'a pas une distance à proprement parler. En effet, par le même argument, si deux communautés différentes sont voisines et possèdent les

mêmes voisins, alors la distance entre ces deux communautés sera nulle. Mais tout comme le cas particulier de la distance entre deux sommets, cela a du sens d'utiliser le terme « distance ».

2.3.2 Justification du choix de la distance

Dans de nombreux articles (comme par exemple [4], [8], et [13]), la distance entre deux sommets est calculée grâce aux propriétés spectrales du graphe. La raison est que ces propriétés spectrales ont une grande importance dans la recherche des communautés dans le graphe. Cependant, toutes ces méthodes nécessitent un grand nombre d'opérations afin de calculer les vecteurs propres. En effet, cela nécessite de l'ordre de $\mathcal{O}(n^3)$ opérations. Notre distance, quant à elle, nécessite uniquement le calcul des probabilités $[P^t]_{ij}$ qui peut se faire de façon efficace comme nous le verrons plus loin. De plus, notre définition fait intervenir de manière indirecte les propriétés spectrales du graphe, comme nous allons le montrer maintenant. Notre algorithme tiendra donc bien compte de ces propriétés tout en étant exécuté de manière efficace.

Remarque 2.3.3. La racine carrée d'une matrice A est notée $A^{\frac{1}{2}}$ et est telle que $A^{\frac{1}{2}}A^{\frac{1}{2}} = A$.

Lemme 2.3.4. *Les valeurs propres de la matrice de transition P sont réelles et sont telles que*

$$-1 < \lambda_n \leq \cdots \leq \lambda_2 < \lambda_1 = 1.$$

De plus, il existe une famille de vecteurs orthonormés $(s_\alpha)_{1\leq\alpha\leq n}$ tels que $v_\alpha = D^{-\frac{1}{2}}s_\alpha$ et $u_\alpha = D^{\frac{1}{2}}s_\alpha$ sont respectivement un vecteur propre à droite et un vecteur propre à gauche associés à la valeur propre λ_α et tels que $\forall\alpha, \beta$

$$\begin{aligned} Pv_\alpha &= \lambda_\alpha v_\alpha, \\ P^{\sim}u_\alpha &= \lambda_\alpha u_\alpha, \\ u_\alpha^{\sim}v_\beta &= \delta_{\alpha\beta}. \end{aligned}$$

Démonstration. Démontrons la première partie de l'énoncé. Soient P la matrice de transition et D la matrice diagonale des degrés. Posons S la matrice telle que $S = D^{\frac{1}{2}}PD^{-\frac{1}{2}}$. Dans ce cas, la matrice S et la matrice P sont des matrices semblables. Ainsi, par la Proposition 1.9, les matrices P et S possèdent les mêmes valeurs propres λ_α.

D'une part, on sait que le graphe G est connexe. D'où, vu la Proposition 1.2, la matrice d'adjacence de G est irréductible, donc la matrice de transition P aussi. De plus, on a supposé que G est apériodique. La matrice P est donc primitive par

la Proposition 1.5. Ainsi, en appliquant le Théorème de Perron 1.7, on a qu'il existe une valeur propre λ_1 de P de multiplicité algébrique simple et telle que toute autre valeur propre complexe de P est de module strictement inférieure à λ_1.

D'autre part, vu que $P = D^{-1}A$, on a $S = D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$. Comme les matrices A et D sont réelles et symétriques, on a

$$\begin{aligned} S^{\sim} &= (D^{-\frac{1}{2}}AD^{-\frac{1}{2}})^{\sim} \\ &= (D^{-\frac{1}{2}})^{\sim}A^{\sim}(D^{-\frac{1}{2}})^{\sim} \\ &= D^{-\frac{1}{2}}AD^{-\frac{1}{2}} \\ &= S \end{aligned}$$

et donc la matrice S est également symétrique. Ainsi, S est une matrice hermitienne, et en employant la Proposition 1.11, on obtient que les valeurs propres de S sont réelles. Par conséquent, celles de P sont également réelles.

Il existe donc une valeur λ_1 de P de multiplicité algébrique simple et telle que $-\lambda_1 < \lambda_n \leq \cdots \leq \lambda_2 < \lambda_1$.

Or, on sait que P est une matrice stochastique puisque $\sum_{j=1}^{n} P_{ij} = 1$. Par conséquent, vu la Proposition 1.8, la plus grande valeur propre de P vaut $\lambda_1 = 1$. On a donc $-1 < \lambda_n \leq \cdots \leq \lambda_2 < \lambda_1 = 1$.

A présent, montrons la deuxième partie de l'énoncé. Par le Lemme 1.10 et vu que S est symétrique, ses vecteurs propres s_α qui correspondent à des valeurs propres λ_α différentes sont orthogonaux. D'où, $s_\alpha^T s_\beta = \delta_{\alpha\beta}$ pour tous α, β. Posons $v_\alpha = D^{-\frac{1}{2}}s_\alpha$ et $u_\alpha = D^{\frac{1}{2}}s_\alpha$. On a

$$Pv_\alpha = PD^{-\frac{1}{2}}s_\alpha = D^{-1}AD^{-\frac{1}{2}}s_\alpha = D^{-\frac{1}{2}}Ss_\alpha = D^{-\frac{1}{2}}\lambda_\alpha s_\alpha = \lambda_\alpha v_\alpha$$

et

$$\begin{aligned} P^{\sim}u_\alpha &= P^{\sim}D^{\frac{1}{2}}s_\alpha = (D^{-1}A)^{\sim}D^{\frac{1}{2}}s_\alpha = A^{\sim}(D^{-1})^{\sim}D^{\frac{1}{2}}s_\alpha = AD^{-\frac{1}{2}}s_\alpha \\ &= D^{\frac{1}{2}}Ss_\alpha = D^{\frac{1}{2}}\lambda_\alpha s_\alpha = \lambda_\alpha u_\alpha. \end{aligned}$$

Donc v_α (resp. u_α) est un vecteur propre à droite (resp. à gauche) de P et on a

$$\begin{aligned} u_\alpha^{\sim}v_\beta &= s_\alpha^{\sim}(D^{\frac{1}{2}})^{\sim}D^{-\frac{1}{2}}s_\beta \\ &= s_\alpha^{\sim}s_\beta \\ &= \delta_{\alpha\beta}. \end{aligned}$$

□

Théorème 2.3.5. *Soit P la matrice de transition du graphe G. Soient $(\lambda_\alpha)_{1\leq\alpha\leq n}$ les valeurs propres associées à la matrice P et, pour tout $\alpha \in \{1, ..., n\}$, on considère un vecteur propre à droite v_α associé à la valeur propre λ_α. La distance r satisfait l'égalité suivante :*

$$r^2(i,j) = \sum_{\alpha=2}^{n} \lambda_\alpha^{2t} \big(v_\alpha(i) - v_\alpha(j)\big)^2.$$

Démonstration. Soient $\lambda_1, ..., \lambda_n$ les valeurs propres de la matrice de transition P et, pour tout $\alpha \in \{1, ..., n\}$, v_α, u_α les vecteurs propres à droite et à gauche associés à la valeur propre λ_α. Vu le Lemme 2.3.4, on sait que les vecteurs u_α, v_β sont tels que $u_\alpha^\sim v_\beta = \delta_{\alpha\beta}$.
D'où, pour tous β,

$$\left(\sum_{\alpha=1}^{n} \lambda_\alpha v_\alpha u_\alpha^\sim\right) v_\beta = \lambda_\beta v_\beta u_\beta^\sim v_\beta = \lambda_\beta v_\beta = P v_\beta.$$

Ainsi,

$$P = \sum_{\alpha=1}^{n} \lambda_\alpha v_\alpha u_\alpha^\sim$$

et donc

$$P^t = \sum_{\alpha=1}^{n} \lambda_\alpha^t v_\alpha u_\alpha^\sim \tag{2.2}$$

i.e.,

$$[P^t]_{ij} = \sum_{\alpha=1}^{n} \lambda_\alpha^t v_\alpha(i) u_\alpha(j).$$

Vu le Lemme 2.3.4, on sait également que $-1 < \lambda_n \leq \cdots \leq \lambda_2 < \lambda_1 = 1$, par conséquent, pour tout $\alpha \geq 2$, on a

$$\lim_{t\to+\infty} \lambda_\alpha^t = 0.$$

On obtient

$$\lim_{t\to+\infty} [P^t]_{ij} = \lim_{t\to+\infty} \sum_{\alpha=1}^{n} \lambda_\alpha^t v_\alpha(i) u_\alpha(j) = \lambda_1 v_1(i) u_1(j) = v_1(i) u_1(j).$$

De plus, par le lemme précédent, on sait que $\lambda_1 = 1$ est une valeur propre de P de vecteur propre v_1. D'où, $Pv_1 = v_1$. Or la somme de chaque ligne de P

vaut 1. Ainsi, $\lambda_1 = 1$ est une valeur propre de vecteur propre $v_1 = (1, ..., 1)^\sim$. Par le Théorème de Perron 1.7, on sait que cette valeur propre est simple, donc si w est un vecteur propre de P de valeur propre 1 alors w est multiple de $(1, ..., 1)^\sim$. On peut dès lors poser, $v_1(i) = \frac{1}{\sqrt{\sum_{k=1}^n d(k)}}$ pour tout i.

Vu le Lemme 2.3.4, on a donc $u_1^\sim v_1 = \delta_{11}$, d'où $u_1(j) = \frac{d(j)}{\sqrt{\sum_{k=1}^n d(k)}}$. Ainsi

$$\lim_{t\to+\infty}[P^t]_{ij} = v_1(i)u_1(j) = \frac{d(j)}{\sum_{k=1}^n d(k)}$$

ce qui est le résultat attendu pour la Proposition 2.1.6.
En appliquant l'égalité (2.2) et le Lemme 2.3.4, on obtient également

$$\begin{aligned}
[P^t]_{i\bullet} &= \sum_{\alpha=1}^n \lambda_\alpha^t v_\alpha(i) u_\alpha \\
&= \sum_{\alpha=1}^n \lambda_\alpha^t v_\alpha(i) D^{\frac{1}{2}} s_\alpha \\
&= D^{\frac{1}{2}} \sum_{\alpha=1}^n \lambda_\alpha^t v_\alpha(i) s_\alpha.
\end{aligned}$$

En utilisant la Définition 2.3.1 de la distance entre deux sommets, on obtient

$$\begin{aligned}
r^2(i,j) &= \left\| D^{-\frac{1}{2}}[P^t]_{i\bullet} - D^{-\frac{1}{2}}[P^t]_{j\bullet} \right\|^2 \\
&= \left\| D^{-\frac{1}{2}} D^{\frac{1}{2}} \sum_{\alpha=1}^n \lambda_\alpha^t v_\alpha(i) s_\alpha - D^{-\frac{1}{2}} D^{\frac{1}{2}} \sum_{\alpha=1}^n \lambda_\alpha^t v_\alpha(j) s_\alpha \right\|^2 \\
&= \left\| \sum_{\alpha=1}^n \lambda_\alpha^t (v_\alpha(i) - v_\alpha(j)) s_\alpha \right\|^2.
\end{aligned}$$

Ainsi, vu que les vecteurs s_α sont deux à deux orthogonaux, par le théorème de Pythagore, on obtient

$$\begin{aligned}
r^2(i,j) &= \Big(\sum_{\alpha=1}^n \lambda_\alpha^t \big(v_\alpha(i) - v_\alpha(j)\big) \|s_\alpha\| \Big)^2 \\
&= \sum_{\alpha=1}^n \lambda_\alpha^{2t} \big(v_\alpha(i) - v_\alpha(j)\big)^2.
\end{aligned}$$

□

Remarquons que dans la démonstration de ce théorème, nous obtenons la preuve de la Proposition 2.1.6. Vu le théorème précédent, on peut conclure que la distance construite tient bien compte des propriétés spectrales du graphe. La méthode développée ici sera donc construite sur les mêmes bases mais aura l'avantage d'être exécutée plus rapidement.

2.4 Choix des communautés à fusionner

Le choix des communautés à fusionner est très important pour faire ressortir la structure du graphe. Afin de réduire la complexité (ce qui sera détaillé dans la section 2.9) et pour assurer que le graphe reste connexe, nous allons fusionner uniquement des communautés qui sont adjacentes, i.e., qui possèdent au moins une arête entre elles. Pour cela, nous allons utiliser la méthode de Ward. Cette méthode est issue de l'article [15], et a été adaptée à notre situation. Le raisonnement de Ward est le suivant : à l'étape k, on fusionne les deux communautés qui vont minimiser la fonction σ_k définie comme suit :

$$\sigma_k = \frac{1}{n} \sum_{\mathcal{C} \in \mathcal{P}_k} \sum_{i \in \mathcal{C}} r^2(i, \mathcal{C}).$$

Le but de cette méthode est de minimiser la moyenne des distances au carré de chaque sommet avec la communauté à laquelle il appartient. A l'étape k, on va donc sélectionner et fusionner les deux communautés les plus proches afin que la valeur de σ_k soit minimale. Cependant, minimiser σ_k pour tout k est un problème connu pour être NP-difficile. En effet, il est équivalent au problème planaire des k-moyennes [1] qui est NP-difficile. Ce résultat est démontré dans l'article [7]. Cependant, lorsque l'on utilise une distance euclidienne, il existe des algorithmes polynomiaux pour résoudre ce problème. Pour ce faire, nous allons introduire une nouvelle définition, qui est la variation $\Delta\sigma(\mathcal{C}_1, \mathcal{C}_2)$ de σ qui est engendrée si on fusionne les communautés $\mathcal{C}_1$ et $\mathcal{C}_2$ en une nouvelle communauté $\mathcal{C}_3$. On définit cette variation comme suit :

$$\Delta\sigma(\mathcal{C}_1, \mathcal{C}_2) = \frac{1}{n}\Big(\sum_{i \in \mathcal{C}_3} r^2(i, \mathcal{C}_3) - \sum_{i \in \mathcal{C}_1} r^2(i, \mathcal{C}_1) - \sum_{i \in \mathcal{C}_2} r^2(i, \mathcal{C}_2) \Big). \tag{2.3}$$

1. Soient k un entier donné et S un ensemble de points. Le problème des k-moyennes est de trouver k centres qui vont minimiser la somme des carrés des distances de chaque point de S au centre le plus proche.

Remarquons que minimiser σ_k est équivalent à minimiser $\Delta\sigma(\mathcal{C}_1, \mathcal{C}_2)$. En effet, si $\mathcal{C}_1 \cup \mathcal{C}_2 = \mathcal{C}_3$, alors

$$\begin{aligned}
\sigma_k - \sigma_{k-1} &= \frac{1}{n} \sum_{\mathcal{C} \in \mathcal{P}_k} \sum_{i \in \mathcal{C}} r^2(i, \mathcal{C}) - \frac{1}{n} \sum_{\mathcal{C} \in \mathcal{P}_{k-1}} \sum_{i \in \mathcal{C}} r^2(i, \mathcal{C}) \\
&= \frac{1}{n} \sum_{\mathcal{C} \in \mathcal{P}_k \setminus \mathcal{C}_3} \sum_{i \in \mathcal{C}} r^2(i, \mathcal{C}) + \frac{1}{n} \sum_{i \in \mathcal{C}_3} r^2(i, \mathcal{C}_3) \\
&\quad - \Big(\frac{1}{n} \sum_{\mathcal{C} \in \mathcal{P}_{k-1} \setminus \{\mathcal{C}_1, \mathcal{C}_2\}} \sum_{i \in \mathcal{C}} r^2(i, \mathcal{C}) + \frac{1}{n} \sum_{i \in \mathcal{C}_1} r^2(i, \mathcal{C}_1) + \frac{1}{n} \sum_{i \in \mathcal{C}_2} r^2(i, \mathcal{C}_2) \Big) \\
&= \frac{1}{n} \sum_{i \in \mathcal{C}_3} r^2(i, \mathcal{C}_3) - \frac{1}{n} \sum_{i \in \mathcal{C}_1} r^2(i, \mathcal{C}_1) - \frac{1}{n} \sum_{i \in \mathcal{C}_2} r^2(i, \mathcal{C}_2) \\
&= \Delta\sigma(\mathcal{C}_1, \mathcal{C}_2).
\end{aligned}$$

Ainsi, minimiser $\Delta\sigma(\mathcal{C}_1, \mathcal{C}_2)$ à l'étape k revient à minimiser σ_k.

On va donc établir une formule permettant de calculer efficacement la variation $\Delta\sigma(\mathcal{C}_1, \mathcal{C}_2)$ de σ qui est engendrée si on fusionne $\mathcal{C}_1$ et $\mathcal{C}_2$.

Remarque 2.4.1. Remarquons tout d'abord que si l'on fusionne les communautés $\mathcal{C}_1$ et $\mathcal{C}_2$ en une communauté $\mathcal{C}_3$, alors on a

$$[P^t]_{\mathcal{C}_3 k} = \frac{1}{|\mathcal{C}_3|} \sum_{i \in \mathcal{C}_3} [P^t]_{ik} = \frac{1}{|\mathcal{C}_1| + |\mathcal{C}_2|} \sum_{i \in \mathcal{C}_1 \cup \mathcal{C}_2} [P^t]_{ik} = \frac{1}{|\mathcal{C}_1| + |\mathcal{C}_2|} \Big(\sum_{i \in \mathcal{C}_1} [P^t]_{ik} + \sum_{i \in \mathcal{C}_2} [P^t]_{ik} \Big).$$

Donc

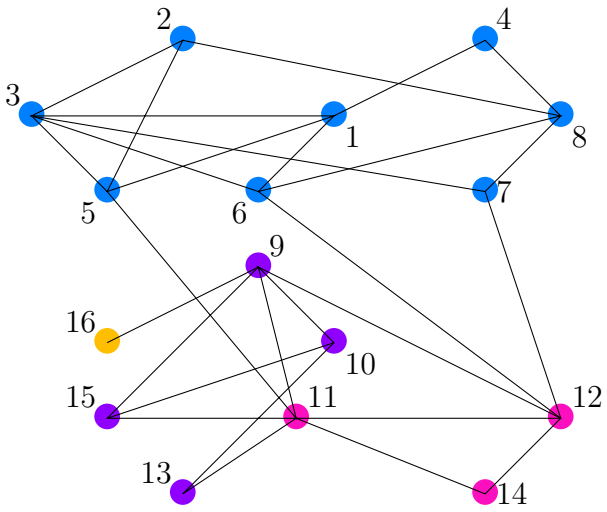
$$(|\mathcal{C}_1| + |\mathcal{C}_2|)[P^t]_{\mathcal{C}_3 k} = |\mathcal{C}_1|[P^t]_{\mathcal{C}_1 k} + |\mathcal{C}_2|[P^t]_{\mathcal{C}_2 k}. \tag{2.4}$$

Cette égalité interviendra à plusieurs reprises dans les démonstrations suivantes. Avant d'obtenir une formule pour calculer $\Delta\sigma$, nous avons besoin du lemme suivant.

Lemme 2.4.2. *Si l'on fusionne les communautés $\mathcal{C}_1$ et $\mathcal{C}_2$ en une communauté $\mathcal{C}_3$, alors on a*

$$\sum_{i \in \mathcal{C}_1} r^2(i, \mathcal{C}_3) = \sum_{i \in \mathcal{C}_1} r^2(i, \mathcal{C}_1) + \frac{|\mathcal{C}_1||\mathcal{C}_2|^2}{(|\mathcal{C}_1| + |\mathcal{C}_2|)^2} r^2(\mathcal{C}_1, \mathcal{C}_2).$$

Exemple 2.4.3. Avant de démontrer ce lemme, illustrons-le sur un exemple. Considérons le graphe suivant :



Graphe 2.2

Pour que le graphe soit plus lisible, les boucles sur chaque sommet n'ont pas été dessinées mais sont bien présentes. Lorsque l'on exécute l'algorithme en prenant $t = 3$, on obtient à la treizième étape la partition suivante :

$$\mathcal{P}_{13} = \{\{1, 2, 3, 4, 5, 6, 7, 8\}, \{9, 10, 13, 15\}, \{11, 12, 14\}, \{16\}\}.$$

Ces étapes seront détaillées dans la section 2.7, mais cette partition nous permettra d'illustrer les lemmes et théorèmes qui vont suivre.
Supposons que $\mathcal{C}_1 = \{9, 10, 13, 15\}$ et $\mathcal{C}_2 = \{11, 12, 14\}$. On a

$$\sum_{i \in \mathcal{C}_1} r^2(i, \mathcal{C}_1) = \sum_{i \in \mathcal{C}_1} \sum_{k=1}^{16} \frac{([P^t]_{ik} - [P^t]_{\mathcal{C}_1 k})^2}{d(k)}$$

et

$$r^2(\mathcal{C}_1, \mathcal{C}_2) = \sum_{k=1}^{16} \frac{([P^t]_{\mathcal{C}_1 k} - [P^t]_{\mathcal{C}_2 k})^2}{d(k)}$$

où

$$[P^t]_{\mathcal{C}_1 k} = \frac{1}{|\mathcal{C}_1|} \sum_{i \in \mathcal{C}_1} [P^t]_{ik}$$

et

$$[P^t]_{\mathcal{C}_2 k} = \frac{1}{|\mathcal{C}_2|} \sum_{i \in \mathcal{C}_2} [P^t]_{ik}.$$

La matrice de transition P à la puissance t est donnée dans l'annexe A. Ainsi, en remplaçant les variables par leurs valeurs, on obtient

$$\begin{aligned}
&\sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_1) + \frac{|\mathcal{C}_1||\mathcal{C}_2|^2}{(|\mathcal{C}_1|+|\mathcal{C}_2|)^2} r^2(\mathcal{C}_1,\mathcal{C}_2) \\
&\qquad = 0.00459459053729148904 + \frac{4.3^2}{(4+3)^2} 0.00944279769443209446 \\
&\qquad \approx 0.01153215619034364016.
\end{aligned}$$

Or

$$\sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_3) = \sum_{i\in\mathcal{C}_1}\sum_{k=1}^{16} \frac{([P^t]_{ik} - [P^t]_{\mathcal{C}_3 k})^2}{d(k)}$$

où

$$[P^t]_{\mathcal{C}_3 k} = \frac{1}{|\mathcal{C}_3|}\sum_{i\in\mathcal{C}_3}[P^t]_{ik} = \frac{1}{|\mathcal{C}_1|+|\mathcal{C}_2|}\Big(\sum_{i\in\mathcal{C}_1}[P^t]_{ik} + \sum_{i\in\mathcal{C}_2}[P^t]_{ik}\Big).$$

Donc

$$\sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_3) \approx 0.01153215619034364016$$

et on a bien l'égalité souhaitée. A présent, démontrons cette équation en toute généralité.

Démonstration. Vu la Définition 2.3.1, on a

$$\sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_3) = \sum_{i\in\mathcal{C}_1}\sum_{k=1}^{n} \frac{([P^t]_{ik} - [P^t]_{\mathcal{C}_3 k})^2}{d(k)}.$$

D'où, vu l'égalité (2.4),

$$\begin{aligned}
\sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_3) &= \sum_{i\in\mathcal{C}_1}\sum_{k=1}^{n} \frac{1}{d(k)}\Big([P^t]_{ik} - \frac{|\mathcal{C}_1|}{|\mathcal{C}_1|+|\mathcal{C}_2|}[P^t]_{\mathcal{C}_1 k} - \frac{|\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}[P^t]_{\mathcal{C}_2 k}\Big)^2 \\
&= \sum_{i\in\mathcal{C}_1}\sum_{k=1}^{n} \frac{1}{d(k)}\Big([P^t]_{ik} - \frac{|\mathcal{C}_1|}{|\mathcal{C}_1|+|\mathcal{C}_2|}[P^t]_{\mathcal{C}_1 k} - \frac{|\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}[P^t]_{\mathcal{C}_1 k} \\
&\qquad + \frac{|\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}[P^t]_{\mathcal{C}_1 k} - \frac{|\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}[P^t]_{\mathcal{C}_2 k}\Big)^2.
\end{aligned}$$

Si l'on regroupe ensemble les trois premiers termes et puis les deux derniers, on obtient

$$\begin{aligned}\sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_3) &= \sum_{i\in\mathcal{C}_1}\sum_{k=1}^{n}\frac{1}{d(k)}\Big(([P^t]_{ik}-[P^t]_{\mathcal{C}_1k})+\frac{|\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}([P^t]_{\mathcal{C}_1k}-[P^t]_{\mathcal{C}_2k})\Big)^2\\ &= \sum_{i\in\mathcal{C}_1}\sum_{k=1}^{n}\frac{1}{d(k)}([P^t]_{ik}-[P^t]_{\mathcal{C}_1k})^2\\ &\quad+\sum_{i\in\mathcal{C}_1}\sum_{k=1}^{n}\frac{1}{d(k)}\frac{|\mathcal{C}_2|^2}{(|\mathcal{C}_1|+|\mathcal{C}_2|)^2}([P^t]_{\mathcal{C}_1k}-[P^t]_{\mathcal{C}_2k})^2\\ &\quad+2\sum_{i\in\mathcal{C}_1}\sum_{k=1}^{n}\frac{1}{d(k)}\frac{|\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}([P^t]_{ik}-[P^t]_{\mathcal{C}_1k})([P^t]_{\mathcal{C}_1k}-[P^t]_{\mathcal{C}_2k})\end{aligned}$$

où la dernière égalité est le développement du produit remarquable.

Regardons ces trois termes séparément. Grâce à la définition de la distance, on a l'égalité suivante pour le premier terme

$$\sum_{i\in\mathcal{C}_1}\sum_{k=1}^{n}\frac{1}{d(k)}([P^t]_{ik}-[P^t]_{\mathcal{C}_1k})^2=\sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_1).$$

Pour le deuxième terme, on peut également appliquer la définition de la distance entre deux communautés et on obtient

$$\begin{aligned}\sum_{i\in\mathcal{C}_1}\sum_{k=1}^{n}\frac{1}{d(k)}\frac{|\mathcal{C}_2|^2}{(|\mathcal{C}_1|+|\mathcal{C}_2|)^2}([P^t]_{\mathcal{C}_1k}-[P^t]_{\mathcal{C}_2k})^2 &= \sum_{i\in\mathcal{C}_1}\frac{|\mathcal{C}_2|^2}{(|\mathcal{C}_1|+|\mathcal{C}_2|)^2}r^2(\mathcal{C}_1,\mathcal{C}_2)\\ = \frac{|\mathcal{C}_2|^2}{(|\mathcal{C}_1|+|\mathcal{C}_2|)^2}r^2(\mathcal{C}_1,\mathcal{C}_2)\sum_{i\in\mathcal{C}_1}1 &= \frac{|\mathcal{C}_2|^2}{(|\mathcal{C}_1|+|\mathcal{C}_2|)^2}r^2(\mathcal{C}_1,\mathcal{C}_2)|\mathcal{C}_1|.\end{aligned}$$

Enfin, si l'on développe le dernier terme, on a

$$\begin{aligned}&\sum_{i\in\mathcal{C}_1}\sum_{k=1}^{n}\frac{1}{d(k)}\frac{|\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}([P^t]_{ik}-[P^t]_{\mathcal{C}_1k})([P^t]_{\mathcal{C}_1k}-[P^t]_{\mathcal{C}_2k})\\ &\quad=\frac{|\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}\sum_{k=1}^{n}\frac{1}{d(k)}([P^t]_{\mathcal{C}_1k}-[P^t]_{\mathcal{C}_2k})\Big(\sum_{i\in\mathcal{C}_1}[P^t]_{ik}-\sum_{i\in\mathcal{C}_1}[P^t]_{\mathcal{C}_1k}\Big).\end{aligned}$$

Or, vu l'égalité (2.1), on a

$$\sum_{i\in\mathcal{C}_1}[P^t]_{ik}=|\mathcal{C}_1|[P^t]_{\mathcal{C}_1k}.$$

De plus,

$$\sum_{i\in\mathcal{C}_1}[P^t]_{\mathcal{C}_1 k} = [P^t]_{\mathcal{C}_1 k}\sum_{i\in\mathcal{C}_1} 1 = |\mathcal{C}_1|[P^t]_{\mathcal{C}_1 k}$$

ainsi,

$$\begin{aligned}\sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_3) &= \sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_1) + |\mathcal{C}_1|\frac{|\mathcal{C}_2|^2}{(|\mathcal{C}_1|+|\mathcal{C}_2|)^2}r^2(\mathcal{C}_1,\mathcal{C}_2)\\ &\quad + \frac{|\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}\sum_{k=1}^{n}\frac{1}{d(k)}([P^t]_{\mathcal{C}_1 k} - [P^t]_{\mathcal{C}_2 k})(|\mathcal{C}_1|[P^t]_{\mathcal{C}_1 k} - |\mathcal{C}_1|[P^t]_{\mathcal{C}_1 k})\\ &= \sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_1) + \frac{|\mathcal{C}_1||\mathcal{C}_2|^2}{(|\mathcal{C}_1|+|\mathcal{C}_2|)^2}r^2(\mathcal{C}_1,\mathcal{C}_2).\end{aligned}$$

□

Théorème 2.4.4. *La variation de σ obtenue en fusionnant les communautés $\mathcal{C}_1$ et $\mathcal{C}_2$ peut être directement calculée grâce à la formule suivante :*

$$\Delta\sigma(\mathcal{C}_1,\mathcal{C}_2) = \frac{1}{n}\frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}r^2(\mathcal{C}_1,\mathcal{C}_2).$$

Exemple 2.4.5. Avant de démontrer ce théorème, reprenons le Graphe 2.2 et illustrons cette égalité. Comme précédemment, lorsque l'on est à l'étape 13 on a la partition $\mathcal{C}_1 = \{1,2,3,4,5,6,7,8\}, \mathcal{C}_2 = \{9,10,13,15\}, \mathcal{C}_3 = \{11,12,14\}$,
$\mathcal{C}_4 = \{16\}$. Rappelons que les valeurs de la matrice de transition à la puissance t se trouvent dans l'annexe A. Regardons les valeurs de $\Delta\sigma$. On a

$$\begin{aligned}\Delta\sigma(\mathcal{C}_1,\mathcal{C}_3) &= \frac{1}{n}\frac{|\mathcal{C}_1||\mathcal{C}_3|}{|\mathcal{C}_1|+|\mathcal{C}_3|}r^2(\mathcal{C}_1,\mathcal{C}_3)\\ &= \frac{1}{16}\frac{24}{11}\sum_{k=1}^{16}\frac{([P^t]_{\mathcal{C}_1 k} - [P^t]_{\mathcal{C}_3 k})^2}{d(k)}\\ &\approx 0.00255387879947437894\end{aligned}$$

et

$$\begin{aligned}\Delta\sigma(\mathcal{C}_2,\mathcal{C}_3) &\approx 0.00101172832440343872\\ \Delta\sigma(\mathcal{C}_2,\mathcal{C}_4) &\approx 0.00101760965575443041.\end{aligned}$$

Remarquons qu'il est inutile de calculer $\Delta\sigma$ pour tous les couples possibles puisque l'on va fusionner uniquement des communautés adjacentes. Notons également que la plus petite valeur obtenue est celle correspondant à la fusion de $\mathcal{C}_2$ et $\mathcal{C}_3$. C'est donc ces deux communautés qui seront choisies pour être fusionnées. A présent, démontrons l'égalité du théorème.

Démonstration. Vu le Lemme 2.4.2 et par symétrie, on a

$$\sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_3) = \sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_1) + \frac{|\mathcal{C}_1||\mathcal{C}_2|^2}{(|\mathcal{C}_1|+|\mathcal{C}_2|)^2} r^2(\mathcal{C}_1,\mathcal{C}_2)$$

et

$$\sum_{i\in\mathcal{C}_2} r^2(i,\mathcal{C}_3) = \sum_{i\in\mathcal{C}_2} r^2(i,\mathcal{C}_2) + \frac{|\mathcal{C}_2||\mathcal{C}_1|^2}{(|\mathcal{C}_1|+|\mathcal{C}_2|)^2} r^2(\mathcal{C}_1,\mathcal{C}_2).$$

On a donc

$$\begin{aligned}\sum_{i\in\mathcal{C}_3=\mathcal{C}_1\cup\mathcal{C}_2} r^2(i,\mathcal{C}_3) &= \sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_3) + \sum_{i\in\mathcal{C}_2} r^2(i,\mathcal{C}_3)\\ &= \sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_1) + \frac{|\mathcal{C}_1||\mathcal{C}_2|^2}{(|\mathcal{C}_1|+|\mathcal{C}_2|)^2} r^2(\mathcal{C}_1,\mathcal{C}_2)\\ &\quad + \sum_{i\in\mathcal{C}_2} r^2(i,\mathcal{C}_2) + \frac{|\mathcal{C}_2||\mathcal{C}_1|^2}{(|\mathcal{C}_1|+|\mathcal{C}_2|)^2} r^2(\mathcal{C}_1,\mathcal{C}_2).\end{aligned}$$

En regroupant le deuxième et le quatrième terme, et en faisant une mise en évidence, on obtient

$$\begin{aligned}\sum_{i\in\mathcal{C}_3=\mathcal{C}_1\cup\mathcal{C}_2} r^2(i,\mathcal{C}_3) &= \sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_1) + \sum_{i\in\mathcal{C}_2} r^2(i,\mathcal{C}_2) + \frac{|\mathcal{C}_1||\mathcal{C}_2|^2}{(|\mathcal{C}_1|+|\mathcal{C}_2|)^2} r^2(\mathcal{C}_1,\mathcal{C}_2)\\ &\quad + \frac{|\mathcal{C}_2||\mathcal{C}_1|^2}{(|\mathcal{C}_1|+|\mathcal{C}_2|)^2} r^2(\mathcal{C}_1,\mathcal{C}_2)\\ &= \sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_1) + \sum_{i\in\mathcal{C}_2} r^2(i,\mathcal{C}_2)\\ &\quad + \frac{|\mathcal{C}_1||\mathcal{C}_2|}{(|\mathcal{C}_1|+|\mathcal{C}_2|)^2} (|\mathcal{C}_1|+|\mathcal{C}_2|) r^2(\mathcal{C}_1,\mathcal{C}_2)\\ &= \sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_1) + \sum_{i\in\mathcal{C}_2} r^2(i,\mathcal{C}_2) + \frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|} r^2(\mathcal{C}_1,\mathcal{C}_2).\end{aligned}$$

Ainsi, en remplaçant dans l'égalité (2.3), on a

$$\begin{aligned}\Delta\sigma(\mathcal{C}_1,\mathcal{C}_2) &= \frac{1}{n}\Big(\sum_{i\in\mathcal{C}_3} r^2(i,\mathcal{C}_3) - \sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_1) - \sum_{i\in\mathcal{C}_2} r^2(i,\mathcal{C}_2)\Big)\\ &= \frac{1}{n}\Big(\sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_1) + \sum_{i\in\mathcal{C}_2} r^2(i,\mathcal{C}_2) + \frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|} r^2(\mathcal{C}_1,\mathcal{C}_2)\\ &\qquad - \sum_{i\in\mathcal{C}_1} r^2(i,\mathcal{C}_1) - \sum_{i\in\mathcal{C}_2} r^2(i,\mathcal{C}_2)\Big)\\ &= \frac{1}{n}\frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|} r^2(\mathcal{C}_1,\mathcal{C}_2).\end{aligned}$$

On obtient bien l'égalité souhaitée.

□

On a donc obtenu une formule qui permet de calculer efficacement la variation $\Delta\sigma(\mathcal{C}_1,\mathcal{C}_2)$ de σ qui est engendrée si on fusionne $\mathcal{C}_1$ et $\mathcal{C}_2$. Cependant, il est inutile de recalculer toutes ces valeurs à chaque étape. En effet, lorsque la valeur a été calculée une fois, par la suite, il suffit de la mettre à jour grâce à une formule dont le calcul s'effectue en temps constant et qui est donc beaucoup plus rapide. C'est cette méthode que nous allons détailler dans la partie suivante.

2.5 Mise à jour des distances entre communautés

Nous allons à présent montrer comment les distances entre communautés peuvent être efficacement modifiées lorsque l'on fusionne deux communautés $\mathcal{C}_1$ et $\mathcal{C}_2$ en une communauté $\mathcal{C}_3$. La formule finale que nous obtenons dans le Théorème 2.5.5 a été créée par Lance, Williams et Jambu et provient de l'article [5]. Pour y arriver, nous allons tout d'abord énoncer et démontrer deux lemmes.

Lemme 2.5.1. *Si on fusionne les communautés $\mathcal{C}_1$ et $\mathcal{C}_2$ en une communauté $\mathcal{C}_3$, alors pour tout autre communauté $\mathcal{C}$ on a*

$$(|\mathcal{C}_1|+|\mathcal{C}_2|)r^2(\mathcal{C}_3,\mathcal{C}) + |\mathcal{C}_1|r^2(\mathcal{C}_1,\mathcal{C}_3) + |\mathcal{C}_2|r^2(\mathcal{C}_2,\mathcal{C}_3) = |\mathcal{C}_1|r^2(\mathcal{C}_1,\mathcal{C}) + |\mathcal{C}_2|r^2(\mathcal{C}_2,\mathcal{C}).$$

Exemple 2.5.2. Reprenons à nouveau notre exemple avec le Graphe 2.2. Pour rappel, à l'étape 13 de notre algorithme, on a obtenu la partition suivante :

$$\mathcal{P}_{13} = \{\{1,2,3,4,5,6,7,8\},\{9,10,13,15\},\{11,12,14\},\{16\}\}.$$

Supposons que l'on fusionne $\mathcal{C}_1 = \{9, 10, 13, 15\}$, et $\mathcal{C}_2 = \{11, 12, 14\}$ en une troisième communauté $\mathcal{C}_3 = \{9, 10, 11, 12, 13, 14, 15\}$. On a, vu les valeurs de la matrice de transition à l'annexe A,

$$[P^t]_{\mathcal{C}_1 k} = \frac{1}{|\mathcal{C}_1|} \sum_{i \in \mathcal{C}_1} [P^t]_{ik}, \quad [P^t]_{\mathcal{C}_3 k} = \frac{1}{|\mathcal{C}_3|} \sum_{i \in \mathcal{C}_3} [P^t]_{ik}$$

$$r^2(\mathcal{C}_1, \mathcal{C}_3) = \sum_{k=1}^{16} \frac{([P^t]_{\mathcal{C}_1 k} - [P^t]_{\mathcal{C}_3 k})^2}{d(k)} \approx 0.00173439141326303821.$$

De même, si on considère $\mathcal{C} = \{16\}$, on peut calculer

$$\begin{aligned} r^2(\mathcal{C}_3, \mathcal{C}) &\approx 0.02249184330914641428 \\ r^2(\mathcal{C}_2, \mathcal{C}_3) &\approx 0.00308336251246762267 \\ r^2(\mathcal{C}_1, \mathcal{C}) &\approx 0.02035219311508859943 \\ r^2(\mathcal{C}_2, \mathcal{C}) &\approx 0.03074059463137516698. \end{aligned}$$

On a, d'une part,

$$\begin{aligned} &(|\mathcal{C}_1| + |\mathcal{C}_2|) r^2(\mathcal{C}_3, \mathcal{C}) + |\mathcal{C}_1| r^2(\mathcal{C}_1, \mathcal{C}_3) + |\mathcal{C}_2| r^2(\mathcal{C}_2, \mathcal{C}_3) \\ &\quad \approx (4+3) * 0.02249184330914641428 + 4 * 0.00173439141326303821 \\ &\qquad + 3 * 0.00308336251246762267 \\ &\quad \approx 0.17363055635447990910 \end{aligned}$$

d'autre part,

$$\begin{aligned} &|\mathcal{C}_1| r^2(\mathcal{C}_1, \mathcal{C}) + |\mathcal{C}_2| r^2(\mathcal{C}_2, \mathcal{C}) \\ &\quad \approx 4 * 0.02035219311508859943 + 3 * 0.03074059463137516698 \\ &\quad \approx 0.17363055635447988134 \end{aligned}$$

et, à un arrondi près, on a bien l'égalité des deux membres. Montrons ce lemme en toute généralité.

Démonstration. En utilisant la Définition 2.3.2 de distance entre deux communautés,

on a,

$$\begin{aligned}
&(|\mathcal{C}_1|+|\mathcal{C}_2|)r^2(\mathcal{C}_3,\mathcal{C})+|\mathcal{C}_1|r^2(\mathcal{C}_1,\mathcal{C}_3)+|\mathcal{C}_2|r^2(\mathcal{C}_2,\mathcal{C}_3)\\
&\quad=(|\mathcal{C}_1|+|\mathcal{C}_2|)\sum_{k=1}^{n}\frac{1}{d(k)}([P^t]_{\mathcal{C}_3k}-[P^t]_{\mathcal{C}k})^2+|\mathcal{C}_1|\sum_{k=1}^{n}\frac{1}{d(k)}([P^t]_{\mathcal{C}_1k}-[P^t]_{\mathcal{C}_3k})^2\\
&\qquad+|\mathcal{C}_2|\sum_{k=1}^{n}\frac{1}{d(k)}([P^t]_{\mathcal{C}_2k}-[P^t]_{\mathcal{C}_3k})^2.
\end{aligned}$$

On peut à présent faire une mise en évidence,

$$\begin{aligned}
&(|\mathcal{C}_1|+|\mathcal{C}_2|)r^2(\mathcal{C}_3,\mathcal{C})+|\mathcal{C}_1|r^2(\mathcal{C}_1,\mathcal{C}_3)+|\mathcal{C}_2|r^2(\mathcal{C}_2,\mathcal{C}_3)\\
&\quad=\sum_{k=1}^{n}\frac{1}{d(k)}\Big[(|\mathcal{C}_1|+|\mathcal{C}_2|)([P^t]_{\mathcal{C}_3k}-[P^t]_{\mathcal{C}k})^2+|\mathcal{C}_1|([P^t]_{\mathcal{C}_1k}-[P^t]_{\mathcal{C}_3k})^2\\
&\qquad+|\mathcal{C}_2|([P^t]_{\mathcal{C}_2k}-[P^t]_{\mathcal{C}_3k})^2\Big].
\end{aligned}$$

Notons l'expression entre crochets I et développons-la. Si l'on effectue les trois produits remarquables, on obtient

$$\begin{aligned}
I&=(|\mathcal{C}_1|+|\mathcal{C}_2|)\Big(([P^t]_{\mathcal{C}_3k})^2+([P^t]_{\mathcal{C}k})^2-2[P^t]_{\mathcal{C}_3k}[P^t]_{\mathcal{C}k}\Big)\\
&\quad+|\mathcal{C}_1|\Big(([P^t]_{\mathcal{C}_1k})^2+([P^t]_{\mathcal{C}_3k})^2-2[P^t]_{\mathcal{C}_1k}[P^t]_{\mathcal{C}_3k}\Big)\\
&\quad+|\mathcal{C}_2|\Big(([P^t]_{\mathcal{C}_2k})^2+([P^t]_{\mathcal{C}_3k})^2-2[P^t]_{\mathcal{C}_2k}[P^t]_{\mathcal{C}_3k}\Big)\\
&=|\mathcal{C}_1|([P^t]_{\mathcal{C}_3k})^2+|\mathcal{C}_1|([P^t]_{\mathcal{C}k})^2-2|\mathcal{C}_1|[P^t]_{\mathcal{C}_3k}[P^t]_{\mathcal{C}k}\\
&\quad+|\mathcal{C}_2|([P^t]_{\mathcal{C}_3k})^2+|\mathcal{C}_2|([P^t]_{\mathcal{C}k})^2-2|\mathcal{C}_2|[P^t]_{\mathcal{C}_3k}[P^t]_{\mathcal{C}k}\\
&\quad+|\mathcal{C}_1|([P^t]_{\mathcal{C}_1k})^2+|\mathcal{C}_1|([P^t]_{\mathcal{C}_3k})^2-2|\mathcal{C}_1|[P^t]_{\mathcal{C}_1k}[P^t]_{\mathcal{C}_3k}\\
&\quad+|\mathcal{C}_2|([P^t]_{\mathcal{C}_2k})^2+|\mathcal{C}_2|([P^t]_{\mathcal{C}_3k})^2-2|\mathcal{C}_2|[P^t]_{\mathcal{C}_2k}[P^t]_{\mathcal{C}_3k}.
\end{aligned}$$

Dès lors, on peut réorganiser les termes de la façon suivante :

$$\begin{aligned}
I &= |\mathcal{C}_1|([P^t]_{\mathcal{C}_3 k})^2 + |\mathcal{C}_2|([P^t]_{\mathcal{C}_3 k})^2 + |\mathcal{C}_1|([P^t]_{\mathcal{C}_3 k})^2 + |\mathcal{C}_2|([P^t]_{\mathcal{C}_3 k})^2 \\
&\quad + |\mathcal{C}_1|([P^t]_{\mathcal{C}k})^2 + |\mathcal{C}_1|([P^t]_{\mathcal{C}_1 k})^2 + |\mathcal{C}_2|([P^t]_{\mathcal{C}k})^2 + |\mathcal{C}_2|([P^t]_{\mathcal{C}_2 k})^2 \\
&\quad - 2|\mathcal{C}_1|[P^t]_{\mathcal{C}_1 k}[P^t]_{\mathcal{C}_3 k} - 2|\mathcal{C}_2|[P^t]_{\mathcal{C}_2 k}[P^t]_{\mathcal{C}_3 k} - 2|\mathcal{C}_1|[P^t]_{\mathcal{C}_3 k}[P^t]_{\mathcal{C}k} \\
&\quad - 2|\mathcal{C}_2|[P^t]_{\mathcal{C}_3 k}[P^t]_{\mathcal{C}k} \\
&= 2(|\mathcal{C}_1| + |\mathcal{C}_2|)([P^t]_{\mathcal{C}_3 k})^2 + |\mathcal{C}_1|\big[([P^t]_{\mathcal{C}k})^2 + ([P^t]_{\mathcal{C}_1 k})^2\big] \\
&\quad + |\mathcal{C}_2|\big[([P^t]_{\mathcal{C}k})^2 + ([P^t]_{\mathcal{C}_2 k})^2\big] - 2[P^t]_{\mathcal{C}_3 k}(|\mathcal{C}_1|[P^t]_{\mathcal{C}_1 k} + |\mathcal{C}_2|[P^t]_{\mathcal{C}_2 k}) \\
&\quad - 2(|\mathcal{C}_1| + |\mathcal{C}_2|)[P^t]_{\mathcal{C}_3 k}[P^t]_{\mathcal{C}k}.
\end{aligned}$$

Or, vu l'égalité (2.4), on a

$$|\mathcal{C}_1|[P^t]_{\mathcal{C}_1 k} + |\mathcal{C}_2|[P^t]_{\mathcal{C}_2 k} = (|\mathcal{C}_1| + |\mathcal{C}_2|)[P^t]_{\mathcal{C}_3 k}$$

donc le premier et le quatrième terme vont se simplifier. De plus, par l'égalité (2.4), le dernier terme peut être transformé de la façon suivante :

$$(|\mathcal{C}_1| + |\mathcal{C}_2|)[P^t]_{\mathcal{C}_3 k}[P^t]_{\mathcal{C}k} = (|\mathcal{C}_1|[P^t]_{\mathcal{C}_1 k} + |\mathcal{C}_2|[P^t]_{\mathcal{C}_2 k})[P^t]_{\mathcal{C}k}.$$

Ainsi,

$$\begin{aligned}
I &= |\mathcal{C}_1|\big[([P^t]_{\mathcal{C}k})^2 + ([P^t]_{\mathcal{C}_1 k})^2\big] + |\mathcal{C}_2|\big[([P^t]_{\mathcal{C}k})^2 + ([P^t]_{\mathcal{C}_2 k})^2\big] \\
&\quad - 2(|\mathcal{C}_1|[P^t]_{\mathcal{C}_1 k} + |\mathcal{C}_2|[P^t]_{\mathcal{C}_2 k})[P^t]_{\mathcal{C}k}.
\end{aligned}$$

On a donc

$$\begin{aligned}
&(|\mathcal{C}_1| + |\mathcal{C}_2|)r^2(\mathcal{C}_3, \mathcal{C}) + |\mathcal{C}_1|r^2(\mathcal{C}_1, \mathcal{C}_3) + |\mathcal{C}_2|r^2(\mathcal{C}_2, \mathcal{C}_3) \\
&\quad = \sum_{k=1}^{n} \frac{1}{d(k)} I \\
&\quad = \sum_{k=1}^{n} \frac{1}{d(k)} \Big[|\mathcal{C}_1|\big(([P^t]_{\mathcal{C}k})^2 + ([P^t]_{\mathcal{C}_1 k})^2\big) - 2|\mathcal{C}_1|[P^t]_{\mathcal{C}_1 k}[P^t]_{\mathcal{C}k}\Big] \\
&\qquad + \sum_{k=1}^{n} \frac{1}{d(k)} \Big[|\mathcal{C}_2|\big(([P^t]_{\mathcal{C}k})^2 + ([P^t]_{\mathcal{C}_2 k})^2\big) - 2|\mathcal{C}_2|[P^t]_{\mathcal{C}_2 k}[P^t]_{\mathcal{C}k}\Big].
\end{aligned}$$

Finalement, en utilisant les produits remarquables et la Définition 2.3.2 de distance entre deux communautés, on obtient

$$\begin{aligned}
&(|\mathcal{C}_1| + |\mathcal{C}_2|)r^2(\mathcal{C}_3, \mathcal{C}) + |\mathcal{C}_1|r^2(\mathcal{C}_1, \mathcal{C}_3) + |\mathcal{C}_2|r^2(\mathcal{C}_2, \mathcal{C}_3) \\
&\quad = |\mathcal{C}_1| \sum_{k=1}^{n} \frac{1}{d(k)}([P^t]_{\mathcal{C}_1 k} - [P^t]_{\mathcal{C}k})^2 + |\mathcal{C}_2| \sum_{k=1}^{n} \frac{1}{d(k)}([P^t]_{\mathcal{C}_2 k} - [P^t]_{\mathcal{C}k})^2 \\
&\quad = |\mathcal{C}_1|r^2(\mathcal{C}_1, \mathcal{C}) + |\mathcal{C}_2|r^2(\mathcal{C}_2, \mathcal{C}).
\end{aligned}$$

□

Lemme 2.5.3. *Si on fusionne les communautés $\mathcal{C}_1$ et $\mathcal{C}_2$ en une communauté $\mathcal{C}_3$, alors on a*

$$|\mathcal{C}_1|r^2(\mathcal{C}_1, \mathcal{C}_3) + |\mathcal{C}_2|r^2(\mathcal{C}_2, \mathcal{C}_3) = \frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1| + |\mathcal{C}_2|} r^2(\mathcal{C}_1, \mathcal{C}_2).$$

Exemple 2.5.4. Illustrons ce lemme sur notre exemple avec le Graphe 2.2 . Les distances dont on a besoin ont déjà été calculées dans l'Exemple 2.4.3, et dans l'Exemple 2.5.2. Les valeurs que l'on avait obtenues sont les suivantes :

$$\begin{aligned}
r^2(\mathcal{C}_1, \mathcal{C}_2) &\approx 0.00944279769443209446 \\
r^2(\mathcal{C}_1, \mathcal{C}_3) &\approx 0.00173439141326303821 \\
r^2(\mathcal{C}_2, \mathcal{C}_3) &\approx 0.00308336251246762267.
\end{aligned}$$

Vérifions l'égalité du lemme. Pour le membre de gauche, on a

$$\begin{aligned}
&|\mathcal{C}_1|r^2(\mathcal{C}_1, \mathcal{C}_3) + |\mathcal{C}_2|r^2(\mathcal{C}_2, \mathcal{C}_3) \\
&\quad \approx 4 * 0.00173439141326303821 + 3 * 0.00308336251246762267 \\
&\quad \approx 0.01618765319045501958
\end{aligned}$$

et pour le membre de droite on a

$$\begin{aligned}
\frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1| + |\mathcal{C}_2|} r^2(\mathcal{C}_1, \mathcal{C}_2) &\approx \frac{12}{7} 0.00944279769443209446 \\
&\approx 0.01618765319045501958.
\end{aligned}$$

Les deux membres sont donc bien identiques.

Démonstration. En utilisant la Définition 2.3.2 de distance entre deux communautés, on a,

$$\begin{aligned}
&|\mathcal{C}_1|r^2(\mathcal{C}_1,\mathcal{C}_3)+|\mathcal{C}_2|r^2(\mathcal{C}_2,\mathcal{C}_3)\\
&\quad=|\mathcal{C}_1|\sum_{k=1}^{n}\frac{1}{d(k)}([P^t]_{\mathcal{C}_1k}-[P^t]_{\mathcal{C}_3k})^2+|\mathcal{C}_2|\sum_{k=1}^{n}\frac{1}{d(k)}([P^t]_{\mathcal{C}_2k}-[P^t]_{\mathcal{C}_3k})^2.
\end{aligned}$$

En développant les produits remarquables, on obtient

$$\begin{aligned}
&|\mathcal{C}_1|r^2(\mathcal{C}_1,\mathcal{C}_3)+|\mathcal{C}_2|r^2(\mathcal{C}_2,\mathcal{C}_3)\\
&\quad=\sum_{k=1}^{n}\frac{1}{d(k)}\Big[|\mathcal{C}_1|([P^t]_{\mathcal{C}_1k})^2+|\mathcal{C}_1|([P^t]_{\mathcal{C}_3k})^2-2|\mathcal{C}_1|[P^t]_{\mathcal{C}_1k}[P^t]_{\mathcal{C}_3k}\\
&\qquad+|\mathcal{C}_2|([P^t]_{\mathcal{C}_2k})^2+|\mathcal{C}_2|([P^t]_{\mathcal{C}_3k})^2-2|\mathcal{C}_2|[P^t]_{\mathcal{C}_2k}[P^t]_{\mathcal{C}_3k}\Big]\\
&\quad=\sum_{k=1}^{n}\frac{1}{d(k)}\Big[|\mathcal{C}_1|([P^t]_{\mathcal{C}_1k})^2+|\mathcal{C}_2|([P^t]_{\mathcal{C}_2k})^2+(|\mathcal{C}_1|+|\mathcal{C}_2|)([P^t]_{\mathcal{C}_3k})^2\\
&\qquad-2[P^t]_{\mathcal{C}_3k}(|\mathcal{C}_1|[P^t]_{\mathcal{C}_1k}+|\mathcal{C}_2|[P^t]_{\mathcal{C}_2k})\Big].
\end{aligned}$$

Or, vu l'égalité (2.4), on a

$$|\mathcal{C}_1|[P^t]_{\mathcal{C}_1k}+|\mathcal{C}_2|[P^t]_{\mathcal{C}_2k}=(|\mathcal{C}_1|+|\mathcal{C}_2|)[P^t]_{\mathcal{C}_3k}.$$

D'où,

$$\begin{aligned}
&|\mathcal{C}_1|r^2(\mathcal{C}_1,\mathcal{C}_3)+|\mathcal{C}_2|r^2(\mathcal{C}_2,\mathcal{C}_3)\\
&\quad=\sum_{k=1}^{n}\frac{1}{d(k)}\Big[|\mathcal{C}_1|([P^t]_{\mathcal{C}_1k})^2+|\mathcal{C}_2|([P^t]_{\mathcal{C}_2k})^2+(|\mathcal{C}_1|+|\mathcal{C}_2|)([P^t]_{\mathcal{C}_3k})^2\\
&\qquad-2[P^t]_{\mathcal{C}_3k}(|\mathcal{C}_1|+|\mathcal{C}_2|)[P^t]_{\mathcal{C}_3k}\Big]\\
&\quad=\sum_{k=1}^{n}\frac{1}{d(k)}\Big[|\mathcal{C}_1|([P^t]_{\mathcal{C}_1k})^2+|\mathcal{C}_2|([P^t]_{\mathcal{C}_2k})^2-(|\mathcal{C}_1|+|\mathcal{C}_2|)([P^t]_{\mathcal{C}_3k})^2\Big].
\end{aligned}$$

Or, l'égalité (2.4) peut se réécrire de la façon suivante :

$$[P^t]_{\mathcal{C}_3k}=\frac{1}{|\mathcal{C}_1|+|\mathcal{C}_2|}(|\mathcal{C}_1|[P^t]_{\mathcal{C}_1k}+|\mathcal{C}_2|[P^t]_{\mathcal{C}_2k}).$$

En élevant les deux membres au carré, on a

$$([P^t]_{\mathcal{C}_3 k})^2 = \frac{1}{(|\mathcal{C}_1|+|\mathcal{C}_2|)^2}\Big(|\mathcal{C}_1|^2([P^t]_{\mathcal{C}_1 k})^2 + |\mathcal{C}_2|^2([P^t]_{\mathcal{C}_2 k})^2 + 2|\mathcal{C}_1||\mathcal{C}_2|[P^t]_{\mathcal{C}_1 k}[P^t]_{\mathcal{C}_2 k}\Big).$$

Ainsi,

$$\begin{aligned}
&|\mathcal{C}_1|r^2(\mathcal{C}_1,\mathcal{C}_3) + |\mathcal{C}_2|r^2(\mathcal{C}_2,\mathcal{C}_3)\\
&\quad= \sum_{k=1}^{n}\frac{1}{d(k)}\Big[|\mathcal{C}_1|([P^t]_{\mathcal{C}_1 k})^2 + |\mathcal{C}_2|([P^t]_{\mathcal{C}_2 k})^2\\
&\qquad - \frac{(|\mathcal{C}_1|+|\mathcal{C}_2|)}{(|\mathcal{C}_1|+|\mathcal{C}_2|)^2}\Big(|\mathcal{C}_1|^2([P^t]_{\mathcal{C}_1 k})^2 + |\mathcal{C}_2|^2([P^t]_{\mathcal{C}_2 k})^2 + 2|\mathcal{C}_1||\mathcal{C}_2|[P^t]_{\mathcal{C}_1 k}[P^t]_{\mathcal{C}_2 k}\Big)\Big]\\
&\quad= \sum_{k=1}^{n}\frac{1}{d(k)}\Big[|\mathcal{C}_1|([P^t]_{\mathcal{C}_1 k})^2 + |\mathcal{C}_2|([P^t]_{\mathcal{C}_2 k})^2 - \frac{1}{|\mathcal{C}_1|+|\mathcal{C}_2|}|\mathcal{C}_1|^2([P^t]_{\mathcal{C}_1 k})^2\\
&\qquad - \frac{1}{|\mathcal{C}_1|+|\mathcal{C}_2|}|\mathcal{C}_2|^2([P^t]_{\mathcal{C}_2 k})^2 - 2\frac{1}{|\mathcal{C}_1|+|\mathcal{C}_2|}|\mathcal{C}_1||\mathcal{C}_2|[P^t]_{\mathcal{C}_1 k}[P^t]_{\mathcal{C}_2 k}\Big].
\end{aligned}$$

A présent, on peut regrouper le premier et le troisième terme ainsi que le deuxième et le quatrième terme. On a donc

$$\begin{aligned}
&|\mathcal{C}_1|r^2(\mathcal{C}_1,\mathcal{C}_3) + |\mathcal{C}_2|r^2(\mathcal{C}_2,\mathcal{C}_3)\\
&\quad= \sum_{k=1}^{n}\frac{1}{d(k)}\Big[([P^t]_{\mathcal{C}_1 k})^2\Big(|\mathcal{C}_1| - \frac{|\mathcal{C}_1|^2}{|\mathcal{C}_1|+|\mathcal{C}_2|}\Big) + ([P^t]_{\mathcal{C}_2 k})^2\Big(|\mathcal{C}_2| - \frac{|\mathcal{C}_2|^2}{|\mathcal{C}_1|+|\mathcal{C}_2|}\Big)\\
&\qquad - 2\frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}[P^t]_{\mathcal{C}_1 k}[P^t]_{\mathcal{C}_2 k}\Big]\\
&\quad= \sum_{k=1}^{n}\frac{1}{d(k)}\Big[([P^t]_{\mathcal{C}_1 k})^2\Big(\frac{|\mathcal{C}_1|(|\mathcal{C}_1|+|\mathcal{C}_2|) - |\mathcal{C}_1|^2}{|\mathcal{C}_1|+|\mathcal{C}_2|}\Big)\\
&\qquad + ([P^t]_{\mathcal{C}_2 k})^2\Big(\frac{|\mathcal{C}_2|(|\mathcal{C}_1|+|\mathcal{C}_2|) - |\mathcal{C}_2|^2}{|\mathcal{C}_1|+|\mathcal{C}_2|}\Big) - 2\frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}[P^t]_{\mathcal{C}_1 k}[P^t]_{\mathcal{C}_2 k}\Big]\\
&\quad= \sum_{k=1}^{n}\frac{1}{d(k)}\Big[([P^t]_{\mathcal{C}_1 k})^2\Big(\frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}\Big) + ([P^t]_{\mathcal{C}_2 k})^2\Big(\frac{|\mathcal{C}_2||\mathcal{C}_1|}{|\mathcal{C}_1|+|\mathcal{C}_2|}\Big)\\
&\qquad - 2\frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}[P^t]_{\mathcal{C}_1 k}[P^t]_{\mathcal{C}_2 k}\Big].
\end{aligned}$$

Finalement, en faisant une simple mise en évidence et en appliquant la Définition 2.3.2

de distance entre deux communautés, on obtient

$$\begin{aligned}
&|\mathcal{C}_1|r^2(\mathcal{C}_1,\mathcal{C}_3)+|\mathcal{C}_2|r^2(\mathcal{C}_2,\mathcal{C}_3)\\
&\quad=\sum_{k=1}^{n}\frac{1}{d(k)}\Big[\frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}\Big(([P^t]_{\mathcal{C}_1k})^2+([P^t]_{\mathcal{C}_2k})^2-2[P^t]_{\mathcal{C}_1k}[P^t]_{\mathcal{C}_2k}\Big)\Big]\\
&\quad=\frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}\sum_{k=1}^{n}\frac{1}{d(k)}([P^t]_{\mathcal{C}_1k}-[P^t]_{\mathcal{C}_2k})^2\\
&\quad=\frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}r^2(\mathcal{C}_1,\mathcal{C}_2).
\end{aligned}$$

□

Grâce à ces deux lemmes, on va pouvoir obtenir une formule qui permet, lorsque l'on fusionne les communautés $\mathcal{C}_1$ et $\mathcal{C}_2$ en une communauté $\mathcal{C}_3$, de mettre à jour les valeurs de $\Delta\sigma$ entre $\mathcal{C}_3$ et toutes ses communautés adjacentes $\mathcal{C}$. Remarquons que si $\Delta\sigma(\mathcal{C}_1,\mathcal{C})$ et $\Delta\sigma(\mathcal{C}_2,\mathcal{C})$ sont connus, alors le calcul se fait en temps constant. Sinon, il suffit de calculer ces valeurs grâce à la formule donnée dans le Théorème 2.4.4. Ce calcul peut se faire en $\mathcal{O}(n)$ opérations.

Théorème 2.5.5. *Si on fusionne les communautés $\mathcal{C}_1$ et $\mathcal{C}_2$ en une communauté $\mathcal{C}_3$, alors pour tout autre communauté $\mathcal{C}$, on a*

$$\Delta\sigma(\mathcal{C}_3,\mathcal{C})=\frac{(|\mathcal{C}_1|+|\mathcal{C}|)\Delta\sigma(\mathcal{C}_1,\mathcal{C})+(|\mathcal{C}_2|+|\mathcal{C}|)\Delta\sigma(\mathcal{C}_2,\mathcal{C})-|\mathcal{C}|\Delta\sigma(\mathcal{C}_1,\mathcal{C}_2)}{|\mathcal{C}_1|+|\mathcal{C}_2|+|\mathcal{C}|}.$$

Exemple 2.5.6. Reprenons notre exemple avec le Graphe 2.2 et supposons que les valeurs suivantes ont déjà été calculées :

$$\begin{aligned}
\Delta\sigma(\mathcal{C}_1,\mathcal{C})&\approx 0.02035219311508859943\\
\Delta\sigma(\mathcal{C}_2,\mathcal{C})&\approx 0.03074059463137516698\\
\Delta\sigma(\mathcal{C}_1,\mathcal{C}_2)&\approx 0.00944279769443209446
\end{aligned}$$

où $\mathcal{C}_1=\{9,10,13,15\}$, $\mathcal{C}_2=\{11,12,14\}$ et $\mathcal{C}=\{16\}$. Par le Théorème 2.5.5, on a

$$\begin{aligned}
\Delta\sigma(\mathcal{C}_3,\mathcal{C})&\approx\frac{1}{4+3+1}\Big((4+1)0.02035219311508859943\\
&\qquad+(3+1)0.03074059463137516698-0.00944279769443209446\Big)\\
&\approx 0.00123002268096894431.
\end{aligned}$$

Or, si on calcule $\Delta\sigma(\mathcal{C}_3, \mathcal{C})$ en utilisant le Théorème 2.4.4, on obtient

$$\begin{aligned}\Delta\sigma(\mathcal{C}_3, \mathcal{C}) &= \frac{1}{n}\frac{|\mathcal{C}_3||\mathcal{C}|}{|\mathcal{C}_3| + |\mathcal{C}|} r^2(\mathcal{C}_3, \mathcal{C}) \approx \frac{1}{16}\frac{7}{8} 0.02249184330914641428 \\ &\approx 0.00123002268096894453\end{aligned}$$

ce qui donne bien la même approximation dans les deux cas. Démontrons cette formule en toute généralité.

Démonstration. Regardons tout d'abord les membres de gauche et de droite de la thèse séparément. Vu le Théorème 2.4.4, d'une part, on a

$$\Delta\sigma(\mathcal{C}_3, \mathcal{C}) = \frac{1}{n}\frac{|\mathcal{C}_3||\mathcal{C}|}{|\mathcal{C}_3| + |\mathcal{C}|} r^2(\mathcal{C}_3, \mathcal{C})$$

d'autre part, on a

$$\begin{aligned}&\frac{(|\mathcal{C}_1| + |\mathcal{C}|)\Delta\sigma(\mathcal{C}_1, \mathcal{C}) + (|\mathcal{C}_2| + |\mathcal{C}|)\Delta\sigma(\mathcal{C}_2, \mathcal{C}) - |\mathcal{C}|\Delta\sigma(\mathcal{C}_1, \mathcal{C}_2)}{|\mathcal{C}_1| + |\mathcal{C}_2| + |\mathcal{C}|} \\ &= \frac{1}{|\mathcal{C}_1| + |\mathcal{C}_2| + |\mathcal{C}|}\left[(|\mathcal{C}_1| + |\mathcal{C}|)\frac{1}{n}\frac{|\mathcal{C}_1||\mathcal{C}|}{|\mathcal{C}_1| + |\mathcal{C}|} r^2(\mathcal{C}_1, \mathcal{C})\right. \\ &\quad \left. + (|\mathcal{C}_2| + |\mathcal{C}|)\frac{1}{n}\frac{|\mathcal{C}_2||\mathcal{C}|}{|\mathcal{C}_2| + |\mathcal{C}|} r^2(\mathcal{C}_2, \mathcal{C}) - |\mathcal{C}|\frac{1}{n}\frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1| + |\mathcal{C}_2|} r^2(\mathcal{C}_1, \mathcal{C}_2)\right] \\ &= \frac{|\mathcal{C}|}{n(|\mathcal{C}_1| + |\mathcal{C}_2| + |\mathcal{C}|)}\left[(|\mathcal{C}_1| + |\mathcal{C}|)\frac{|\mathcal{C}_1|}{|\mathcal{C}_1| + |\mathcal{C}|} r^2(\mathcal{C}_1, \mathcal{C})\right. \\ &\quad \left. + (|\mathcal{C}_2| + |\mathcal{C}|)\frac{|\mathcal{C}_2|}{|\mathcal{C}_2| + |\mathcal{C}|} r^2(\mathcal{C}_2, \mathcal{C}) - \frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1| + |\mathcal{C}_2|} r^2(\mathcal{C}_1, \mathcal{C}_2)\right].\end{aligned}$$

Si l'on multiplie ces deux équations par $\frac{n(|\mathcal{C}_1|+|\mathcal{C}_2|+|\mathcal{C}|)}{|\mathcal{C}|}$ on obtient d'une part,

$$\Delta\sigma(\mathcal{C}_3, \mathcal{C})\left[\frac{n(|\mathcal{C}_1| + |\mathcal{C}_2| + |\mathcal{C}|)}{|\mathcal{C}|}\right] = |\mathcal{C}_3| r^2(\mathcal{C}_3, \mathcal{C})$$

vu que $|\mathcal{C}_1| + |\mathcal{C}_2| = |\mathcal{C}_3|$. D'autre part, on a

$$\begin{aligned}&\left[\frac{(|\mathcal{C}_1| + |\mathcal{C}|)\Delta\sigma(\mathcal{C}_1, \mathcal{C}) + (|\mathcal{C}_2| + |\mathcal{C}|)\Delta\sigma(\mathcal{C}_2, \mathcal{C}) - |\mathcal{C}|\Delta\sigma(\mathcal{C}_1, \mathcal{C}_2)}{|\mathcal{C}_1| + |\mathcal{C}_2| + |\mathcal{C}|}\right]\left[\frac{n(|\mathcal{C}_1| + |\mathcal{C}_2| + |\mathcal{C}|)}{|\mathcal{C}|}\right] \\ &= (|\mathcal{C}_1| + |\mathcal{C}|)\frac{|\mathcal{C}_1|}{|\mathcal{C}_1| + |\mathcal{C}|} r^2(\mathcal{C}_1, \mathcal{C}) + (|\mathcal{C}_2| + |\mathcal{C}|)\frac{|\mathcal{C}_2|}{|\mathcal{C}_2| + |\mathcal{C}|} r^2_{\mathcal{C}_2\mathcal{C}} - \frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1| + |\mathcal{C}_2|} r^2(\mathcal{C}_1, \mathcal{C}_2) \\ &= |\mathcal{C}_1| r^2(\mathcal{C}_1, \mathcal{C}) + |\mathcal{C}_2| r^2(\mathcal{C}_2, \mathcal{C}) - \frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1| + |\mathcal{C}_2|} r^2(\mathcal{C}_1, \mathcal{C}_2).\end{aligned}$$

La thèse est donc équivalente à

$$|\mathcal{C}_3|r^2(\mathcal{C}_3,\mathcal{C}) = |\mathcal{C}_1|r^2(\mathcal{C}_1,\mathcal{C}) + |\mathcal{C}_2|r^2(\mathcal{C}_2,\mathcal{C}) - \frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}r^2(\mathcal{C}_1,\mathcal{C}_2). \tag{2.5}$$

Vu le Lemme 2.5.1, on a

$$|\mathcal{C}_1|r^2(\mathcal{C}_1,\mathcal{C})+|\mathcal{C}_2|r^2(\mathcal{C}_2,\mathcal{C}) = (|\mathcal{C}_1|+|\mathcal{C}_2|)r^2(\mathcal{C}_3,\mathcal{C})+|\mathcal{C}_1|r^2(\mathcal{C}_1,\mathcal{C}_3)+|\mathcal{C}_2|r^2(\mathcal{C}_2,\mathcal{C}_3). \tag{2.6}$$

De plus, par le Lemme 2.5.3, on a

$$|\mathcal{C}_1|r^2(\mathcal{C}_1,\mathcal{C}_3) + |\mathcal{C}_2|r^2(\mathcal{C}_2,\mathcal{C}_3) = \frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}r^2(\mathcal{C}_1,\mathcal{C}_2). \tag{2.7}$$

D'où, en remplaçant (2.7) dans (2.6), on obtient

$$|\mathcal{C}_1|r^2(\mathcal{C}_1,\mathcal{C}) + |\mathcal{C}_2|r^2(\mathcal{C}_2,\mathcal{C}) = (|\mathcal{C}_1| + |\mathcal{C}_2|)r^2(\mathcal{C}_3,\mathcal{C}) + \frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}r^2(\mathcal{C}_1,\mathcal{C}_2).$$

Ainsi, vu que $|\mathcal{C}_3| = |\mathcal{C}_1| + |\mathcal{C}_2|$, on obtient

$$|\mathcal{C}_1|r^2(\mathcal{C}_1,\mathcal{C}) + |\mathcal{C}_2|r^2(\mathcal{C}_2,\mathcal{C}) = |\mathcal{C}_3|r^2(\mathcal{C}_3,\mathcal{C}) + \frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1|+|\mathcal{C}_2|}r^2(\mathcal{C}_1,\mathcal{C}_2)$$

qui est bien l'égalité (2.5) souhaitée.

□

2.6 Qualité d'une partition

Notre algorithme nous donne donc une suite $(\mathcal{P}_k)_{1\leq k\leq n}$ de partitions du graphe en communautés. On voudrait à présent savoir laquelle de ces partitions représente au mieux la séparation des communautés. Dans la section2.4, on a introduit la définition de la variation $\Delta\sigma(\mathcal{C}_1,\mathcal{C}_2)$ de σ qui est engendrée si on fusionne les communautés $\mathcal{C}_1$ et $\mathcal{C}_2$. Cette valeur sera « petite » si on fusionne deux communautés « proches » et au contraire elle sera « grande » si la distance entre les deux communautés que l'on fusionne est « grande ». Ainsi, à l'étape k, la valeur $\Delta\sigma_k = \sigma_{k+1} - \sigma_k$ est « grande » si les deux communautés que l'on fusionne à l'étape k sont loin l'une de l'autre. Inversement, si $\Delta\sigma_k$ est « grand », alors la partition obtenue à l'étape précédente est probablement une bonne partition. On introduit donc une nouvelle quantité η_k qui est le rapport entre $\Delta\sigma_k$ et $\Delta\sigma_{k-1}$

$$\eta_k = \frac{\Delta\sigma_k}{\Delta\sigma_{k-1}}.$$

Afin de détecter la meilleure partition, on va chercher la valeur maximale de η_k et on prendra la partition $\mathcal{P}_{k-1}$.

2.7 Algorithme complet et exemple

Maintenant que nous avons tous les éléments en main, décrivons l'algorithme complet et illustrons-le sur un exemple. Après avoir construit la matrice d'adjacence A du graphe et choisi un entier t, les étapes de l'algorithme sont les suivantes :

1. Création de $\mathcal{P}_1$: chaque sommet est une communauté à lui seul.
2. Création du vecteur D des degrés.
3. Création de la matrice de transition P où $P_{ij} = \frac{A_{ij}}{D_i}$.
4. Elever la matrice de transition à la puissance t.
5. Création de la matrice des distances entre sommets

$$R_{ij} = \begin{cases} \sqrt{\sum_{k=1}^{n} \frac{([P^t]_{ik} - [P^t]_{jk})^2}{d(k)}} & \text{si } v_i \text{ et } v_j \text{ sont adjacents,} \\ \infty & \text{sinon.} \end{cases}$$

6. Création de la matrice S où l'élément à la ligne i et colonne j vaut $\Delta\sigma(\mathcal{C}_i, \mathcal{C}_j)$

$$S_{ij} = \begin{cases} \frac{1}{n} \frac{|\mathcal{C}_i||\mathcal{C}_j|}{|\mathcal{C}_i|+|\mathcal{C}_j|} r^2(\mathcal{C}_i, \mathcal{C}_j) & \text{si } \mathcal{C}_i \text{ et } \mathcal{C}_j \text{ sont adjacents,} \\ \infty & \text{sinon} \end{cases}$$

   où $|\mathcal{C}_i| = |\mathcal{C}_j| = 1$.
7. Pour $k = 2, ..., n$, effectuer les étapes suivantes :
   (a) Chercher la position du minimum de la matrice S en ne considérant pas la diagonale (car on ne va pas fusionner une communauté avec elle-même), noter i et j l'indice de la ligne et de la colonne où il se trouve, en supposant sans perdre de généralité, que $i < j$. On va donc, dans les étapes suivantes, faire des modifications pour fusionner $\mathcal{C}_i$ et $\mathcal{C}_j$.
   (b) Egaler $\Delta\sigma_k$ à la valeur du minimum, et si $k > 2$, alors calculer

$$\eta_k = \frac{\Delta\sigma_k}{\Delta\sigma_{k-1}}.$$

   (c) Modifier la matrice S
      i. Si $\mathcal{C}_i \cup \mathcal{C}_j = \mathcal{C}_l$ alors, pour toute communauté $\mathcal{C}$ adjacente à $\mathcal{C}_l$, mettre à jour $\Delta\sigma(\mathcal{C}_l, \mathcal{C})$ de la façon suivante :

$$\Delta\sigma(\mathcal{C}_l, \mathcal{C}) = \frac{(|\mathcal{C}_i| + |\mathcal{C}|)\Delta\sigma(\mathcal{C}_i, \mathcal{C}) + (|\mathcal{C}_j| + |\mathcal{C}|)\Delta\sigma(\mathcal{C}_j, \mathcal{C}) - |\mathcal{C}|\Delta\sigma(\mathcal{C}_i, \mathcal{C}_j)}{|\mathcal{C}_i| + |\mathcal{C}_j| + |\mathcal{C}|}.$$

Notons que si $\Delta\sigma(\mathcal{C}_i, \mathcal{C}) = \infty$ ou $\Delta\sigma(\mathcal{C}_j, \mathcal{C}) = \infty$, i.e., si ils n'ont pas encore été calculés, alors on calcule leur valeur grâce à la formule suivante

$$\Delta\sigma(\mathcal{C}_1, \mathcal{C}_2) = \frac{1}{n}\frac{|\mathcal{C}_1||\mathcal{C}_2|}{|\mathcal{C}_1| + |\mathcal{C}_2|} r^2(\mathcal{C}_1, \mathcal{C}_2)$$

où $r^2(\mathcal{C}_1, \mathcal{C}_2) = \sum_{k=1}^{n} \frac{([P^t]_{\mathcal{C}_1 k} - [P^t]_{\mathcal{C}_2 k})^2}{d(k)}$.

ii. Supprimer la ligne j et et la colonne j de la matrice S.

(d) Modifier la matrice P^t

i. Remplacer la ligne i par

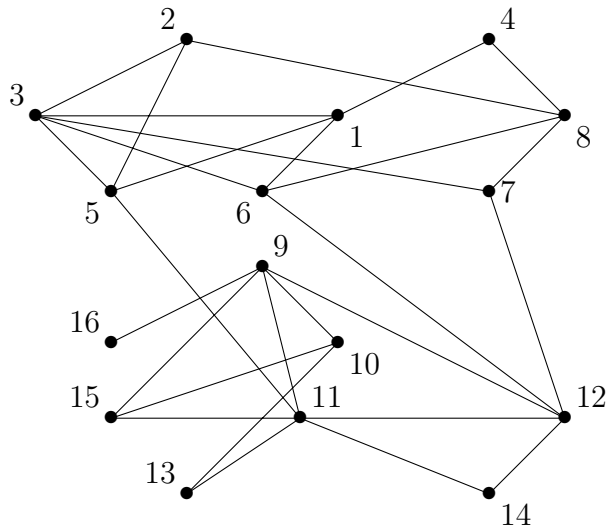
$$(P^t)_{\mathcal{C}_i \cup \mathcal{C}_j, h} = \frac{|\mathcal{C}_i|[P^t]_{\mathcal{C}_i h} + |\mathcal{C}_j|[P^t]_{\mathcal{C}_j h}}{|\mathcal{C}_i| + |\mathcal{C}_j|} \qquad \text{où } h \in \{1, ..., n\}.$$

ii. Supprimer la ligne j de la matrice P^t.

(e) Fusionner les communautés $\mathcal{C}_i$ et $\mathcal{C}_j$ et créer la partition $\mathcal{P}_k$.

8. Chercher l'élément k_0 appartenant à $\{1, ..., n\}$ tel que η_{k_0} soit maximum. La meilleure partition est donc $\mathcal{P}_{k_0}$.

Exemple 2.7.1. Reprenons le Graphe 2.2 qui est le suivant :



En exécutant l'algorithme détaillé ci-dessus, on obtient les étapes suivantes :

k	η_k	$\mathcal{P}_k$
1		$\{1\},\{2\},\{3\},\{4\},\{5\},\{6\},\{7\},\{8\},\{9\},$ $\{10\},\{11\},\{12\},\{13\},\{14\},\{15\},\{16\}$
2		$\{1\},\{2,3\},\{4\},\{5\},\{6\},\{7\},\{8\},\{9\},$ $\{10\},\{11\},\{12\},\{13\},\{14\},\{15\},\{16\}$
3	1.29688464347721232883	$\{1\},\{2,3\},\{4\},\{5\},\{6\},\{7\},\{8\},\{9\},$ $\{10,15\},\{11\},\{12\},\{13\},\{14\},\{16\}$
4	1.76358075549389248238	$\{1\},\{2,3,5\},\{4\},\{6\},\{7\},\{8\},\{9\},$ $\{10,15\},\{11\},\{12\},\{13\},\{14\},\{16\}$
5	1.00494826469289888493	$\{1,6\},\{2,3,5\},\{4\},\{7\},\{8\},\{9\},$ $\{10,15\},\{11\},\{12\},\{13\},\{14\},\{16\}$
6	1.13734022738747886372	$\{1,6,8\},\{2,3,5\},\{4\},\{7\},\{9\},$ $\{10,15\},\{11\},\{12\},\{13\},\{14\},\{16\}$
7	1.12072812102249463884	$\{1,6,8\},\{2,3,5\},\{4\},\{7\},\{9\},$ $\{10,13,15\},\{11\},\{12\},\{14\},\{16\}$
8	1.23246142791968682495	$\{1,6,8\},\{2,3,5\},\{4\},\{7\},\{9\},$ $\{10,13,15\},\{11\},\{12,14\},\{16\}$
9	1.26651075713782579335	$\{1,6,7,8\},\{2,3,5\},\{4\},\{9\},$ $\{10,13,15\},\{11\},\{12,14\},\{16\}$
10	1.11991244427602310019	$\{1,6,7,8\},\{2,3,5\},\{4\},\{9\},$ $\{10,13,15\},\{11,12,14\},\{16\}$
11	1.23869863087424780801	$\{1,2,3,5,6,7,8\},\{4\},\{9\},\{10,13,15\},$ $\{11,12,14\},\{16\}$
12	1.22193887133106001201	$\{1,2,3,5,6,7,8\},\{4\},\{9,10,13,15\},$ $\{11,12,14\},\{16\}$

Détaillons l'étape quand $k = 13$. A cette étape, on a les matrices suivantes :

$$S = \begin{pmatrix} 0 & 0.000246 & \infty & 0.002299 & \infty \\ 0.000246 & 0 & \infty & \infty & \infty \\ \infty & \infty & 0 & 0.001012 & 0.001018 \\ 0.002299 & \infty & 0.001012 & 0 & \infty \\ \infty & \infty & 0.001018 & \infty & 0 \end{pmatrix}$$

$$
\begin{aligned}
[P^t]_{1\bullet} &= (0.114, 0.097, 0.141, 0.069, 0.102, 0.103, 0.080, 0.117, 0.019,\\
&\qquad 0.005, 0.047, 0.067, 0.007, 0.020, 0.009, 0.002)\\
[P^t]_{2\bullet} &= (0.168, 0.077, 0.120, 0.153, 0.077, 0.109, 0.063, 0.177, 0,\\
&\qquad 0, 0.013, 0.043, 0, 0, 0, 0)\\
[P^t]_{3\bullet} &= (0.007, 0.005, 0.008, 0, 0.027, 0.012, 0.012, 0.003, 0.172,\\
&\qquad 0.162, 0.165, 0.077, 0.113, 0.043, 0.138, 0.055)\\
[P^t]_{4\bullet} &= (0.027, 0.021, 0.042, 0.009, 0.048, 0.054, 0.053, 0.029, 0.108,\\
&\qquad 0.058, 0.158, 0.141, 0.054, 0.103, 0.069, 0.024)\\
[P^t]_{5\bullet} &= (0, 0, 0, 0, 0.012, 0.014, 0.014, 0, 0.290,\\
&\qquad 0.097, 0.102, 0.081, 0.033, 0.026, 0.109, 0.222)
\end{aligned}
$$

et $\Delta\sigma_{12} = 0.000186$. Notons que les valeurs de ces matrices ont été arrondies. Exécutons à présent les étapes de la boucle.

(a) On cherche tout d'abord le minimum de la matrice S excepté la diagonale. Il se trouve en ligne 1, colonne 2 et sa valeur est 0.000246. On va donc faire des modifications pour pouvoir fusionner $\mathcal{C}_1 = \{1, 2, 3, 5, 6, 7, 8\}$ et $\mathcal{C}_2 = \{4\}$.

(b) On égale $\Delta\sigma_{13}$ à ce minimum et on calcule $\eta_{13} = \frac{\Delta\sigma_{13}}{\Delta\sigma_{12}} = \frac{0.000246}{0.000186} \approx 1.32$.

(c) On modifie la matrice S

i Pour toute communauté $\mathcal{C}$ adjacente à $\mathcal{C}_1 \cup \mathcal{C}_2$, mettre à jour $\Delta\sigma(\mathcal{C}_1 \cup \mathcal{C}_2, \mathcal{C})$. Ici, la seule communauté adjacente à $\mathcal{C}_1 \cup \mathcal{C}_2$ est la communauté $\mathcal{C}_4 = \{11, 12, 14\}$. On va donc calculer $\Delta\sigma(\mathcal{C}_1 \cup \mathcal{C}_2, \mathcal{C}_4)$. On a

$$
\begin{aligned}
\Delta\sigma(\mathcal{C}_1 \cup \mathcal{C}_2, \mathcal{C}_4) = \frac{1}{|\mathcal{C}_1| + |\mathcal{C}_2| + |\mathcal{C}_4|}\Big(&(|\mathcal{C}_1| + |\mathcal{C}_4|)\Delta\sigma(\mathcal{C}_1, \mathcal{C}_4)\\
&+ (|\mathcal{C}_2| + |\mathcal{C}_4|)\Delta\sigma(\mathcal{C}_2, \mathcal{C}_4) - |\mathcal{C}_4|\Delta\sigma(\mathcal{C}_1, \mathcal{C}_2)\Big).
\end{aligned}
$$

Cependant, $\Delta\sigma(\mathcal{C}_2, \mathcal{C}_4) = \infty$. On va donc le calculer

$$
\Delta\sigma(\mathcal{C}_2, \mathcal{C}_4) = \frac{1}{16}\frac{|\mathcal{C}_2||\mathcal{C}_4|}{|\mathcal{C}_2| + |\mathcal{C}_4|} r^2(\mathcal{C}_2, \mathcal{C}_4) \approx \frac{1}{16}\frac{3}{1+3} 0.031175 \approx 0.001461.
$$

On a donc

$$\begin{aligned}\Delta\sigma(\mathcal{C}_1 \cup \mathcal{C}_2, \mathcal{C}_4) &\approx \frac{1}{7+1+3}\Big((7+3)*0.002299 + (1+3)*0.001461 - 3*0.00024\Big)\\ &\approx 0.002556.\end{aligned}$$

ii On supprime la ligne 2 et la colonne 2. A présent, la matrice S vaut

$$S = \begin{pmatrix} 0 & \infty & 0.002556 & \infty \\ \infty & 0 & 0.001012 & 0.001018 \\ 0.002556 & 0.001012 & 0 & \infty \\ \infty & 0.001018 & \infty & 0 \end{pmatrix}.$$

d) On modifie la matrice P^t

i) Remplacer la ligne 1 par

$$(P^t)_{\mathcal{C}_1\cup\mathcal{C}_2,h} = \frac{|\mathcal{C}_1|[P^t]_{\mathcal{C}_1 h} + |\mathcal{C}_2|[P^t]_{\mathcal{C}_2 h}}{|\mathcal{C}_1| + |\mathcal{C}_2|} \qquad \text{où } h \in \{1, ..., 16\}.$$

Par exemple,

$$(P^t)_{\mathcal{C}_1\cup\mathcal{C}_2,3} = \frac{7*0.141 + 1*0.120}{7+1} = 0.138375.$$

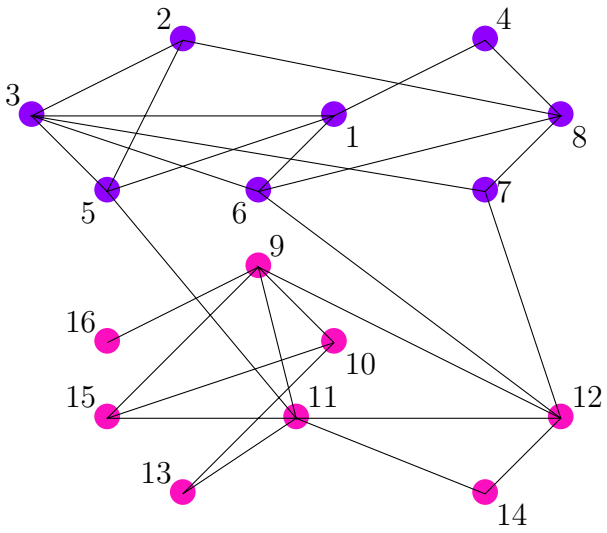
ii On supprime la ligne 2.

e) On fusionne les communautés $\mathcal{C}_1$ et $\mathcal{C}_2$ et on crée la partition $\mathcal{P}_{13}$.

On a donc,

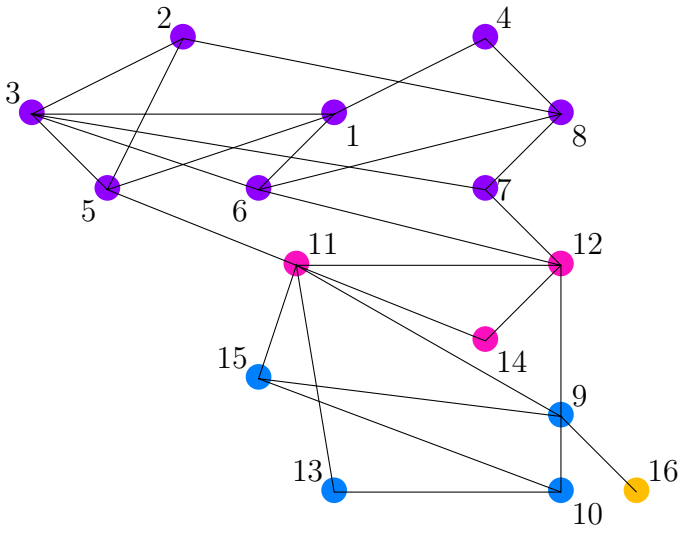
k	η_k	$\mathcal{P}_k$
13	1.32621224712892948894	$\{1,2,3,4,5,6,7,8\},\{9,10,13,15\},$ $\{11,12,14\},\{16\}$
14	4.10499399004491039022	$\{1,2,3,4,5,6,7,8\},\{9,10,11,12,13,14,15\},\{16\}$
15	1.21576380862344857192	$\{1,2,3,4,5,6,7,8\},\{9,10,11,12,13,14,15,16\}$
16	5.94119769313709866765	$\{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16\}$

La valeur de η_k est maximale lorsque $k = 16$, donc la meilleure partition est $\mathcal{P}_{15}$ comme illustré sur le Graphe 2.3 ci-dessous.



Graphe 2.3

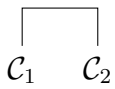
Cependant, on remarque que lorsque $k = 14$, η_k est beaucoup plus grand que tous les autres (excepté quand $k = 16$). Regardons sur notre graphe cette répartition des sommets :



La partition $\mathcal{P}_{13}$ ci-dessus a l'air d'être également une bonne partition mais moins que $\mathcal{P}_{15}$.

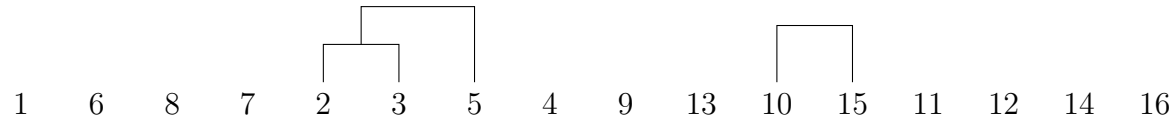
2.8 Construction du dendrogramme

Illustrons l'exécution de l'algorithme de l'Exemple 2.7.1 sur un dendrogramme. Pour rappel, il s'agit d'un arbre pour lequel chaque feuille correspond à un sommet du graphe de départ. On va construire l'arbre de bas en haut. A chaque étape de l'algorithme, on va regarder les deux communautés $\mathcal{C}_1$ et $\mathcal{C}_2$ qui sont fusionner et on va les relier dans l'arbre de la façon suivante :

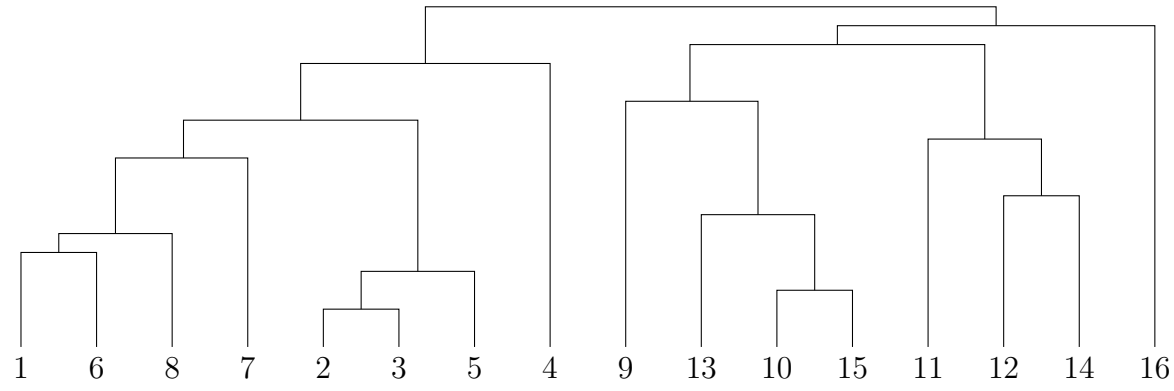


Notons que à chaque étape le segment horizontal construit sera une unité plus haute que le précédent. Ainsi, en regardant uniquement l'illustration du dendrogramme on peut retrouver toutes les étapes de l'algorithme. La barre horizontale correspond à présent à la communauté $\mathcal{C}_3$ qui est l'union de $\mathcal{C}_1$ et $\mathcal{C}_2$.

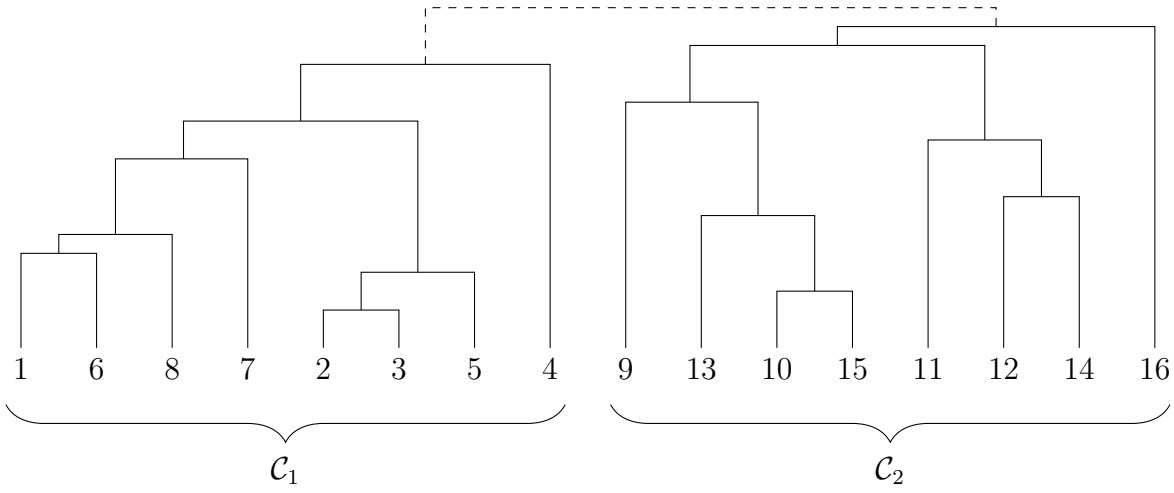
Dans notre exemple, à la première étape on fusionne le sommet 2 et le sommet 3, à l'étape suivante le 10 et le 15 et à la troisième étape on fusionne la communauté $\{2, 3\}$ avec le sommet 5. Ainsi, les trois premières étapes de l'algorithme sont représentées dans le dendrogramme comme suit :



Enfin, lorsque l'on exécute toutes les étapes de l'algorithme on obtient le graphe suivant :



On peut également représenter sur ce graphe le résultat final lorsque l'on choisit le η_k maximal. Dans ce cas ci, le η_k est maximal lorsque k est égal à 16. On va donc sur le dendrogramme ignorer les étapes plus grandes ou égales à 16 et on obtient les différentes communautés qui sont au nombre de deux dans notre cas :



2.9 Complexité théorique de l'algorithme

A présent, intéressons-nous à la complexité théorique de l'algorithme. La première étape consiste à créer la première partition et nécessite $\mathcal{O}(n)$ opérations.

La deuxième étape est la création du vecteur des degrés. Pour cela, on doit sommer chaque ligne de la matrice d'adjacence et donc cette étape demande $\mathcal{O}(n^2)$ opérations. Ensuite, on doit créer la matrice de transition P. Cette étape peut également se faire en $\mathcal{O}(n^2)$.

L'étape suivante consiste à élever la matrice P à la puissance t. Lorsque l'on effectue le produit matriciel entre deux matrices carrées A et B de dimension n en utilisant la formule

$$C_{ij} = \sum_{k=1}^{n} A_{ik} B_{kj},$$

le calcul de chaque élément de la nouvelle matrice se fait en n opérations. Or on a n^2 éléments à calculer donc le produit se fait en $\mathcal{O}(n^3)$. Le calcul de la puissance de notre matrice de transition aurait donc une complexité de $\mathcal{O}(tn^3)$. Cependant, dans notre cas, la matrice P est une matrice creuse, i.e., la plupart des éléments sont nuls. On peut donc stocker chaque élément non nul par un triplet contenant la valeur de

l'élément, ainsi que la ligne et la colonne où il se trouve. Dans notre cas, la matrice P possède $2m$ éléments non nuls, puisque le graphe possède m arêtes. Lorsque l'on fait le produit de la matrice avec elle-même, pour chaque ligne on a besoin d'au plus $2m$ opérations. Vu que l'on a n lignes et que l'on doit effectuer t produits matriciels, la complexité est en $\mathcal{O}(tnm)$.

A la cinquième étape, on va créer la matrice R des distances entre sommets. Si deux sommets v_i et v_j sont adjacents, on calcule leur distance via la formule

$$R_{ij} = \sqrt{\sum_{k=1}^{n} \frac{([P^t]_{ik} - [P^t]_{jk})^2}{d(k)}}$$

qui nécessite $\mathcal{O}(n)$ opérations. Puisque le graphe possède m arêtes, on va devoir effectuer ce calcul $2m$ fois. Ainsi la création de la matrice R est en $\mathcal{O}(mn)$.

Ensuite, on doit créer la matrice S des $\Delta\sigma$. Vu que l'on connait déjà la distance entre chaque paire de sommets, cette étape est simplement en $\mathcal{O}(n^2)$.

Ainsi, pour effectuer ces six premières étapes de base, la complexité théorique vaut $\mathcal{O}(n + n^2 + n^2 + tnm + mn + n^2) = \mathcal{O}(tnm)$.

A présent, calculons la complexité de la boucle que l'on devra effectuer $n - 1$ fois. Tout d'abord, on doit chercher le minimum dans la matrice S. Pour cela, si on le fait naïvement, on va parcourir toute la matrice qui est de taille $n - k + 2$. Cette étape de la boucle s'effectue donc en

$$\sum_{k=2}^{n}(n-k+2)^2 = n^2 + (n-1)^2 + \cdots + 2^2$$
$$\in \mathcal{O}\left(\frac{1}{6}(n-1)n(2(n-1)+1)\right) = \mathcal{O}(n^3).$$

Cependant, en changeant la structure des données, on peut améliorer la complexité de cette étape en $\mathcal{O}(n \log n)$. En effet, si on stocke la matrice S sous la forme d'un arbre équilibré [2], alors la recherche du minimum, l'ajout ou la suppression d'un élément se font en $\mathcal{O}(\log n)$.

La deuxième étape de la boucle est la mise à jour de la valeur de $\Delta\sigma_k$ qui se fait soit en temps constant si on peut directement appliquer le Théorème 2.5.5, soit en $\mathcal{O}(n)$ si certaines valeurs de $\Delta\sigma$ sont à calculer. Regardons, lorsque l'on va exécuter l'algorithme, combien de fois cette valeur devra être calculée. A chaque étape k, si

2. Dans un arbre, si pour n'importe quel noeud la différence de hauteur entre ses deux fils diffère d'au plus un, alors l'arbre est équilibré.

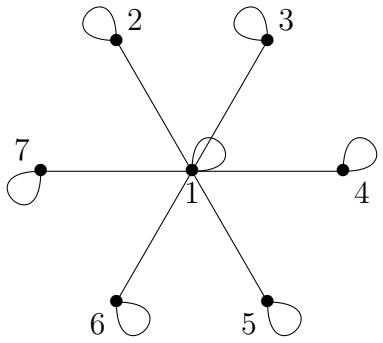
on fusionne les communautés $\mathcal{C}_i$ et $\mathcal{C}_j$, le nombre de fois que l'on devra mettre à jour une valeur $\Delta\sigma$ est égal au nombre de voisins de $\mathcal{C}_i \cup \mathcal{C}_j$ car on effectue la modification uniquement pour les communautés adjacentes. Soit H la hauteur du dendrogramme. Pour une hauteur $h \leq H$ donnée, les communautés avec la même hauteur h sont deux à deux disjointes, et la somme de leurs voisins est plus petite que $2m$ puisque au total il y a m arêtes. Ainsi, si on considère toutes les hauteurs que l'on va devoir parcourir, on aura au plus $2mH$ valeurs de $\Delta\sigma$ à calculer à chaque étape k. La complexité théorique de cette étape est donc en $\mathcal{O}(nmH)$.

La dernière étape de la boucle consiste à modifier la matrice de transition. Pour un k donné, le calcul du nouveau vecteur nécessite $\mathcal{O}(n)$ opérations. Puisque l'on effectue la boucle $n-1$ fois, la complexité est en $\mathcal{O}(n^2)$.

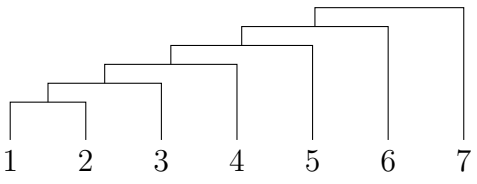
Ainsi la complexité totale de l'algorithme est $\mathcal{O}(nmt + n\log n + nmH + n^2)$ c'est-à-dire, en $\mathcal{O}(nmt + nmH)$.

Cependant, la valeur de t que l'on choisit est assez « petite » car le processus de chemin aléatoire converge de façon exponentielle. En pratique, prendre $t = \mathcal{O}(\log n)$ convient. Donc, la complexité théorique de l'algorithme est $\mathcal{O}(nmH)$.

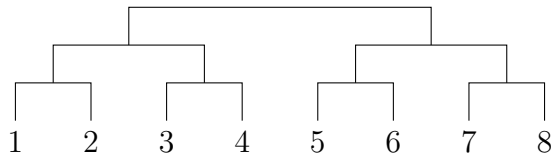
Analysons cette complexité en fonction de H. Le pire cas est quand H vaut $n-1$, i.e., quand les sommets sont fusionnés un à un avec une même grande communauté. Ce cas peut par exemple se produire lorsque l'on a un graphe en étoile. Voici un petit exemple de ce style de graphe avec 7 sommets.



Dans ce cas, le dendrogramme est de hauteur 6 et est le suivant :



Le meilleur cas se produit lorsque le dendrogramme est équilibré, et dans ce cas la hauteur est en $\mathcal{O}(\log n)$. Le dendrogramme dans ce cas-ci, pour un graphe à 8 sommets, est de hauteur 3 :

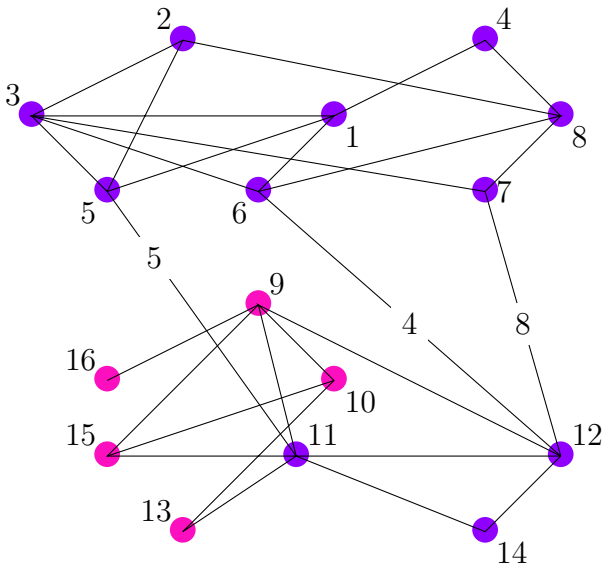


La complexité théorique de cet algorithme est donc en $\mathcal{O}(nmH)$ et on peut espérer qu'il soit en $\mathcal{O}(nm \log n)$. Remarquons qu'il s'agit d'une complexité théorique. Dans le cadre de ce mémoire, l'implémentation de l'algorithme est réalisée avec une complexité en $\mathcal{O}(n^3)$.

2.10 Généralisation aux graphes pondérés

L'algorithme du chemin aléatoire décrit ci-dessus, à jusqu'à présent été appliqué uniquement à des graphes non pondérés. Cependant, il se généralise tout naturellement aux graphes pondérés. En effet, considérons un graphe G pondéré. Dans ce cas, l'élément A_{ij} de la matrice d'adjacence est égal au poids de l'arête $\{v_i, v_j\}$, et le degré d'un sommet v_i est quant à lui égal à la somme des poids des arêtes incidentes à v_i. La définition de la matrice de transition reste inchangée, i.e., $P_{ij} = \frac{A_{ij}}{d(i)}$ et malgré le changement de la matrice d'adjacence, la matrice de transition est toujours une matrice stochastique. C'est pourquoi, toutes les propriétés et théorèmes que nous avons démontrés sont toujours vérifiés et l'algorithme se déroule de la même manière que lorsque l'on était dans le cas d'un graphe non pondéré.

Si l'on reprend le Graphe 2.2 et que l'on ajoute un poids sur 3 arêtes, en exécutant l'algorithme on obtient la répartition en communautés suivante :



Les poids jouent donc un rôle important dans la détection des communautés, mais ne changent rien dans l'implémentation de l'algorithme.

Chapitre 3

Méthode heuristique

Dans ce chapitre, nous allons présenter un deuxième algorithme permettant d'extraire les communautés dans un graphe. Il s'agit d'une méthode heuristique qui a été développée dans l'article [3] par Vincent Blondel, Jean-Loup Guillaume, Renaud Lambiotte et Etienne Lefebvre. Leur algorithme peut être appliqué aux graphes non-pondérés tout comme aux graphes pondérés et il repose sur la notion de modularité d'une partition qui a été introduite par Mark Newman [8].

3.1 Introduction et modularité

Rappelons tout d'abord deux définitions adaptées aux graphes pondérés. L'élément A_{ij} de la matrice d'adjacence d'un graphe G est le poids de l'arête entre le sommet v_i et le sommet v_j. Le degré d'un sommet v_i est égal à $d_i = \sum_{j=1}^{n} A_{ij}$, i.e., la somme des poids des arêtes incidentes à v_i.

Introduisons à présent la notion de modularité d'une partition d'un graphe. La modularité permet de mesurer la qualité de la séparation des sommets en communautés dans un graphe. Supposons avoir une répartition en communautés. Définissons une nouvelle fonction f qui à chaque sommet v_i associe l'indice de la communauté à laquelle il appartient. Dans ce cas, $\mathcal{C}_{f(i)}$ est la communauté contenant le sommet v_i.

Si on regarde la proportion du nombre d'arêtes appartenant à une communauté, i.e., les arêtes reliant deux sommets d'une même communauté, par rapport au nombre total d'arêtes, elle est égale à

$$\frac{\sum_{i=1}^{n}\sum_{j=1}^{n} A_{ij}\delta(\mathcal{C}_{f(i)}, \mathcal{C}_{f(j)})}{\sum_{i=1}^{n}\sum_{j=1}^{n} A_{ij}} \tag{3.1}$$

où

$$\delta(\mathcal{C}_{f(i)}, \mathcal{C}_{f(j)}) = \begin{cases} 1 & \text{si } \mathcal{C}_{f(i)} = \mathcal{C}_{f(j)}, \text{i.e., si } v_i \text{ et } v_j \text{ appartiennent} \\ & \text{à la même communauté,} \\ 0 & \text{sinon.} \end{cases}$$

Ainsi, au plus ce nombre sera grand au plus on peut espérer avoir une bonne partition puisque cela signifie qu'il existe peu d'arêtes entre les différentes communautés. Afin d'améliorer la modularité, une autre valeur sera utilisée. Il s'agit de la probabilité que le sommet v_i et le sommet v_j soient connectés si le graphe est construit aléatoirement tout en respectant le degré de chaque sommet. Cette probabilité est égale à

$$\frac{d_i d_j}{\sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij}}. \tag{3.2}$$

Notons

$$p = \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij}$$

autrement dit, p est la somme des poids de toutes les arêtes. Ainsi, en utilisant les valeurs (3.1) et (3.2), la modularité d'une partition est définie de la façon suivante :

$$Q = \frac{1}{2p} \sum_{i=1}^{n} \sum_{j=1}^{n} \Big(A_{ij} - \frac{d_i d_j}{2p} \Big) \delta(c_i, c_j).$$

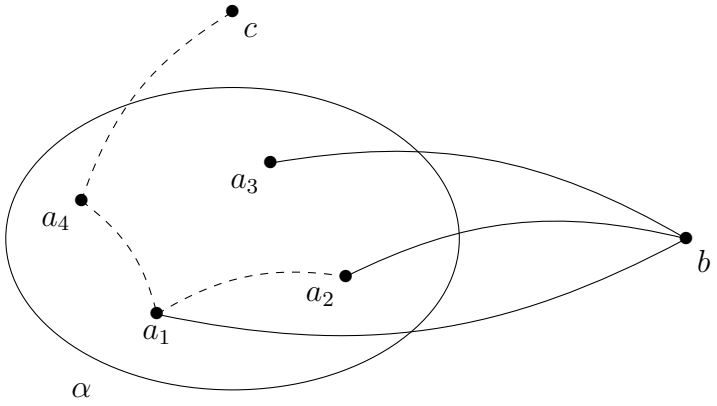
Le but de l'algorithme est donc de maximiser cette modularité afin d'obtenir une bonne partition en communautés par rapport à celle que l'on pourrait obtenir de façon aléatoire.

3.2 Algorithme

L'initialisation de la méthode, tout comme celle du chemin aléatoire, consiste à assigner une communauté différente à chaque sommet. Ensuite, on va itérer deux étapes un certain nombre de fois.

A la première étape, pour chaque sommet isolé v_i, i.e., pour chaque sommet étant seul dans sa communauté, on va tout d'abord considérer tous ses voisins et ensuite calculer pour chacun d'entre eux le gain de modularité si on place le sommet v_i dans la communauté de son voisin.

Pour illustrer le calcul du gain de modularité, considérons avoir une communauté nommée α et supposons que l'on souhaite déplacer le sommet b dans cette communauté.



Si on calcule la différence entre la modularité obtenue si b appartient à α et la modularité dans la situation initiale où b est un sommet isolé, tous les termes vont s'annuler sauf ceux qui font intervenir directement le sommet b. Vu que la matrice d'adjacence A est symétrique, on a

$$\begin{aligned}\Delta Q(b,\alpha) &= \frac{1}{2p}\sum_{a_i\in\alpha} A_{a_i b} + \frac{1}{2p}\sum_{a_j\in\alpha} A_{ba_j} - \frac{1}{(2p)^2}\sum_{a_i\in\alpha} d_{a_i}d_b - \frac{1}{(2p)^2}\sum_{a_j\in\alpha} d_b d_{a_j} \\ &= \frac{1}{p}\sum_{a_i\in\alpha} A_{ba_i} - \frac{d_b}{2p^2}\sum_{a_i\in\alpha} d_{a_i}.\end{aligned}$$

D'où, si on note

$$w_{b\to\alpha} = \sum_{a_i\in\alpha} A_{ba_i}$$

i.e., la somme des poids des arêtes dont l'origine est le sommet b et l'extrémité est un sommet appartenant à α , et si on note

$$w_\alpha = \sum_{a_i\in\alpha} d_{a_i}$$

i.e., la somme des degrés de tous les sommets de α, alors

$$\Delta Q(b,\alpha) = \frac{w_{b\to\alpha}}{p} - \frac{d_b w_\alpha}{2p^2} = \frac{2pw_{b\to\alpha} - d_b w_\alpha}{2p^2}.$$

Si il existe un gain de modularité positif, on va prendre celui qui est maximum et déplacer v_i dans la communauté correspondante. Si à chaque fois que l'on calcule le gain de modularité on obtient une valeur négative, alors on ne déplace pas le

sommet v_i. On répète cette étape jusqu'à ce que plus aucun changement ne soit possible. Remarquons que le résultat de cette première étape dépend de l'ordre dans lequel on considère les sommets isolés, mais cela n'aura pas un impact significatif sur le résultat final.

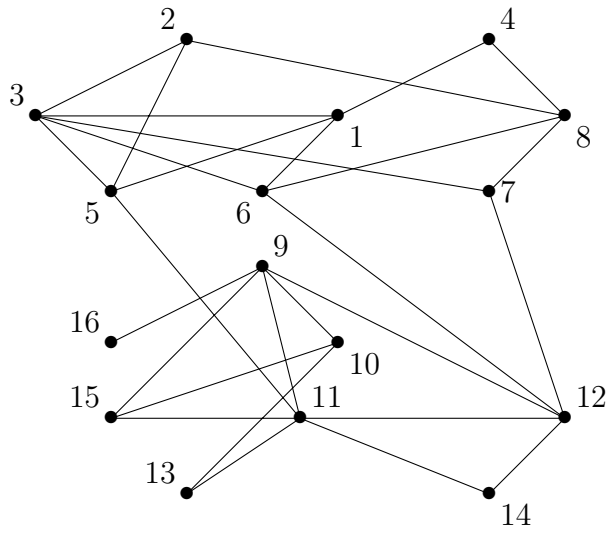
Notons également que, en pratique, il suffit de calculer le numérateur de ΔQ car le dénominateur étant fixé, maximiser la fraction revient à maximiser le numérateur. Par la suite, on notera la valeur de ce numérateur ΔQ_N.

Lorsque cette première étape est terminée, on exécute la deuxième. Cette phase consiste à créer un nouveau graphe G' où les sommets de celui-ci sont les communautés construites lors de la première étape. De plus, dans le graphe G', il existe une arête entre deux sommets v'_i et v'_j si il existe au moins une arête dans G entre un sommet de la communauté représentée par v'_i et un sommet de la communauté représentée par v'_j. Le poids de cette arête sera égal à la somme des poids des arêtes existantes entre v'_i et v'_j. Concernant les arêtes qui sont incluses dans une communauté, elles vont devenir une boucle sur le nouveau sommet et le poids de cette boucle sera égal à la somme des poids des boucles présentes sur les sommets et du double de la somme des poids des arêtes entre deux sommets différents de la communauté.

Ensuite, on recommence à la première étape avec le nouveau graphe G' jusqu'à ce que plus aucun changement ne soit possible. A la fin de cet algorithme on obtient un graphe dont chaque sommet, qui est constitué d'un ensemble de sommets du graphe initial, représente une communauté.

3.3 Exemple

Illustrons cet algorithme sur un exemple. Reprenons celui développé lors de l'algorithme du chemin aléatoire sur le Graphe 2.2 mais où les boucles sur chaque sommet ont été supprimées et où le poids sur chaque arête vaut 1.



On a

$$p = \frac{1}{2} \sum_{i=1}^{16} \sum_{j=1}^{16} A_{ij} = 28.$$

Commençons la première étape en prenant le sommet 1. On a

- $\Delta Q_N(1,3) = 2.28.1 - 4.5 = 36$
- $\Delta Q_N(1,4) = 2.28.1 - 4.2 = 48$
- $\Delta Q_N(1,5) = 2.28.1 - 4.4 = 40$
- $\Delta Q_N(1,6) = 2.28.1 - 4.4 = 40.$

La valeur de ΔQ_N est maximale lorsque l'on place le sommet 1 avec le sommet 4. On va donc fusionner ces deux sommets puis recommencer en utilisant le sommet 2.

- $\Delta Q_N(2,3) = 2.28.1 - 3.5 = 41$
- $\Delta Q_N(2,5) = 2.28.1 - 3.4 = 46$
- $\Delta Q_N(2,8) = 2.28.1 - 3.4 = 46.$

Ici, ΔQ_N est maximal dans deux cas. On prendra comme convention de choisir le premier. Ainsi, on fusionne le sommet 2 et le sommet 5. Pour l'étape suivante, prenons le sommet 3. Dans ce cas-ci, on a

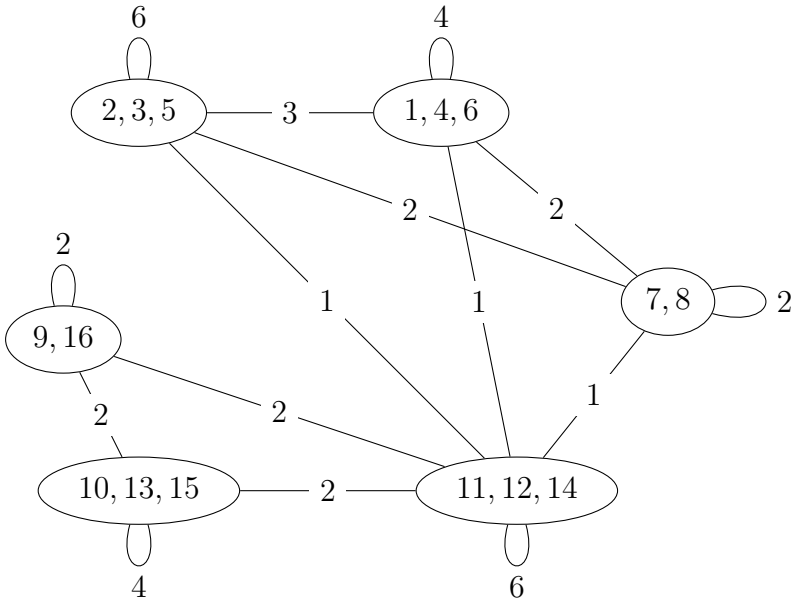
- $\Delta Q_N(3,\{1,4\}) = 2.28.1 - 5.4 = 36$
- $\Delta Q_N(3,\{2,5\}) = 2.28.2 - 5.5 = 87$
- $\Delta Q_N(3,6) = 2.28.1 - 5.4 = 36$

- $\Delta Q_N(3,7) = 2.28.1 - 5.3 = 41.$

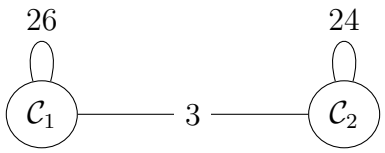
La valeur de ΔQ_N est maximale dans le deuxième cas. D'où, on déplace le sommet 3 dans la communauté $\{2,5\}$. En continuant cette première phase de l'algorithme, on arrive à la partition des sommets suivante :

$$\mathcal{P} = \{\{1,4,6\},\{2,3,5\},\{7,8\},\{9,16\},\{10,13,15\},\{11,12,14\}\}.$$

On exécute ensuite la deuxième étape de l'algorithme afin de créer le graphe suivant :



On recommence la même procédure à la première étape. Après avoir terminé les deux phases, on obtient les communautés $\mathcal{C}_1 = \{1,2,3,4,5,6,7,8\}$ et $\mathcal{C}_2 = \{9,10,11,12,13,14,15,16\}$ ainsi que le graphe suivant :



Dans ce cas ci, p vaut 53 et si on calcule le gain de modularité on a

$$\Delta Q_N(\mathcal{C}_1,\mathcal{C}_2) = 2.53.3 - 27.29 < 0.$$

Ainsi, l'algorithme se termine et on obtient comme résultat deux communautés, $\mathcal{C}_1$ et $\mathcal{C}_2$, qui sont dans ce cas-ci les mêmes que celles obtenues grâce à la méthode du chemin aléatoire.

Chapitre 4

Implémentation et application

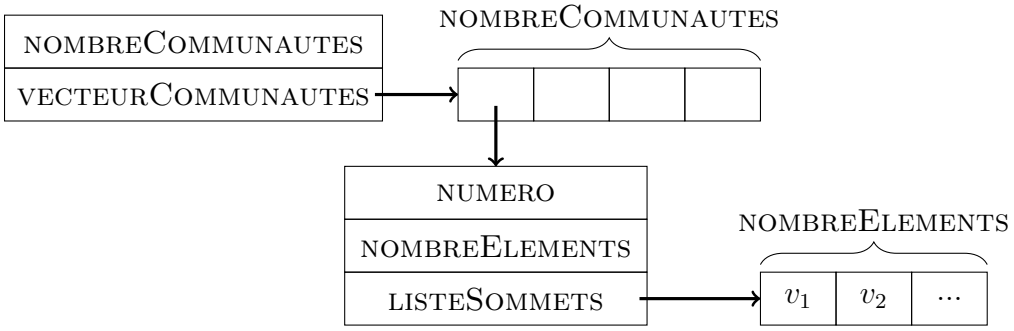
Les deux algorithmes présentés dans ce mémoires ont été implémentés dans le langage de programmation C et sont disponibles en annexe de ce travail sur la plateforme « MatheO ».

Dans ce chapitre, nous allons dans un premier temps décrire la structure des données utilisée dans ces deux programmes et ensuite nous détaillerons les résultats obtenus lorsque l'on applique les algorithmes sur la base de données institutionnelle « ORBi » de l'Université de Liège [1].

4.1 Méthode du chemin aléatoire

Dans cette section nous allons décrire la structure de données utilisée pour l'algorithme du chemin aléatoire. Pour cette méthode, deux structures différentes sont utilisées. La première structure, appelée COMMUNAUTE, permet de représenter une communauté. Elle est composée de trois éléments. Le premier est une variable de type SIZE_T nommée NUMERO qui est un nombre attribué à la communauté et qui permet de faire facilement le lien avec les matrices construites par la suite. Le deuxième est une variable de même type, appelée NOMBREELEMENTS et comme son nom l'indique contient le nombre d'éléments présents dans la communauté. La dernière variable, LISTESOMMETS, est un pointeur vers un tableau d'entiers contenant les sommets présents dans la communauté. La deuxième structure, appelée ENSEMBLEDESCOMMUNAUTES, représente l'ensemble des communautés du graphe. Elle est composée de deux variables. La première, NOMBRECOMMUNAUTES, est de type SIZE_T et correspond au nombre de communautés présentes dans l'ensemble. La deuxième variable, VECTEURCOMMUNAUTES, est un pointeur vers un tableau de COMMUNAUTES de taille NOMBRECOMMUNAUTÉS et contient toutes les communautés de l'ensemble.

Voici un schéma récapitulatif de ces deux structures.



4.2 Algorithme heuristique

A présent, nous allons décrire la structure de données utilisée pour l'algorithme heuristique. Dans ce cas-ci, la structure est plus complexe.

La structure de données pour les communautés est composée de cinq éléments au lieu de trois. Le premier est, comme dans le premier cas, une variable de type size_t nommée numero qui est un nombre attribué à la communauté et qui permet de faire facilement le lien avec les matrices construites par la suite.

La structure restante est quant à elle modifiée. En effet, lors de l'exécution de cet algorithme, on a un graphe de départ puis au fur et à mesure, on va construire de nouveaux graphes de plus en plus petits. Cependant, il faut garder en mémoire les sommets qui ont été regroupés à la première étape puis aux étapes suivantes pour qu'à la fin de l'algorithme on puisse identifier les communautés avec les sommets du graphe de départ. C'est pourquoi, on utilise deux variables, de type size_t, qui contiennent le nombre d'éléments dans la communauté. Une des deux s'appelle nbrElementsActuel et contient le nombre de sommets présents dans le graphe actuellement traité et éventuellement plus petit que le graphe initial. L'autre est nommée nbrElementsDepart et contient le nombre d'éléments présents dans la communauté si l'on se réfère au graphe initial. Les deux dernières variables sont chacune un pointeur vers un tableau d'entiers. L'une s'appelle listeSommetsActuel et contient la liste des sommets présents dans le graphe actuellement traité tandis que l'autre est nommée listeSommetsDepart contient tous les sommets si l'on se réfère au graphe de départ.

La deuxième structure est la même que dans le cas de l'algorithme du chemin aléatoire et porte le même nom. Illustrons tout d'abord cette structure sur l'Exemple 3.3 avant d'en donner un schéma.

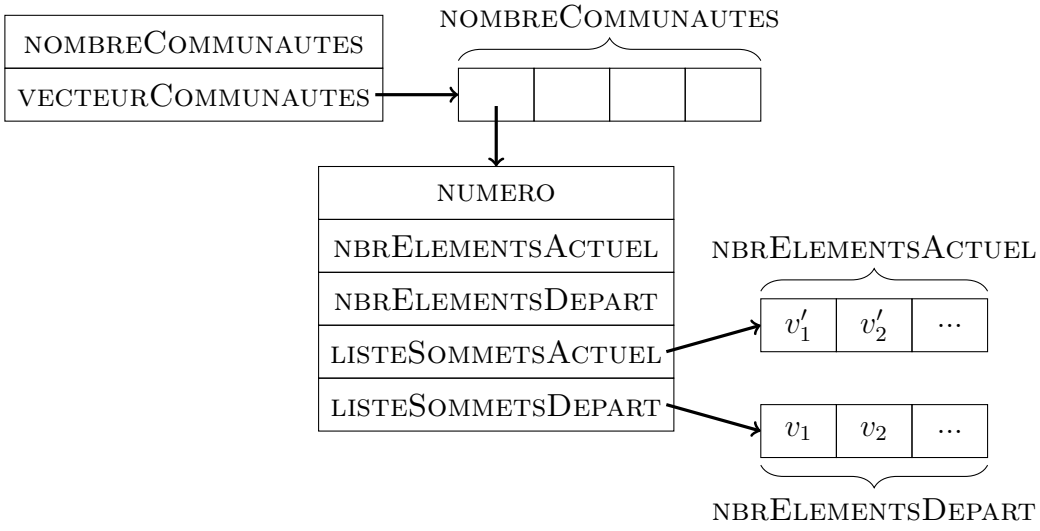
A la fin de la première étape, on obtient la partition des sommets suivante :

$$\mathcal{P} = \{\{1, 4, 6\}, \{2, 3, 5\}, \{7, 8\}, \{9, 16\}, \{10, 13, 15\}, \{11, 12, 14\}\}.$$

Ensuite on modifie le graphe initial où chaque sommet correspond à une communauté. Une fois le graphe modifié, dans la structure EnsembleDesCommunautes, la première variable est donc égale à 6 puisqu'on initialise l'algorithme en attribuant une communauté différente pour chaque sommet. La variable vecteurCommunautes pointe vers un tableau de taille 6 où chaque élément représente une communauté. La variable numero de chacune des communautés est différente et va de 1 à 6. Si on considère la première communauté, la variable nbrElementsActuel est égale à 1 puisque chaque communauté contient un seul sommet du nouveau graphe. La variable nbrElementsDepart est égale à 3 car dans ce cas-ci on regarde tous les sommets présents dans la communauté lorsque l'on se réfère au graphe de départ. Concernant les deux dernières variables, l'une pointe vers un tableau de un élément qui contient le nom du sommet qui a été créé avec la première communauté, et la deuxième pointe vers un tableau de 3 éléments qui sont les sommets $1, 4$ et 6.

Remarquons que pour la présentation des exemples de ce travail, les numérotations des sommets, des étapes, ... commencent à 1. Cependant dans l'implémentation du programme tous les comptes se font à partir de 0.

Voici à présent le schéma récapitulatif de cette structure de données.



4.3 Application

Dans cette section, nous allons détailler les résultats obtenus lorsque l'on exécute les deux algorithmes sur la base de données institutionnelle de l'Université de Liège. Afin de respecter le règlement général sur la protection des données, les données ont été anonymisées, et nous n'avons donc pas les moyens d'approfondir ces résultats. Le graphe G construit à partir de cette base de données est composé de 10413 sommets qui correspondent aux chercheurs faisant partie de l'Université de Liège. Remarquons que lorsque l'on consulte le site internet « ORBi », le nombre de chercheurs est légèrement plus grand. Cette différence est dûe au fait que dans la base de données utilisée pour ce travail, certaines données ne possèdent pas le format adéquat pour être traitées et donc certains auteurs n'ont pas été pris en compte. Concernant les arêtes, il existe une arête entre deux chercheurs si ils ont écrit au moins un article ensemble. En fonction du nombre d'articles écrits en commun, un poids est également attribué aux arêtes. Notons également que pour l'algorithme du chemin aléatoire, une boucle de poids 1 est ajoutée sur chaque sommet.

Une fois le graphe construit, avant d'exécuter les deux algorithmes, il faut vérifier que le graphe est connexe. Or le graphe G extrait de la base de données « ORBi » ne l'est pas. Il faut donc tout d'abord identifier les différentes composantes connexes du graphe. Cela peut se faire facilement en implémentant un algorithme de parcours en profondeur. Le résultat de cet algorithme est le suivant : il existe

- une composante de 9637 sommets
- une composante de 5 sommets qui correspondent tous les 5 à des chercheurs du département de langues modernes
- 3 composantes de 4 sommets, une composée uniquement de criminologues, une seconde composée de deux chercheurs en sciences historiques et deux chercheurs en sciences de l'antiquité, et la troisième composée de chercheurs dont le département est non identifiable via la base de données
- 2 composantes de 3 sommets, toutes les deux composées de 3 chercheurs du département de HEC
- 17 composantes de 2 sommets
- 719 composantes avec un unique sommet.

On peut constater que la composition des petites composantes connexes a du sens. Vu leur taille, il n'est pas utile d'exécuter les algorithmes de recherche de communautés sur ces petites composantes. Cependant, il est intéressant de l'exécuter sur la grande composante composée de près de 10000 sommets. Notons G' le sous-graphe de G qui correspond à la première composante connexe.

4.3.1 Méthode du chemin aléatoire

Premièrement exécutons l'algorithme du chemin aléatoire sur G'. Un choix pour le paramètre t doit être fait. Plusieurs tests ont été réalisés. Tout d'abord, on a pris une valeur de t égale à 6 et à 10, cependant les résultats n'ont pas été concluants. L'algorithme a ensuite été testé avec une valeur plus petite. On a pris $t = 3$ et $t = 4$. Les résultats pour ces deux valeurs sont forts similaires, néanmoins, celui avec un $t = 3$ semble être légèrement meilleur. En effet, lorsque $t = 4$, le résultat final contient moins de communautés mais celles-ci sont moins homogènes, i.e., sont composées de chercheurs provenant de plus de départements différents. Détaillons donc le résultat obtenu lorsque l'on exécute l'algorithme avec $t = 3$.

Le nombre de communautés obtenu dans G' est égal à 1193 dont 1028 sont composées de moins de 10 sommets. Parmi les communautés restantes, nous allons détailler les 10 plus grandes. Dans l'annexe B, deux tableaux sont donnés. Le premier, dans la section B.1, est le lien entre le nom des départements et le numéro qu'il lui a été attribué pour l'implémentation. Dans ce tableau, on peut voir que 44 départements ont été considérés. Le dernier département porte le nom d'un point d'interrogation. En effet, le département de certains chercheurs est inexistant dans la base de données institutionnelle « ORBi », et donc un département neutre a été créé pour pouvoir les inclure. Le second tableau, présent dans la section B.2, est l'analyse détaillée des 10 plus grandes communautés. Chaque ligne correspond à une communauté. Ensuite, pour une ligne i fixée, le nombre inscrit dans la colonne j correspond au nombre de chercheurs présents dans la communauté i et qui appartiennent au département j. Enfin, dans la toute dernière colonne, le nombre total de chercheurs présents dans chaque communauté a été ajouté. Le tableau complet est disponible en annexe de ce travail sur la plateforme « MatheO ».

On peut donc voir, grâce à ces deux tableaux, que la première communauté contient des chercheurs de presque tous les départements.

La deuxième communauté est principalement composée de chercheurs faisant partie du département « AgroBio Tech », ainsi que du département de chimie.

La communauté suivante est composée de 361 chercheurs. Parmi eux, 272 font partie de la faculté de médecine vétérinaire, 46 appartiennent à la faculté des sciences, des sciences appliquées ou de la faculté de médecine, 34 n'ont pas été identifiés à partir de la base de données et les 9 chercheurs restants sont répartis dans différents départements comme par exemple celui « d'AgroBio Tech ».

La quatrième communauté est composée de 316 sommets dont 181 qui correspondent à des chercheurs de la faculté de médecine, 80 font partie de la faculté de psychologie et 37 sont des chercheurs de la faculté des sciences.

Les trois communautés suivantes sont principalement composées de chercheurs

faisant partie de la faculté de médecine.

La huitième communauté est composée de 101 chimistes, 2 physiciens, 13 chercheurs du département de sciences de la vie, un chercheur de la faculté de médecine, un autre faisant partie du département aérospatiale, et un chercheur de la faculté de médecine vétérinaire.

Les deux communautés suivantes sont principalement un mélange de chercheurs de la faculté des sciences, médecine, sciences appliquées et de la faculté de médecine vétérinaire.

Regardons d'un peu plus près les communautés contenant des mathématiciens. Certains appartiennent à des communautés exclusivement composées de mathématiciens. Pour ce type de communautés, on recense :

— une communauté de 12
— une communauté de 10
— une communauté de 3
— deux communautés de 2
— 10 communautés d'une seule personne.

Ensuite il y a des mathématiciens qui appartiennent à des communautés mixtes, i.e., qui sont composées de chercheurs de différents départements. Pour ces communautés, on a

— 6 mathématiciens qui font partie de la première communauté qui est un mélange de presque tous les départements
— un mathématicien qui appartient à une communauté composée de 12 physiciens et un chercheur du département de sciences cliniques
— un mathématicien qui fait partie d'une communauté composée de 10 géographes et 2 chercheurs du département historique
— un mathématicien qui appartient à une communauté avec 6 chercheurs de « Montéfiore » et un chercheur « d'AgroBio Tech »
— 4 mathématiciens sont dans une communauté avec un chercheur du département d'aérospatiale et mécanique
— 3 mathématiciens sont avec un chercheur du département de HEC
— 2 font partie d'une communauté avec un chercheur de « Montéfiore »
— un mathématicien est dans une communauté avec une personne faisant partie des services généraux.

L'interprétation de ces résultats ne peut pas se faire plus en profondeur afin de garder la confidentialité des chercheurs et respecter le règlement général sur la protection des données.

4.3.2 Méthode heuristique

L'exécution du deuxième algorithme nous donne un résultat fort différent et nous apporte peu d'informations. En effet, cet algorithme a séparé le graphe G' en seulement 7 communautés.

Une première communauté est composée de 9053 chercheurs provenant de tous les départements. Les autres communautés sont beaucoup plus petites.
On recense :

- une communauté qui contient principalement des chercheurs de la faculté de médecine et de la faculté de psychologie
- une communauté composée quasiment que de chimistes et de chercheurs du département « d'AgroBio Tech »
- une communauté composée de 21 architectes, 11 chercheurs du département « ArGenCo », un mathématicien et quelques autres chercheurs appartenant à divers départements
- une communauté de 38 physiciens et 4 chercheurs de différents départements
- une communauté composée de 28 mathématiciens, 4 chercheurs du département de HEC, un chercheur du département d'aérospatiale, un chercheur en sciences sociales, une personne faisant partie du département des services généraux et un chercheur dont le département n'a pas pu être identifié
- une communauté avec 13 chercheurs de sciences historiques.

On constate que ces six communautés ont du sens, cependant la communauté de 9053 chercheurs ne nous apporte pas beaucoup d'informations mais permet de supposer que tous les départements de l'Université sont fort liés et collaborent entre eux. Un tableau contenant l'analyse détaillée de ces résultats est donné dans l'annexe B à la section B.3.

On remarque que, dans ce cas-ci, pour un même graphe, les deux algorithmes nous donnent un résultat fort différent. Ce résultat n'est cependant pas étonnant puisqu'il n'existe pas de définition rigoureuse pour une communauté.

Bibliographie

[1] Disponible via `https://orbi.uliege.be`.

[2] Lada Adamic : *World Wide Web, Graph Structure*, pages 10058–10072. Springer New York, New York, NY, 2009.

[3] Vincent Blondel, Jean-Loup Guillaume, Renaud Lambiotte et Etienne Lefebvre : Fast unfolding of communities in large networks. *Journal of Statistical Mechanics : Theory and Experiment*, 2008(10):P10008, 2008.

[4] Peter Chin, Anup Rao et Van Vu : Stochastic block model and community detection in sparse graphs : A spectral algorithm with optimal rate of recovery. *In Proceedings of The 28th Conference on Learning Theory*, volume 40 de *Proceedings of Machine Learning Research*, pages 391–423, Paris, France, 2015. PMLR.

[5] Michel Jambu et Marie-Odile Lebeaux : *Cluster Analysis and Data Analysis*. North Holland Publishing, 1983.

[6] Hawoong Jeong, Bálint Tombor, Reka Albert, Zoltán Oltvai et Albert-Laszlo Barabási : The large-scale organization of metabolic networks. *Nature*, 407(6804):651–654, 2000.

[7] Meena Mahajan, Prajakta Nimbhorkar et Kasturi Varadarajan : The planar K-means problem is NP-hard. *Theoretical Computer Science*, 442:13–21, 2012. Special Issue of the Workshop on Algorithms and Computation (WALCOM 2009).

[8] Mark Newman : Spectral methods for community detection and graph partitioning. *Phys. Rev. E*, 88:042822, Oct 2013.

[9] Pascal Pons et Matthieu Latapy : Computing communities in large networks using random walks. *Journal of Graph Algorithms and Applications*, 10(2):191–218, 2006.

[10] Michel Rigo : Algèbre linéaire. Département de Mathématique de l'Université de Liège, 2009–2010. Disponible via `http://www.discmath.ulg.ac.be/cours/main_math.pdf`, notes de cours.

[11] Michel Rigo : Théorie des graphes. Département de Mathématique de l'Université de Liège, 2009–2010. Disponible via `http://www.discmath.ulg.ac.be/cours/main_graphes.pdf`, notes de cours.

[12] Jean-Jacques Ruch et Marie-Line Chabanol : Chaînes de markov. Préparation à l'Agrégation Bordeaux 1, 2012–2013. Disponible via `https://www.math.u-bordeaux.fr/~mchabano/Agreg/ProbaAgreg1213-COURS5-CM.pdf`, consulté le 05/10/2018, notes de cours.

[13] Hua-Wei Shen et Xue-Qi Cheng : Spectral methods for the detection of network community structure : a comparative analysis. *Journal of Statistical Mechanics : Theory and Experiment*, 2010(10):P10020, oct 2010.

[14] Vincent Antonio Traag : *Algorithms and Dynamical Models for Communities and Reputation in Social Networks*. Springer Theses. Springer, Heidelberg, 2014.

[15] Joe Ward : Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58(301):236–244, 1963.

[16] Todd Waskiewicz : Friend of a friend influence in terrorist social network. *In Proceedings on the International Conference on Artificial Intelligence*. The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2012. Disponible via `https://www.hsdl.org/?view&did=744909`.

Annexe A

Matrice de transition

$[P^t]_{1\bullet}$= (
0.15044444444444446396 0.08788888888888890527 0.14155555555555557334
0.10088888888888891682 0.11626984126984128698 0.12022222222222224419
0.06188888888888889606 0.10622222222222221788 0.0123809523809523811
0.00000000000000000000 0.03504761904761904967 0.04338095238095238115
0.00571428571428571515 0.01238095238095238311 0.00571428571428571515
0.00000000000000000000)

$[P^t]_{2\bullet}$= (
0.10986111111111111771 0.13006944444444445974 0.16715277777777781676
0.05750000000000000944 0.13304563492063495311 0.08486111111111112326
0.07111111111111111105 0.13145833333333334370 0.00714285714285714263
0.00000000000000000000 0.03797619047619048283 0.04839285714285715412
0.00714285714285714263 0.00714285714285714263 0.00714285714285714263
0.00000000000000000000)

$[P^t]_{3\bullet}$= (
0.11796296296296295003 0.11143518518518517879 0.14990740740740737813
0.05999999999999999778 0.11841931216931214643 0.11546296296296294781
0.08837962962962962743 0.09638888888888887119 0.01726190476190475956
0.00000000000000000000 0.04448412698412698013 0.05351190476190476053
0.00476190476190476147 0.01726190476190475956 0.00476190476190476147
0.00000000000000000000)

$[P^t]_{4\bullet}$= (0.16814814814814993 0.07666666666666675 0.12000000000000944
0.15259259259258523 0.07666666666666667462 0.10888888888888889617
0.06333333333333332482 0.17703703703703702055 0.00000000000000000000
0.00000000000000000000 0.01333333333333333419 0.04333333333333333481
0.00000000000000000000 0.00000000000000000000 0.00000000000000000000
0.00000000000000000000)

$[P^t]_{5\bullet}$= (0.11626984126984130086 0.10643650793650796804 0.14210317460317462013
0.04600000000000000616 0.12889909297052154491 0.07931746031746032821
0.05031746031746031633 0.07716666666666668895 0.02646258503401360471
0.02142857142857142530 0.08389115646258503423 0.05184353741496598028
0.01931972789115646294 0.02408163265306122180 0.02170068027210884237
0.00476190476190476147)

$[P^t]_{6\bullet}$= (0.12022222222222223031 0.06788888888888890139 0.13855555555555557068
0.06533333333333334048 0.07931746031746032821 0.10711111111111111527
0.07377777777777777546 0.12500000000000002776 0.02253968253968253663
0.00555555555555555490 0.04831746031746031456 0.09765079365079364082
0.00476190476190476147 0.02809523809523809326 0.01031746031746031550
0.00555555555555555490)

$[P^t]_{7\bullet}$= (0.07736111111111111660 0.07111111111111111105 0.13256944444444443421
0.04750000000000000056 0.06289682539682539542 0.09222222222222221932
0.10618055555555555580 0.13770833333333332149 0.03025793650793650452
0.00694444444444444406 0.05248015873015873106 0.11977182539682540430
0.00595238095238095205 0.03720238095238094511 0.01289682539682539611
0.00694444444444444406)

$[P^t]_{8\bullet}$= (0.10622222222222223176 0.10516666666666668606 0.11566666666666666763
0.10622222222222223176 0.07716666666666666119 0.12500000000000002776
0.11016666666666667662 0.14588888888888890127 0.01500000000000000118
0.00000000000000000000 0.02500000000000000139 0.05350000000000000588
0.00000000000000000000 0.01500000000000000118 0.00000000000000000000
0.00000000000000000000)

$[P^t]_{9\bullet}$= (0.01031746031746031897 0.00476190476190476233 0.01726190476190476303
0.00000000000000000000 0.02205215419501133436 0.01878306878306878341
0.02017195767195767084 0.01250000000000000069 0.18858654572940286576
0.12433862433862433172 0.14903628117913830797 0.10121882086167799386
0.06689342403628116551 0.04771352985638699362 0.11980347694633408651
0.09656084656084655549)

$[P^t]_{10\bullet}$= (0.00000000000000000000 0.00000000000000000000 0.00000000000000000000
0.00000000000000000000 0.02678571428571428423 0.00694444444444444406
0.00694444444444444406 0.00000000000000000000 0.18650793650793651146
0.18055555555555555247 0.15178571428571427382 0.06150793650793651146
0.13789682539682540652 0.03373015873015872829 0.15873015873015872135
0.04861111111111110494)

$[P^t]_{11\bullet}$= (0.02503401360544217635 0.02170068027210884237 0.03812925170068027364
0.00571428571428571428 0.05992225461613215343 0.03451247165532879413
0.02998866213151927043 0.01785714285714285615 0.12774538386783282351
0.08673469387755100568 0.15213880790411399291 0.11039844509232261960
0.08196873987690311836 0.08239390994493032971 0.09656624554583735642
0.02919501133786847613)

$[P^t]_{12\bullet}$= (0.03615079365079364865 0.03226190476190476247 0.05351190476190476053
0.02166666666666666741 0.04320294784580498704 0.08137566137566137892
0.07984788359788358436 0.04458333333333333592 0.10121882086167799386
0.04100529100529100301 0.12879818594104308627 0.13278628117913832130
0.03415532879818593964 0.08997543461829175804 0.05234315948601662993
0.02711640211640211490)

$[P^t]_{13\bullet}$= (0.00952380952380952467 0.00952380952380952467 0.00952380952380952467
0.00000000000000000000 0.03219954648526077157 0.00793650793650793607
0.00793650793650793607 0.00000000000000000000 0.13378684807256233102
0.18386243386243383835 0.19126039304610731318 0.06831065759637186541
0.15197467876039300050 0.04648526077097504816 0.12585034013605439496
0.02182539682539682072)

$[P^t]_{14\bullet}$= (0.02063492063492063794 0.00952380952380952467 0.03452380952380952606
0.00000000000000000000 0.04013605442176870763 0.04682539682539682557
0.04960317460317460042 0.02500000000000000139 0.09542705971277398724
0.04497354497354497105 0.19225245653817080171 0.17995086923658351608
0.04648526077097504816 0.13775510204081631294 0.05971277399848827494
0.01719576719576719481)

$[P^t]_{15\bullet}$= (0.00714285714285714263 0.00714285714285714263 0.00714285714285714263
0.00000000000000000000 0.02712585034013605123 0.01289682539682539611
0.01289682539682539611 0.00000000000000000000 0.17970521541950112976
0.15873015873015872135 0.16899092970521539803 0.07851473922902493796
0.09438775510204081010 0.04478458049886620967 0.14597505668934240841
0.05456349206349206393)

$[P^t]_{16\bullet}$= (0.00000000000000000000 0.00000000000000000000 0.00000000000000000000
0.00000000000000000000 0.01190476190476190410 0.01388888888888888812
0.01388888888888888812 0.00000000000000000000 0.28968253968253965258
0.09722222222222220989 0.10218253968253966646 0.08134920634920633775
0.03273809523809523281 0.02579365079365079222 0.10912698412698411399
0.22222222222222220989)

Annexe B

Résultats

B.1 Lien département et numéro dans le programme

Numéro	Département
1	Philosophie
2	Langue et Littératures romanes
3	Langues modernes
4	Sciences de l'Antiquité
5	Sciences historiques
6	Médias, Culture et Communication
7	Droit
8	Science politique
9	Criminologie
10	Astrophysique, Géophysique et Océanographie
11	Biologie, Ecologie et Evolution
12	Chimie
13	Géographie
14	Géologie
15	Mathématique
16	Physique
17	Sciences de la vie
18	Sciences et gestion de l'environnement
19	Sciences biomédicales et précliniques
20	Sciences cliniques
21	Sciences dentaires
22	Sciences pharmaceutiques
23	Sciences de la motricité
24	Sciences de la Santé publique
25	Département d'Aérospatiale et Mécanique

Numéro	Département
26	ArGEnCo
27	Département de Chemical Engineering
28	Département d'Electricité, Electronique et Informatique (Montefiore)
29	Département clinique des animaux de compagnie et des équidés - DCA
30	Département clinique des animaux de production - DCP
31	Département des maladies infectieuses et parasitaires - DMI
32	Département de morphologie et Pathologie - DMP
33	Département de gestion vétérinaire des Ressources Animales - DRA
34	Département des sciences des denrées alimentaires - DDA
35	Département des sciences fonctionnelles - DSF
36	Psychologie
37	Logopédie
38	Sciences de l'Education
39	Sciences sociales
40	Agro-Bio Tech
41	HEC
42	Architecture
43	Services généraux
44	?

B.2 Résultat dix premières communautés : algorithme du chemin aléatoire

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1	1	2	3	0	0	1	2	0	1	16	14	98	1	2	6	14	135	5	184	780	6	150	10
2	0	0	0	0	0	0	0	0	0	0	14	77	0	0	0	0	22	2	2	4	1	0	0
3	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	2	12	1	9	15	0	2	0
4	0	0	0	0	0	1	0	0	0	0	0	8	0	0	0	5	24	0	8	167	0	3	2
5	0	1	2	0	0	0	0	1	0	0	0	0	0	0	0	0	3	0	3	51	1	2	138
6	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	35	0	38	143	1	1	0
7	0	1	0	0	0	0	0	0	0	0	0	3	0	0	0	0	37	0	42	112	1	23	7
8	0	0	0	0	0	0	0	0	0	0	0	101	0	0	0	2	13	0	1	0	0	0	0
9	0	0	0	0	0	0	2	0	0	0	4	10	0	0	0	0	72	0	2	4	0	0	1
10	0	0	0	0	0	0	0	0	0	0	0	56	0	0	0	6	0	0	1	0	1	0	0

24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	Total
44	3	13	0	3	2	1	2	16	0	1	4	20	0	22	3	8	4	6	35	19	1637
0	0	1	1	1	0	0	3	0	0	0	0	1	0	1	4	279	4	0	1	6	424
0	2	0	0	0	57	43	91	27	0	3	51	0	0	2	1	4	1	0	1	34	361
1	0	1	0	6	0	0	0	0	0	0	0	78	1	1	0	2	1	0	1	6	316
42	13	7	0	2	0	0	0	0	1	0	0	1	0	0	1	1	0	0	1	2	273
0	0	0	0	2	0	0	1	1	0	16	0	0	0	0	0	3	0	0	1	10	254
0	0	1	0	0	0	0	0	1	0	0	0	2	0	1	0	0	0	0	5	4	240
0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	120
1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0	2	2	103
0	0	1	25	0	0	0	0	0	0	0	0	0	0	0	0	4	1	1	2	4	102

B.3 Résultat de la méthode heuristique

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	51	75	90	48	138	49	233	111	5	259	189	458	97	127	29	159	554	240
2	1	0	1	0	0	0	1	1	13	0	0	0	0	0	0	0	3	0
3	0	0	0	0	0	0	0	0	0	0	3	50	0	0	0	0	2	0
4	0	0	0	0	4	0	0	0	0	0	0	0	11	0	1	0	0	1
5	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	38	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	0	0	0
7	0	0	0	0	13	0	0	0	0	0	0	0	0	0	0	0	0	0

19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
360	1710	72	261	171	134	319	328	34	263	145	103	110	48	1	56	65	161
3	65	0	1	36	1	0	0	0	0	0	1	0	0	0	0	0	112
1	4	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	1	0	0	0	0	1	19	0	0	0	0	0	0	0	0	0	2
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

37	38	39	40	41	42	43	44	Total
6	213	154	709	268	44	140	266	9053
26	8	14	0	1	1	9	7	305
0	0	0	59	1	0	4	0	125
0	0	1	0	0	21	1	0	63
0	0	0	0	0	1	0	0	42
0	0	1	0	4	0	1	1	36
0	0	0	0	0	0	0	0	13