Structure multi-échelle des grands graphes

Thomas Aynaud
encadré par Jean-Loup Guillaume et Matthieu Latapy

équipe Complex Networks, LIP6 - CNRS - Université Pierre et Marie Curie

Mars à septembre 2008

Réseaux

modélisés par des graphes

- Un réseau informatique : les nœuds étant les machines et les connexions les arêtes
- Le web : les pages étant les nœuds et les liens hypertextes les arêtes
- Interactions sociales : les nœuds représentant les personnes, et les arêtes leurs relations

Les graphes issues de données réelles seront appelés **graphes de terrain**

Plusieurs grandeurs associées

- La densité
- Le degré moyen et maximal
- La distribution de degré, $p(k)$ la probabilité pour un nœud d'être de degré k
- La transitivité et le clustering
- La distance moyenne

Des propriétés remarquables

- Faible densité
- Distribution de degrés hétérogène
- Fort clustering
- Faibles distances

Communautés dans les réseaux sociaux

Exemples

- Un réseau "à la Facebook" : les groupes d'amis
- Un réseau de chercheurs liés par co-publication : les groupes thématiques

A différentes échelles

- Langue
- Région
- Ville...

En informatique

Exemples

- Sur le web : sites ou pages traitant du même sujet
- Sur un réseau pair à pair : fichiers sur les mêmes thèmes

A différentes échelles

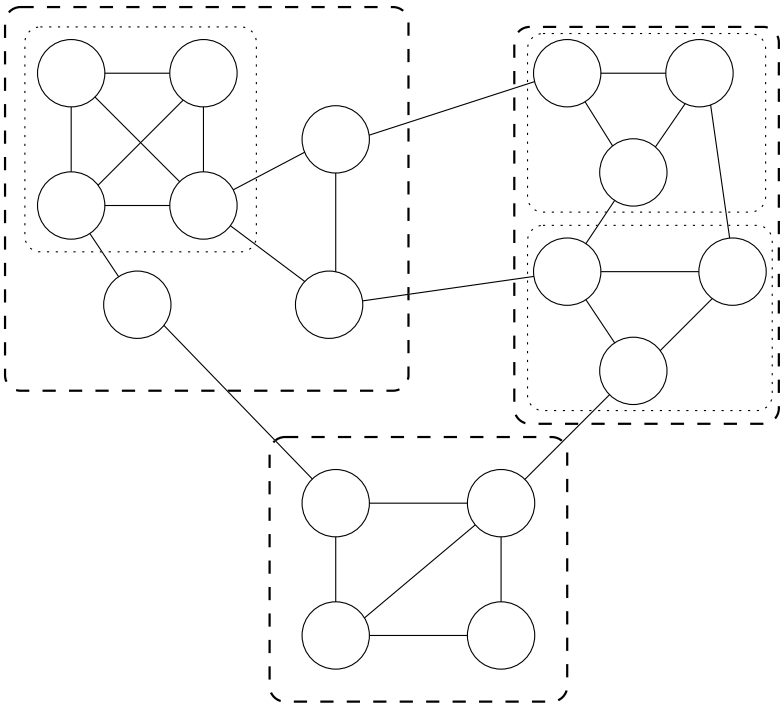
- Pages traitant de l'automobile
- Pages traitant d'une marque particulière
- Pages traitant d'un modèle précis

Il semble exister une structure communautaire multi-échelle

Objectifs généraux

- **Mieux comprendre cette structure**
- **L'utiliser pour dessiner les graphes**
- L'utiliser pour améliorer les algorithmes
- L'utiliser pour compresser les graphes

Intuition



Rechercher des parties denses avec peu de liens entre elles.

Vers une définition...

- Détecter les communautés revient à partitionner l'ensemble des nœuds
- On évalue la partition par une fonction de qualité à maximiser, en prenant en compte l'intuition précédente

Définition la plus acceptée

La modularité

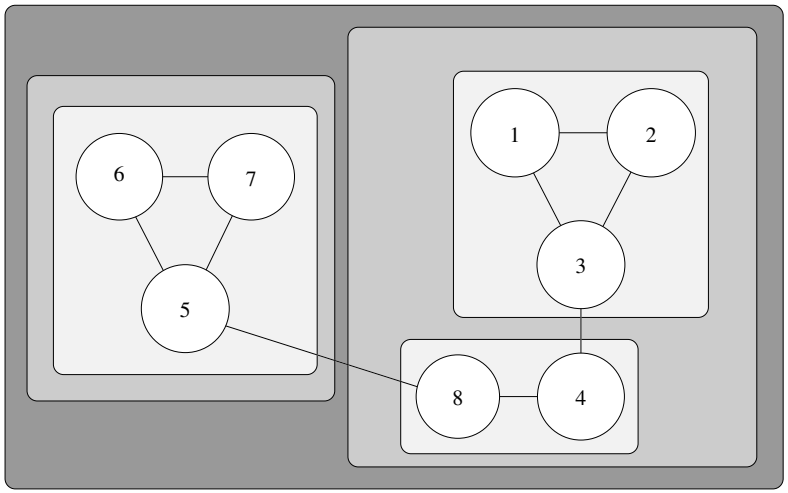
Compare la proportion d'arêtes à l'intérieur d'un sous ensemble avec celle qu'un sous ensemble aléatoire de même taille devrait avoir

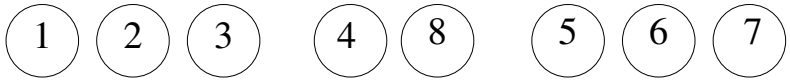
$$Q(\pi) = \sum_{s \in \pi} \left\{ \frac{l_s}{L} - \left(\frac{d_s}{2L} \right)^2 \right\}$$

- Comprise entre -1 et 1
- De moyenne 0 sur l'ensemble des partitions

M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Phys. Rev. E*, vol. 69, Feb 2004.

Deux grandes méthodes générales

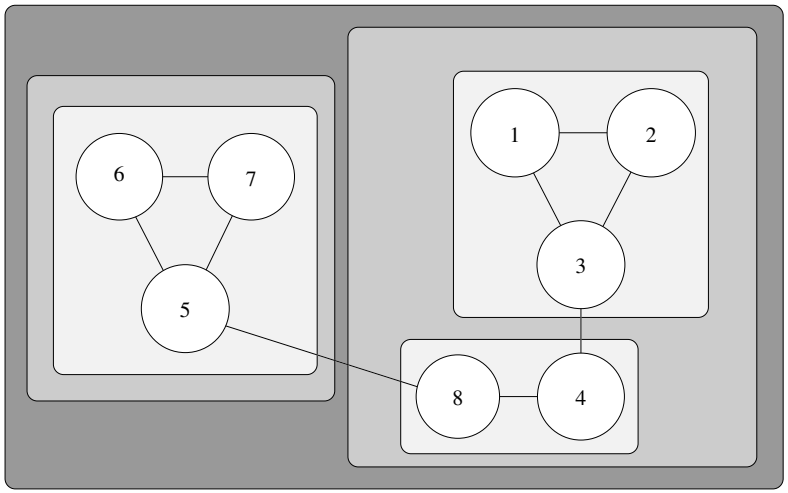


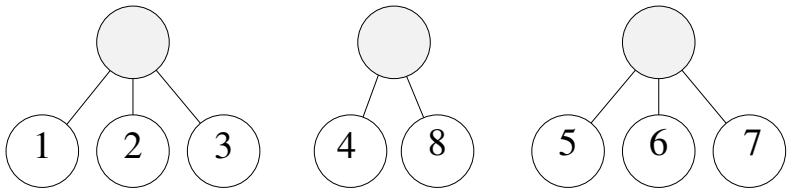


- Méthodes séparatives : de haut en bas
- **Méthodes agglomératives : de bas en haut**

Chacune aboutissant à la construction d'un arbre de feuilles les nœuds du graphe initial

Deux grandes méthodes générales

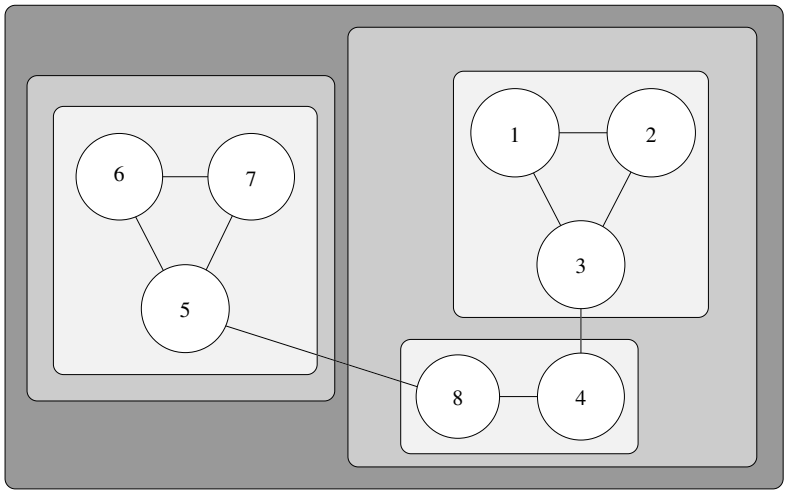


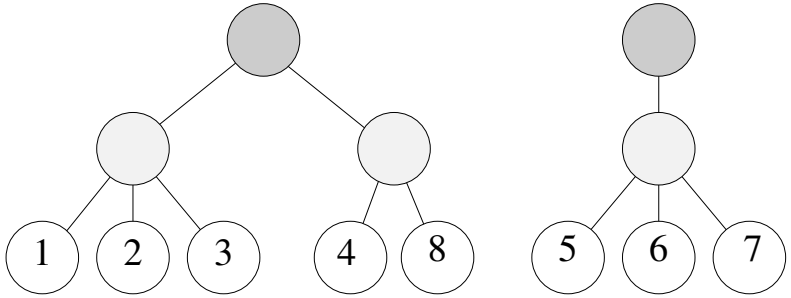


- Méthodes séparatives : de haut en bas
- **Méthodes agglomératives : de bas en haut**

Chacune aboutissant à la construction d'un arbre de feuilles les nœuds du graphe initial

Deux grandes méthodes générales

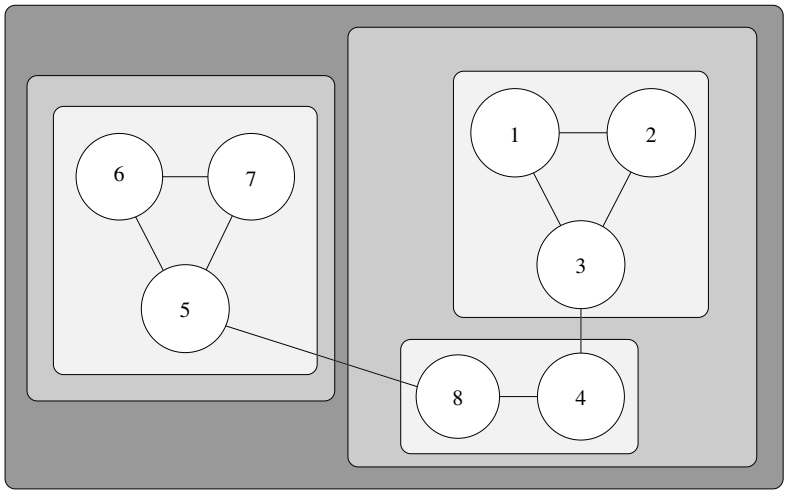


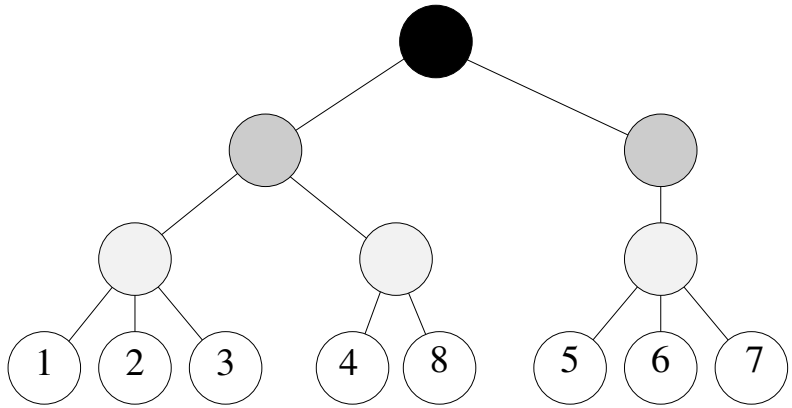


- Méthodes séparatives : de haut en bas
- **Méthodes agglomératives : de bas en haut**

Chacune aboutissant à la construction d'un arbre de feuilles les nœuds du graphe initial

Deux grandes méthodes générales

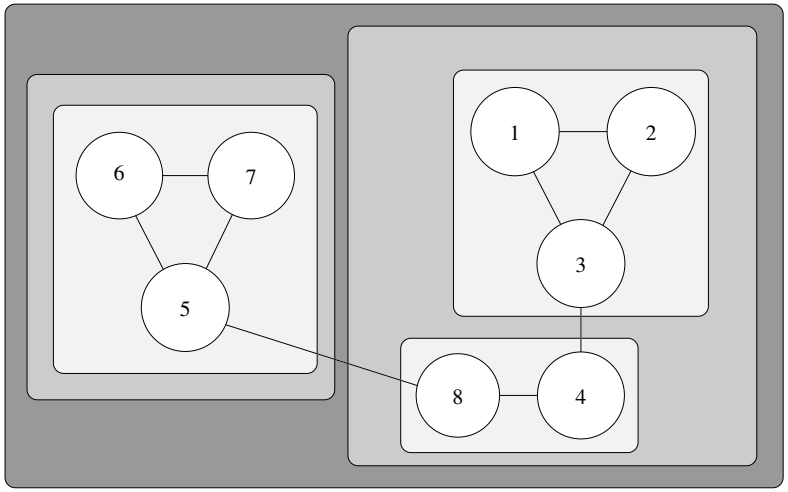


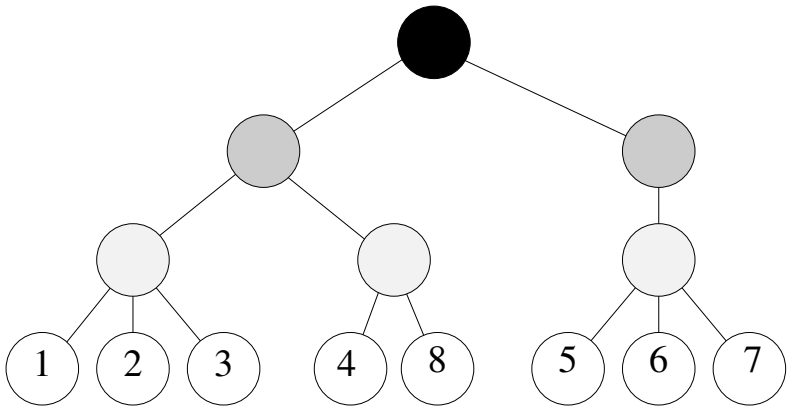


- Méthodes séparatives : de haut en bas
- Méthodes agglomératives : de bas en haut

Chacune aboutissant à la construction d'un arbre de feuilles les nœuds du graphe initial

Utilisation de cet arbre





- Chaque niveau de profondeur défini une partition
- On cherche celle qui maximise la modularité

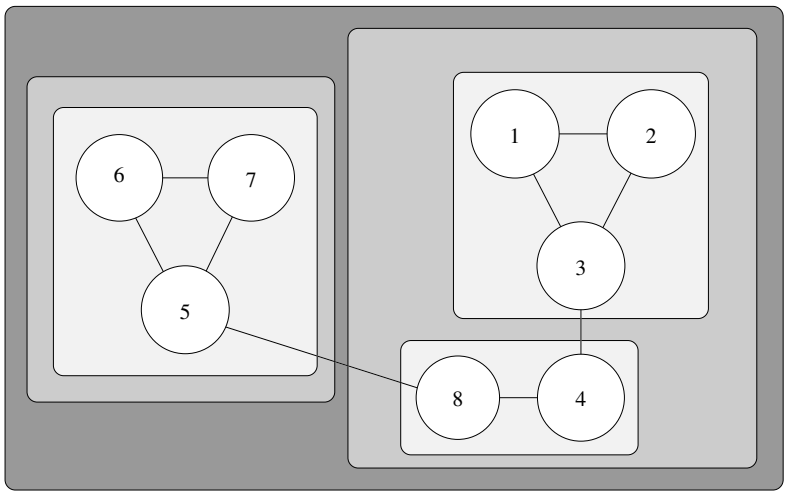
- Plusieurs heuristiques différentes pour décider quelles parties regrouper/séparer
- Utilisation d'une heuristique particulière, pour laquelle le niveau juste en dessous de la racine est toujours le meilleur.

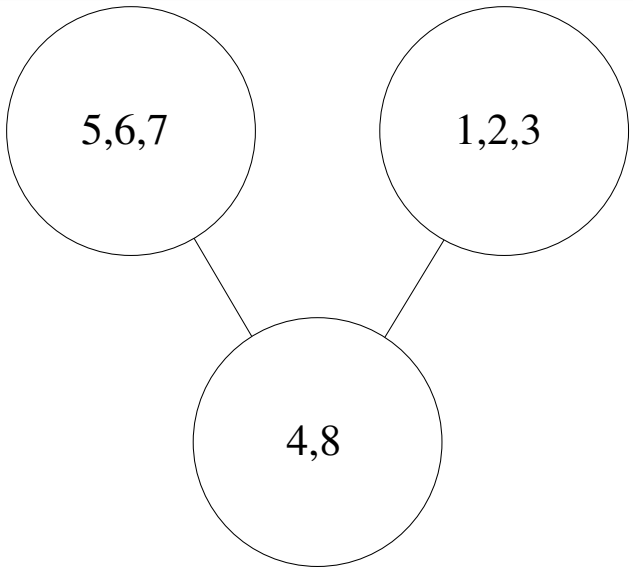
Graphes entre les communautés

Graphes entre communautés

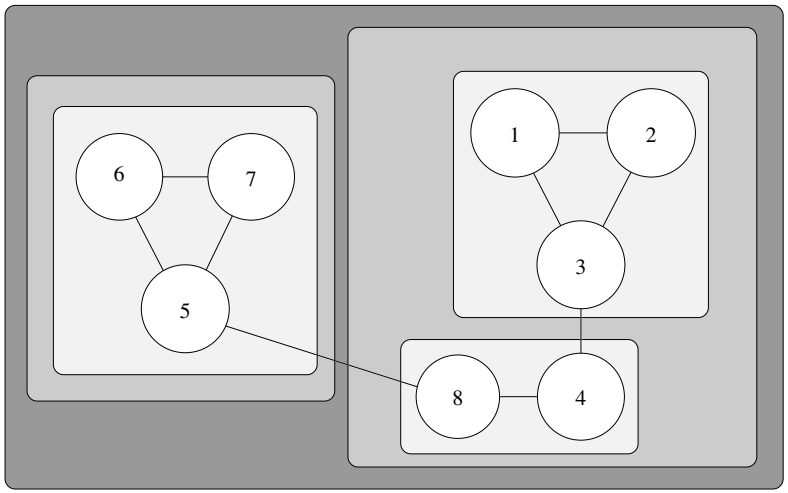
Pour chaque partition un graphe quotient :

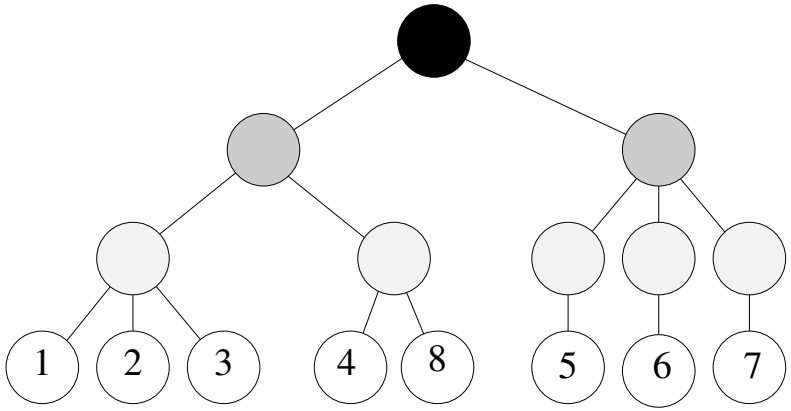
- Un nœud par partie
- Un lien entre deux nœuds s'il existe un lien entre des éléments des parties qu'ils représentent





Une autre approche





Calcul des communautés dans le graphe initial, extraction des sous-graphes correspondants et calcul récursif dans ces sous-graphes

- Une série de graphes entre ces communautés
- Une série de graphes à l'intérieur de communautés imbriquées

Méthode générale

- générer ces différentes séries de graphes à partir de plusieurs graphes initiaux
- chercher à comparer les deux partitions
- calculer sur chacun certaines grandeurs classiques d'étude des réseaux

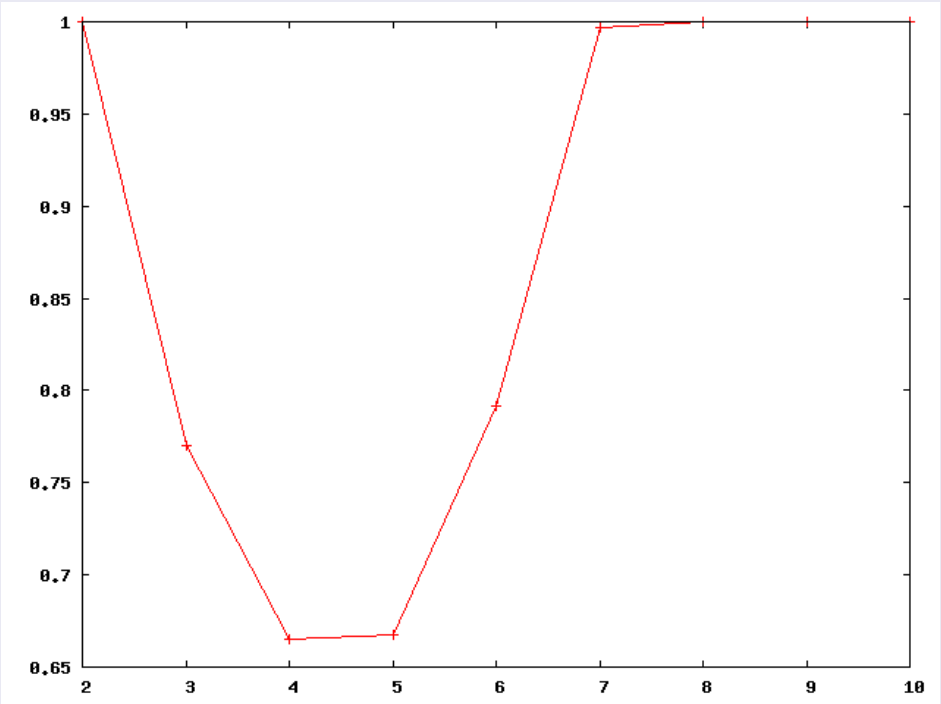
Graphes étudiés

- Clients pair à pair, 439000 nœuds et $3,8$ millions d'arêtes
- Fichiers pair à pair, 226000 nœuds et $1,7$ millions d'arêtes
- Arxiv, graphe de co-publication, 9400 nœuds et 24000 d'arêtes
- **Webndu**, graphe du web, 325000 nœuds et $1,1$ millions d'arêtes

Comparaison des partitions

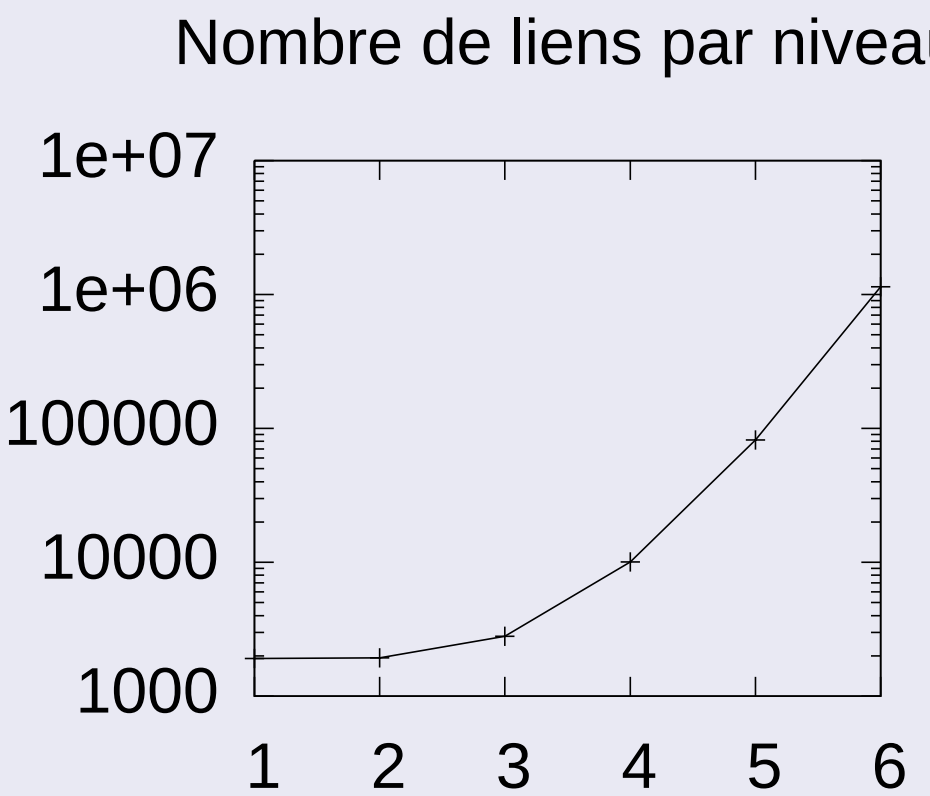
Calcul de **l'information mutuelle normalisée**, normalisation de l'information mutuelle de la théorie de l'information

Information mutuelle normalisée



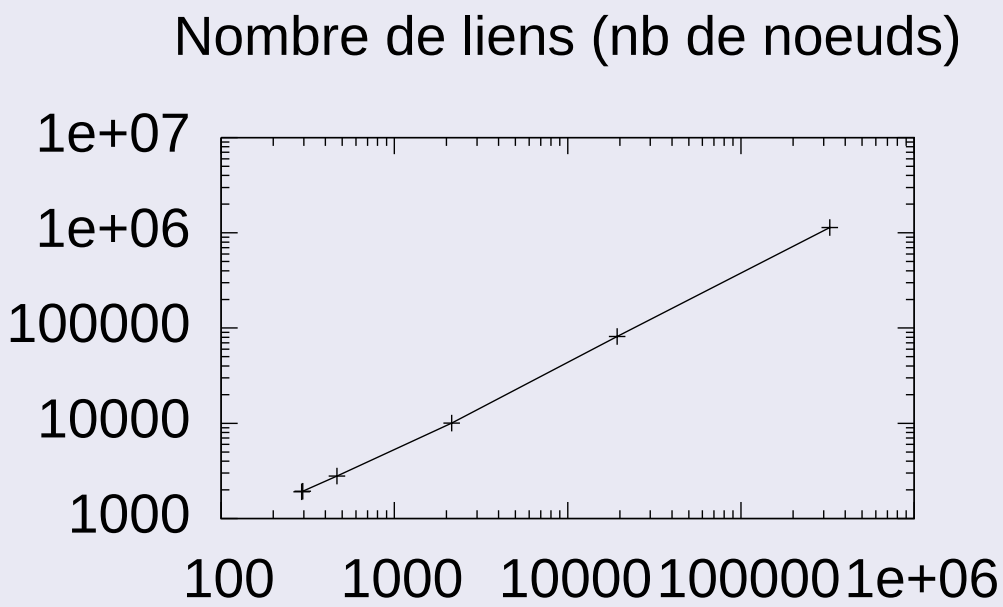
Le graphe entre les communautés

Nombre de liens par niveau



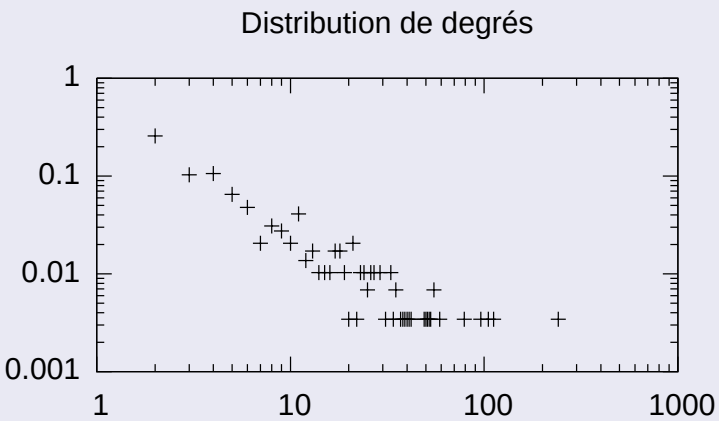
Le graphe entre les communautés

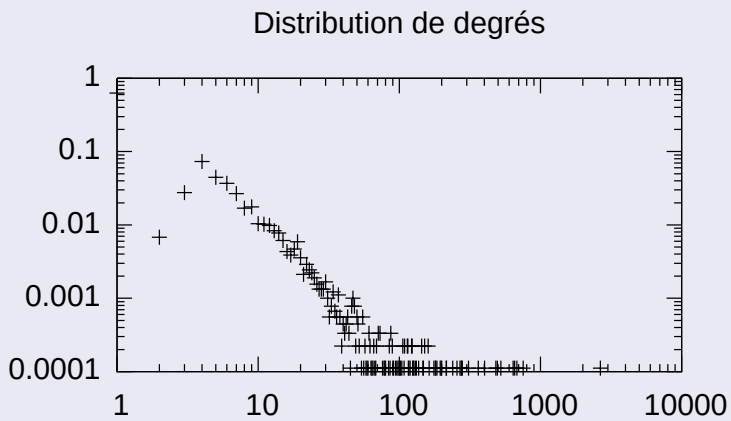
Nombre de liens en fonction du nombre de nœuds



Le graphe entre les communautés

Distribution de degrés





Le graphe entre les communautés

Scale-free

- Degré moyen et maximum très variables
- Distribution de degrés hétérogène
- Conservation du caractère *scale-free*

Petit monde

- Distance moyenne toujours très faible
- Transitivité et clustering sont croissants durant le déroulement de l'algorithme
- Conservation du caractère *petit monde*

Autres faits marquants

Avec la première approche

- Les grandeurs évoluent régulièrement
- Les graphes à l'intérieur des communautés évoluent peu, et se comportent comme des sous graphes denses.

Avec la deuxième approche

- Les résultats sur les graphes entre communautés dépendent beaucoup du graphe initial
- Les sous graphes extraits récursivement ont une très forte modularité, signe qu'il y a localement des communautés

Objectifs

- Ne pas dessiner tout le graphe
- Utiliser des outils de dessins simples et la décomposition en communautés pour schématiser le graphe

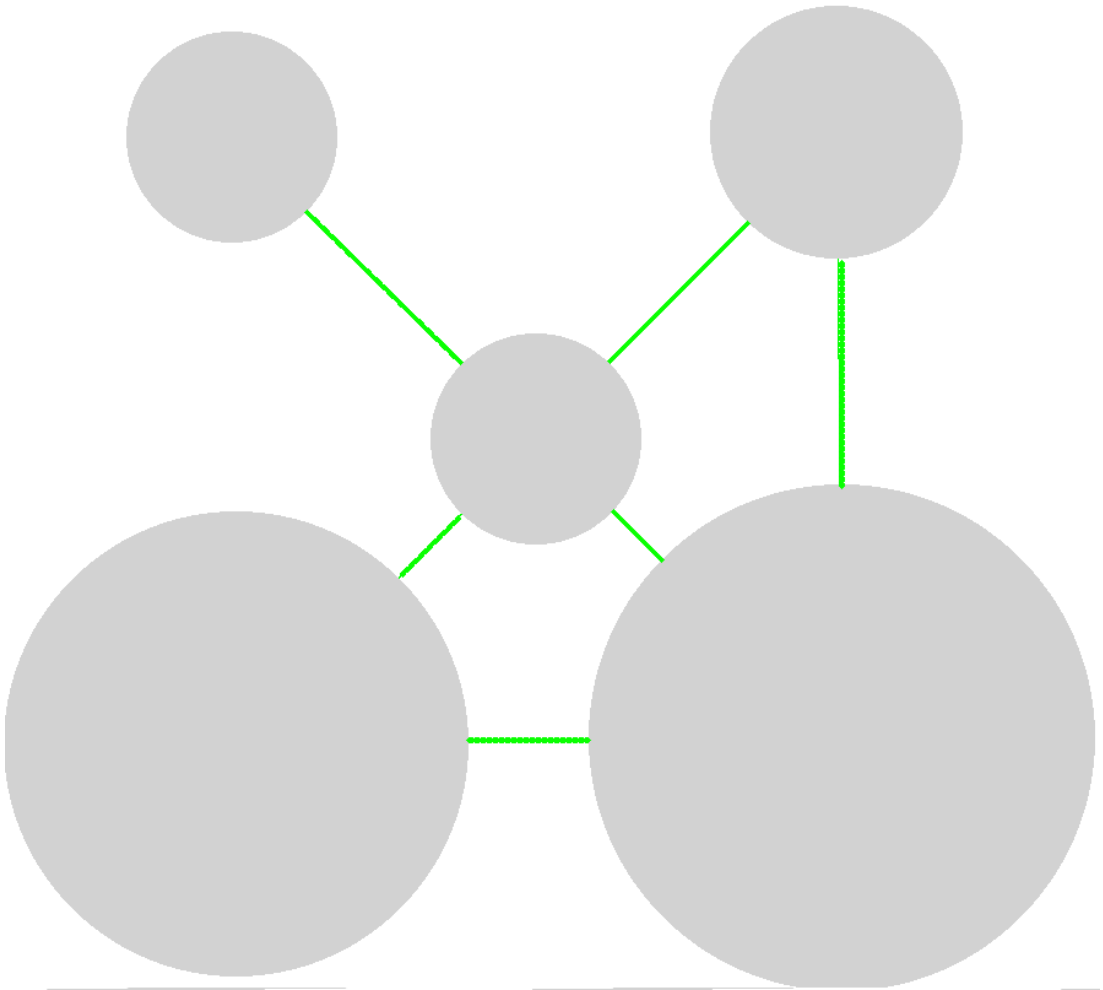
Principe : dessin récursif

- Représenter initialement le graphe entre communautés
- Quand on sélectionne une communauté, extraire le sous graphe correspondant et le représenter à l'intérieur du dessin de la communauté

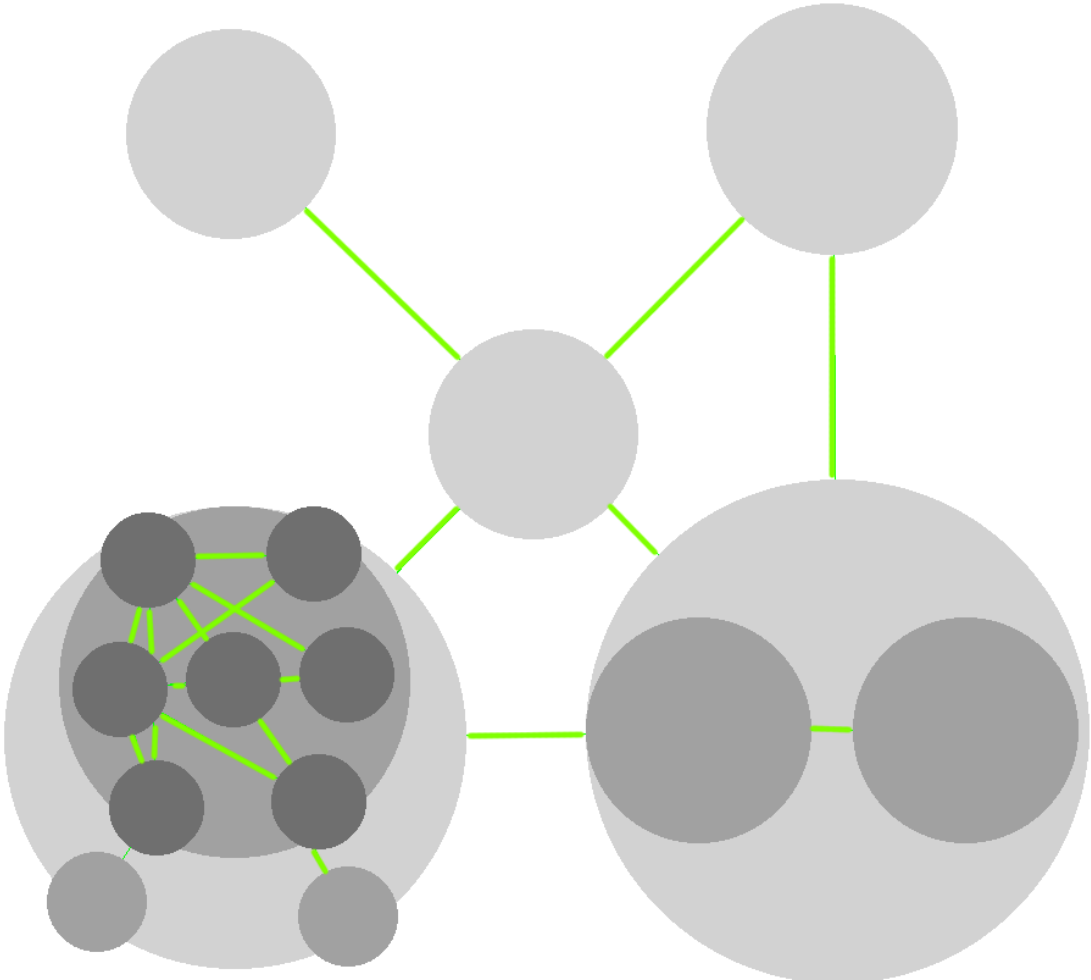
Réalisation

- Implémenté en C++
- Utilisant le même algorithme de calcul des communautés
- Utilisant Graphviz pour le placement des nœuds

Résultats obtenus

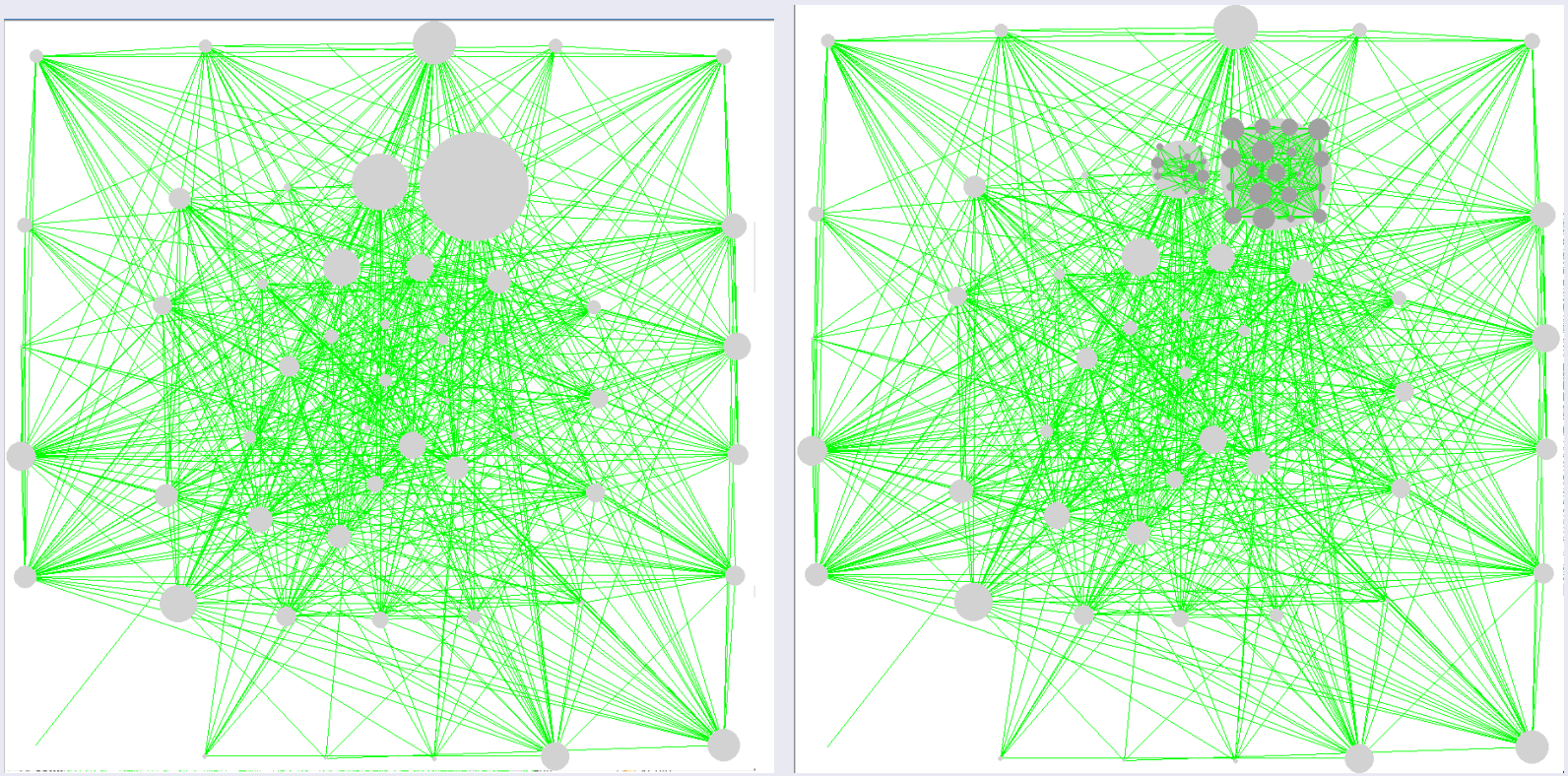


Résultats obtenus



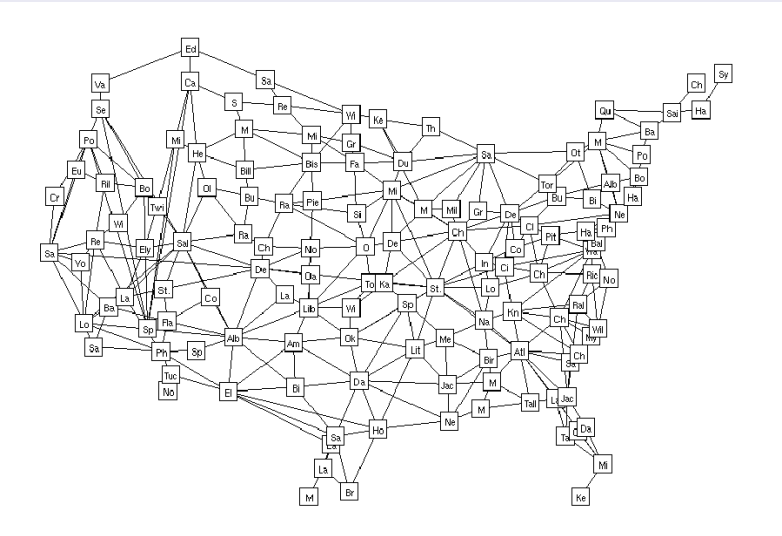
Graphes encore trop gros et très denses

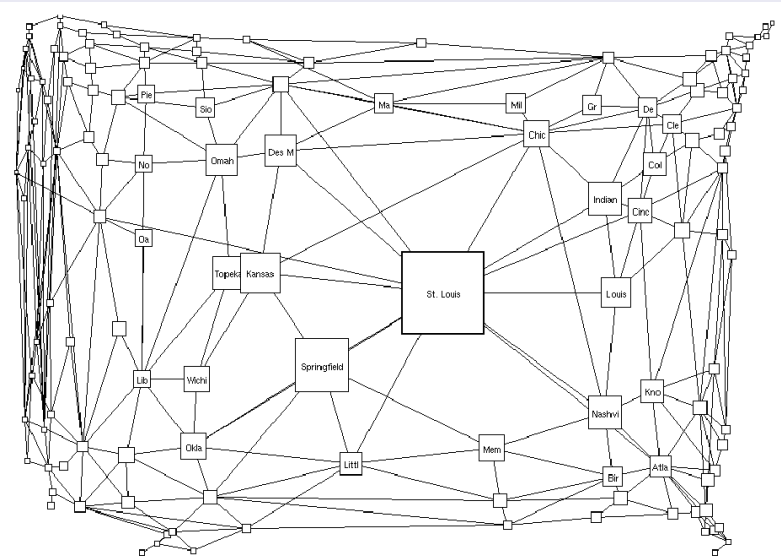
Problème



Graphes encore trop gros et très denses

Suite possible, rendu *oeil de poisson*

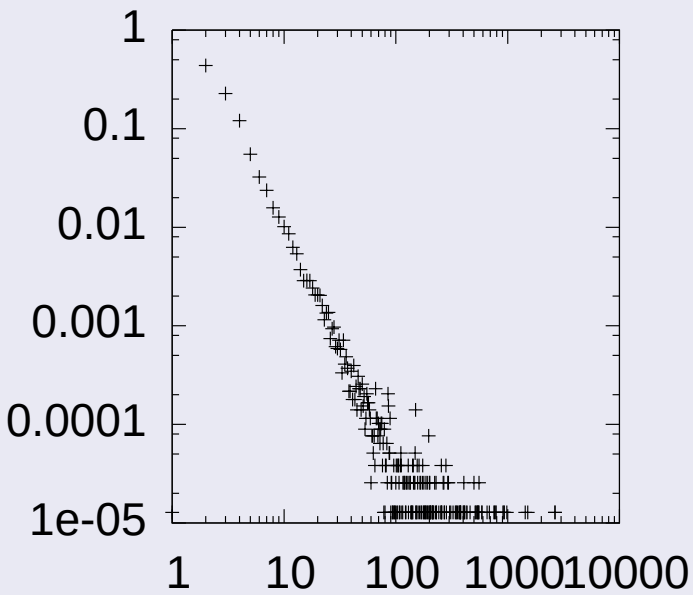




M. Sarkar and M. H. Brown, "Graphical fisheye views of graphs," in *CHI '92 : Proceedings of the SIGCHI conference on Human factors in computing systems*, (New York, NY, USA), pp. 83–91, ACM.

Distribution du nombre de nœuds à représenter

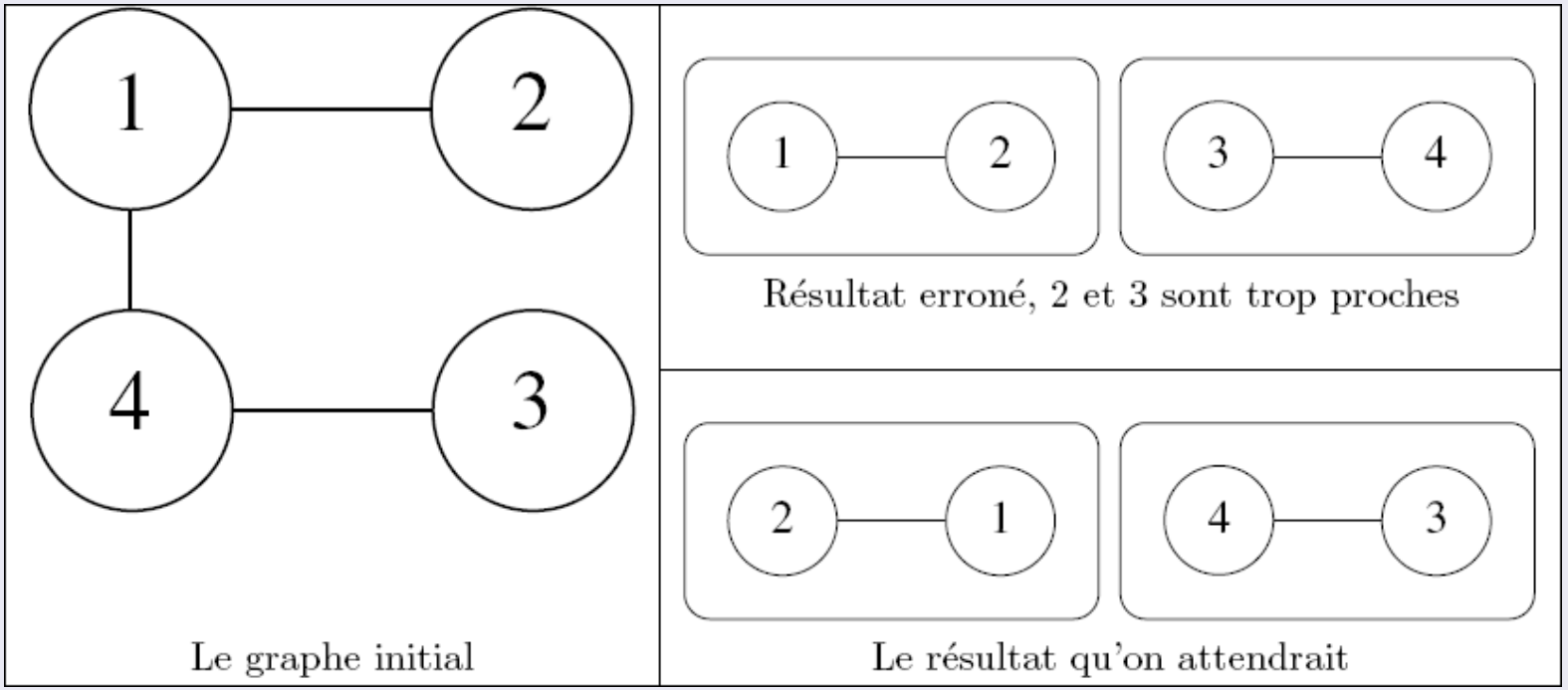
Problème



Suite possible

Analyse du moment où le nombre de nœuds explose, peut être qu'il n'y a plus de communautés ?

Propriétés fausses



Suite possible

Intégration des autres communautés dans le rendu, sous forme de forces exercées sur les nœuds

Deux objectifs

- Analyse de différentes échelles
- Utilisation pour le dessin de graphe

Analyse de différentes échelles

- Peu de résultats généraux
- On constate quand même une régularité

Dessin de graphe

- Beaucoup de questions encore ouvertes

Merci pour votre attention !

Comparaison des partitions

Calcul de l'information mutuelle entre les partitions, pour évaluer une distance, et de **l'information mutuelle normalisée**

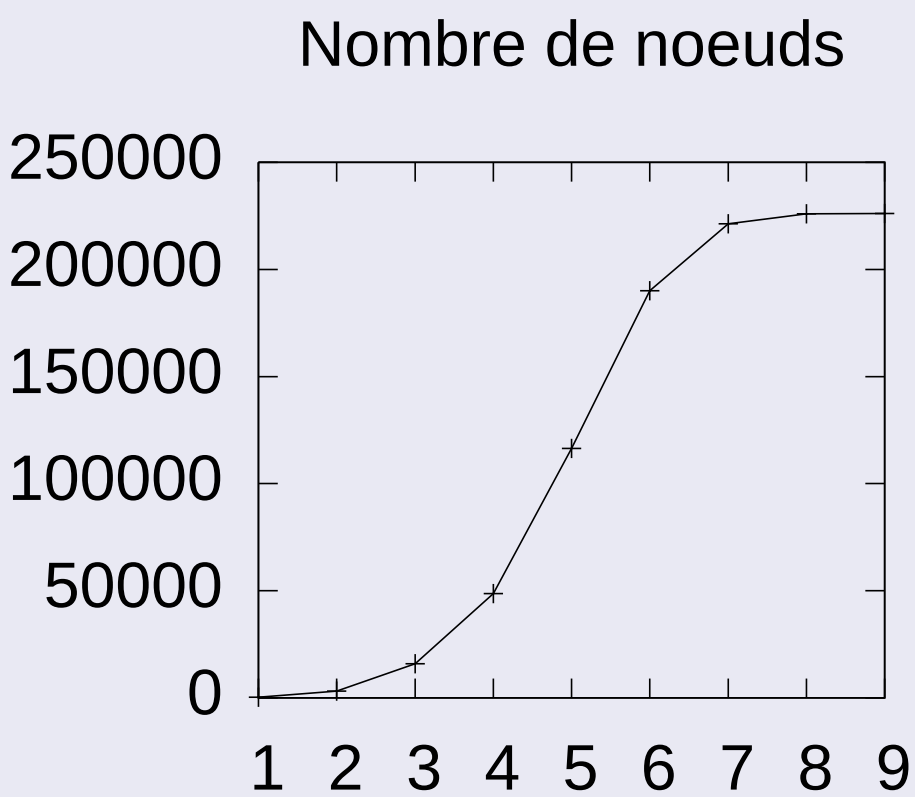
Information mutuelle

$$I(X,Y) = \sum_{x,y} P(x,y) \log \frac{P(x,y)}{P(x)\,P(y)}$$

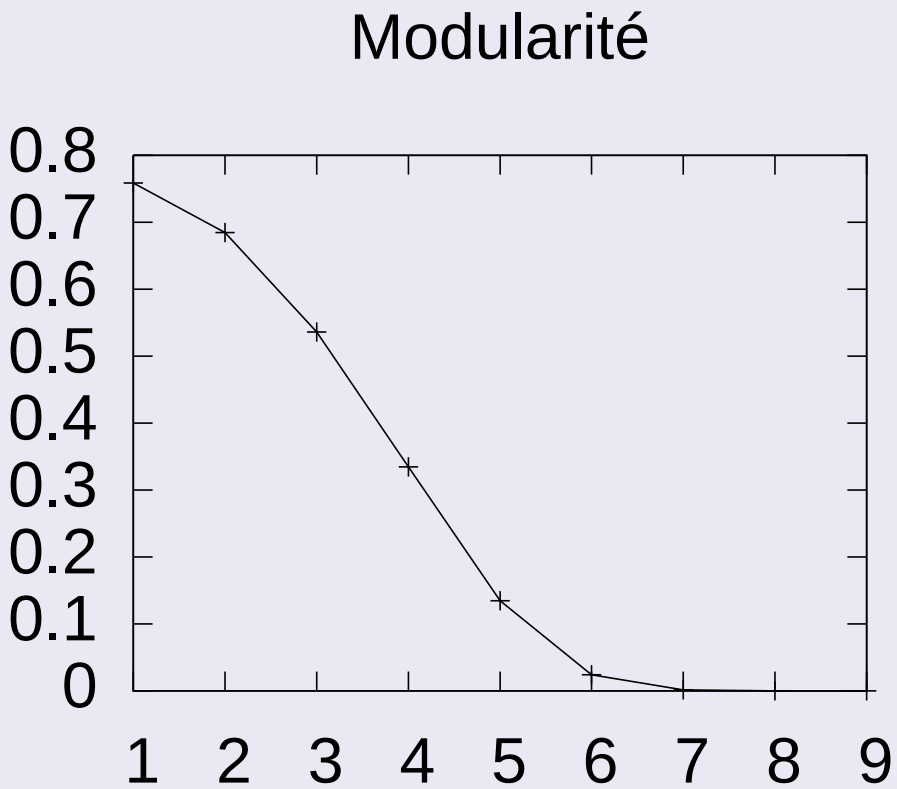
Information mutuelle normalisée

$$NMI(X,Y) = \frac{2I(X,Y)}{H(X) + H(Y)}$$

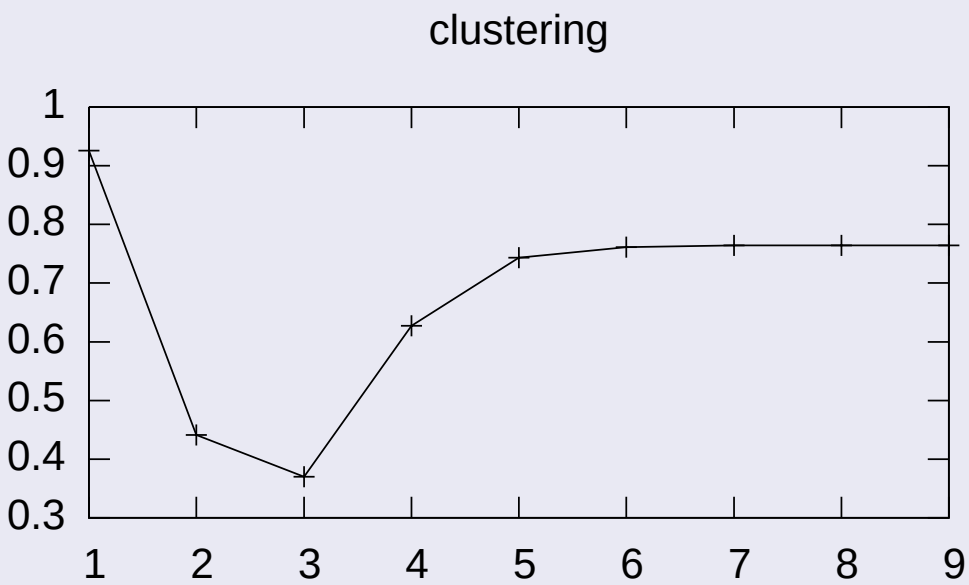
Nombre de nœuds des graphes extraits



Modularité des graphes extraits



Clustering du graphe extrait



Modularité du graphe extrait récursivement