Théorie des graphes

PROJET

Accessibilité et vulnérabilité des réseaux

Le projet de ce semestre va vous permettre de voir comment la théorie des graphes et certains des algorithmes classiques vus en cours peuvent être utilisés dans l’étude de l’accessibilité et de la vulnérabilité des réseaux, en particulier des réseaux de transport. Au niveau des territoires, de telles études sont indispensables tant pour améliorer l’accessibilité et l’offre au niveau des infrastructures (politiques de désenclavement de certains territoires, de décongestion du trafic …), que pour mener des politiques efficaces de réduction et de prévention des risques. Bien sûr, ce projet n’est qu’une introduction à cette problématique complexe. Dans un premier temps, vous travaillerez sur des modèles très simplifiés, et seuls les indicateurs les plus simples seront utilisés. Les exemples étudiés dans ce projet seront des réseaux connexes de transport routier ou ferré simplifiés supposés non orientés. Mais, comme je l’illustrerai tout au long de l’énoncé, les techniques d’analyse développées pourraient tout à fait être utilisées pour l’étude d’autres types de réseaux (réseaux sociologiques, réseaux écologiques …).

L’utilisation de la théorie des graphes en géographie des territoires et pour la prévention des risques est relativement récente. C’est un domaine de recherche en plein développement. Ce projet comprend donc une partie documentation non négligeable. Faites des recherches (le web foisonne d’informations sur le sujet), prenez des notes et notez vos références afin de pouvoir les restituer lors de la soutenance et dans votre rapport.

1. Problématique, concepts, outils

a. Centralité et accessibilité

La centralité d'un sommet dans un graphe renvoie à la place, l'importance de ce sommet par rapport aux autres sommets du graphe. Elle peut être estimée de plusieurs manières : nombre de voisins, proximité par rapport aux autres sommets, point de passage important… Selon la nature du réseau étudié, certains indicateurs sont plus significatifs et discriminants que d'autres. Il convient donc de les comparer.

On peut définir l'accessibilité d'un lieu « comme une mesure de la facilité de ce lieu à être rejoint par un acteur, une clientèle, une information ou un service. L'accessibilté exprime donc l'offre de possibilités de déplacement donnée par le système de transport / de communication pour atteindre une (ou plusieurs) localisation(s) … » [1].
« La notion d'accessibilité dans un graphe est fortement liée aux notions de centralité et de plus court chemin. Elle permet de rendre compte de la capacité à atteindre plus ou moins facilement un (ou des) sommets(s) d'un graphe depuis un sommet donné. » [2]

b. Les indices de centralité des sommets

Les indices de centralité d'un sommet que nous allons étudier sont les suivants :

- **La centralité de degré**

L'indicateur le plus simple pour estimer l'importance d'un sommet est tout simplement le degré du sommet. Il montre le potentiel de ce sommet pour établir des liens directs avec les autres sommets.
On peut obtenir un indice normalisé (pour le rendre insensible à la taille du réseau et pouvoir comparer structurellement des réseaux de tailles différentes), en divisant les degrés par $n - 1$ (degré maximal d'un sommet dans un graphe simple). La centralité de degré normalisée sera noté $C_D(s)$.

- **La centralité de vecteur propre**

On peut la voir comme une extension de la centralité de degré qui tient compte de l'importance des nœuds voisins. Elle évalue l'importance d'un nœud au sein du réseau en fonction de l'importance de ses voisins. L'idée sous-jacente est qu'un lien avec un sommet peu connecté « vaut » moins qu'un lien avec un sommet très connecté. Cet indicateur est fréquemment utilisé avec les réseaux sociologiques par exemple pour

l'étude de la transmission des virus : il est certainement plus pertinent de regarder « combien de personnes fréquentent les personnes que l'on rencontre soi-même » [3]. On le notera $C_{VP}(s)$.
L'algorithme pour calculer les centralités de vecteur propre de chaque sommet est itératif puisqu'ils se calculent de proche en proche …

- Initialisation :
  Pour chaque sommet, attribuer la valeur 1 à l'indice :
  $$\forall i, C_{VP}(s_i) = 1$$
- Faire :
  - Pour chaque sommet s_i, faire la somme des indices des ses voisins :
    $$\forall i, c(s_i) = \sum_{s_j \epsilon\, Succ(s_i)} C_{VP}(s_j)$$
  - Calculer
    $$\lambda = \sqrt{\sum_i c(s_i)^2}$$
  - Pour chaque sommet, recalculer l'indice :
    $$\forall i, C_{VP}(s_i) = \frac{c(s_i)}{\lambda}$$

  Tant que (λ « varie trop »)

A la première itération l'indice correspond tout simplement au degré. Puis à chaque itération successive, il tient compte des voisinages plus éloignés …

Pour information : le PageRank de Google qui mesure la popularité d'une page web pour son classement dans le moteur de recherche est une variante de cet indice.

- **La centralité de proximité**

Elle évalue l'importance d'un sommet relativement aux autres sommets du graphe en estimant la distance moyenne à parcourir pour aller de ce sommet à n'importe quel autre sommet du graphe. C'est une mesure « d'éloignement moyen ». « En matière de transmission virale, l'indice de centralité de proximité est représentatif de l'efficacité de propagation d'une maladie depuis un individu à travers tout le réseau social » [4].

La mesure utilisée pour estimer « la distance » entre deux sommets s_i et s_j est la **longueur** $d(s_i, s_j)$ **du plus court chemin** les reliant.
La centralité de proximité d'un sommet s est égale à l'inverse de la somme des longueurs des plus courts chemins de s aux autres sommets du graphe. On peut la normaliser en la multipliant par $n-1$. On la notera $C_P(s)$.

$$C_P(s_i) = \frac{n-1}{\sum_{j \neq i} d(s_i, s_j)} \quad (1)$$

Les longueurs des plus courts chemins peuvent être calculées avec l'algorithme de Dijkstra.

- **La centralité d'intermédiarité**

La centralité d'intermédiarité correspond à la fréquence avec laquelle un sommet se trouve sur les plus courts chemins reliant deux autres sommets quelconques du graphe. « Un niveau élevé de centralité d'intermédiarité n'est pas forcement corrélé avec un degré important du sommet : un nœud avec un faible degré faisant le lien entre deux groupes de sommets aura une centralité d'intermédiarité élevée. Les nœuds avec une forte centralité d'intermédiarité sont des points de passages importants pour relier rapidement deux sommets du graphes » [2]. On peut voir cet indice comme une mesure de l'utilité, de l'indispensabilité d'un sommet.
Notez que cette mesure peut aussi être appliquée aux arêtes du graphe.

La centralité d'intermédiarité d'un sommet s_i est donnée par :

$$\sum_{j<k} \frac{n\_pcc_{jk}(i)}{n\_pcc_{jk}}$$

où $n\_pcc_{jk}(i)$ est le nombre de plus court chemin allant de s_j à s_k passant par s_i et n_pcc_{jk} est le nombre total de plus courts chemins allant de s_j à s_k.

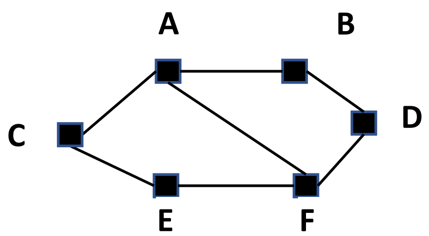
Pour que cet indice ne dépende pas de la taille du réseau, on peut le normaliser en le divisant par sa valeur maximale possible qui serait atteinte pour un sommet au milieu d'un graphe en étoile, à savoir $\frac{1}{2} C_{n-1}^2 = \frac{n^2-3n+2}{2}$.
L'indice de centralité normalisé qu'on notera $C_I(s)$ est ainsi donné par :

$$C_I(s_i) = \frac{2 \sum_{j<k} \frac{n\_pcc_{jk}(i)}{n\_pcc_{jk}}}{n^2 - 3n + 2} \quad (2)$$

Si quelques soient deux sommets du graphe, il n'existe qu'un seul plus court chemin allant de l'un à l'autre, l'algorithme de calcul des centralités intermédiaires ne pose pas de difficulté particulière. $n\_pcc_{jk}(i)$ vaut 1 si s_i se trouve sur le plus court chemin allant de s_j à s_k, 0 sinon ; et n_pcc_{jk} vaut toujours 1. Pour trouver les plus courts

chemins on utilise l'algorithme de Dijkstra et en remontant de prédécesseur en prédécesseur on peut déterminer si un sommet appartient ou non aux chemins trouvés. En revanche s'il existe plusieurs plus courts chemins reliant deux mêmes sommets, le calcul est plus compliqué, car l'algorithme de Dijkstra ne donne qu'un des plus courts chemins … Or un sommet qui n'est pas situé sur tous les plus courts chemins perd de son « pouvoir » puisqu'il existe des plus courts chemins alternatifs.
Dans ce cas, pour trouver les centralités intermédiaires, on peut adapter et modifier légèrement l'algorithme de Dijkstra pour qu'il compte également le nombre de plus courts chemins entre le sommet initial et les autres sommets. (Réfléchissez : que vaut le nombre de plus courts chemins d'un sommet quand on met à jour sa distance par rapport au nombre de plus courts chemins de son nouveau prédécesseur ? que vaut le nombre de plus courts chemins d'un sommet si on le « voit » avec une distance égale à celle qu'on avait déjà trouvé ?) En relançant Dijkstra sur tous les sommets, vous obtiendrez donc pour chaque paire de sommets (s_j, s_k), le nombre n_pcc_{jk} de plus courts chemins allant de s_j à s_k.
Ensuite, il faut trouver l'astuce pour savoir si un sommet donné s_i se trouve sur au moins un des plus courts chemins reliant deux autres sommets et dans ce cas déterminer sur combien.

Explication avec l'exemple ci-contre :



Calcul de la centralité intermédiaire du sommet A :

- Dijkstra (équivalent à un BFS dans le cas non orienté) adapté nous dira que la longueur du plus court chemin allant de B à C est $d(B,C) = 2$ et que le nombre de plus courts chemins allant de B à C est $n\_pcc_{BC} = 1$. Il nous dira aussi que $d(B,A)$=1 et $d(A,C)$=1. $d(B,A) + d(A,C)$=2 : en passant par A on peut emprunter un chemin de longueur 2 pour aller de B à C . Donc A se trouve sur le plus court chemin allant de B à C. $n_{pcc_{BC}}(A) = 1$
  Il faut donc ajouter 1/1=1 à la centralité intermédiaire de A
- Dijkstra adapté nous dira que $d(B,D) = 1$ et $n\_pcc_{BD} = 1$. Il nous dira aussi que $d(B,A)$=1 et $d(A,D)$=2. $d(B,A) + d(A,D)$=3, donc A ne peut pas se trouver sur un plus court chemin allant de B à D.
  A cette étape on n'ajoute rien (0) à la centralité intermédiaire de A
- Dijkstra adapté nous dira que $d(B,E) = 3$ et $n\_pcc_{BE} = 3$. Il nous dira aussi que $d(B,A)$=1 et $d(A,E)$=2. $d(B,A) + d(A,E)$=3, donc A se trouve sur au moins 1 des plus courts chemins allant de B à E.
  Il y a 1 plus court chemin allant de B à A ($B \to A$) ; il y a 2 plus courts chemins allant de A à E ($A \to F \to E$ et $A \to C \to E$) … il y a au total 2 plus courts chemins allant de B à E passant par A ($B \to A \to F \to E$ et $B \to A \to C \to E$) : $n_{pcc_{BE}}(A) = 2$
  Il faut donc ajouter 2/3 à la centralité intermédiaire de A
  - …. Et il faut recommencer ainsi pour chaque autre paire de sommets distincts de A : $(B,F), (C,D), (C,E)$ …

Avec cet approche, vous n'obtiendrez pas l'algorithme le plus efficace. L'algorithme de Brandes [5] est plus performant.

c. Vulnérabilité

Les réseaux d'infrastructures et en particulier les réseaux de transport sont soumis à des risques. Ils sont vulnérables et peuvent être endommagés par des catastrophes naturelles ou par des accidents, des erreurs humaines. Les dommages provoqués concernent non seulement l'infrastructure elle-même (dommages matériels) mais ils mettent aussi en péril les fonctions assurées par le réseau qui risque de ne plus remplir correctement son rôle (dégradations et perturbations fonctionnelles).

De l'analyse des risques pouvant dégrader un réseau à l'évaluation des dommages potentiellement provoqués, l'étude de la vulnérabilité est complexe et doit tenir compte de nombreux facteurs. Nous n'en proposons ici qu'un début d'approche très superficielle. La question à laquelle vous tenterez de répondre est la suivante : sur un réseau routier, que se passe-t-il si un tronçon de route devient inutilisable (suite à une inondation, un éboulement …) ?

La première étape sera bien sûr de regarder l'impact de la suppression d'une arête sur la connexité : le réseau reste-t-il connexe ou certaines zones se retrouvent-elles isolées ? Ensuite vous étudierez son effet sur les indices de centralités détaillés précédemment. Comment la suppression d'une arête modifie-t-elle ces indices ? Dans quelle mesure certains sommets ayant des indices de centralité peu élevés en temps normal, se révèlent-ils essentiels pour garantir la résistance du réseau ?

2. Travail à faire

- ***Représentations – Chargements des graphes (3 points)***

Après avoir attentivement lu l'intégralité de l'énoncé :

– **Choisissez le modèle de structures de données qui vous semble le plus adapté et faites le digramme de classes correspondant.**

– **Mettez en place le chargement des fichiers décrivant les graphes**.

Les formats de ces fichiers sont imposés, votre code devra pouvoir lire des fichiers qui seront spécifiquement conçus pour la soutenance (que vous n'aurez pas encore vus). La *topologie* d'un réseau (quels sommets, quelles arêtes) et ses *pondérations* seront décrites séparément dans des fichiers distincts. Cette démarche permet de pouvoir étudier une même topologie **sans ou avec pondérations** et dans ce cas de pouvoir l'étudier selon des systèmes de pondérations différents (on veut pouvoir éventuellement changer le système de pondérations pour étudier nos réseaux routiers selon différents critères : la distance ou la durée des trajets entre les sommets, les flux transitant par les différents axes …).

Explication du format de fichier de topologie :

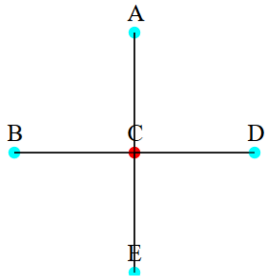
Les sommets sont repérés par un indice à partir de 0. Ils possèdent aussi un nom et des coordonnées x,y (qui peuvent servir pour dessiner les graphes).
Les arêtes sont repérées par un indice à partir de 0. Pour chaque arête on note l'indice de ses extrémités. [On prévoira de pouvoir étendre l'étude au cas des graphes orientés. Dans ce cas les indices des extrémités devront toujours être écrits dans l'ordre extrémité de départ – extrémité d'arrivée.]

Dans le fichier sont donc indiquées dans l'ordre suivant, sur des lignes distinctes, les informations suivantes :

– L'orientation (0 : non orienté, 1 : orienté)
– Le nombre de sommets (ordre du graphe)
– Pour chaque sommet (une ligne par sommet) : son indice, son nom, ses coordonnées x,y
– Le nombre d'arêtes (taille du graphe)
– Pour chaque arête (une ligne par arête) : son indice, les numéros de ses extrémités

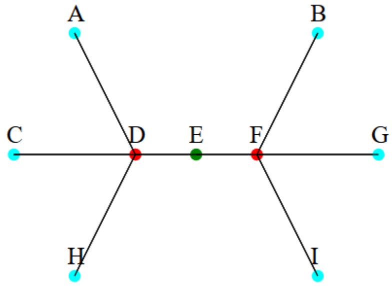
Exemples de fichier de topologie pour tester vos algorithmes et étudier les différents indices de centralité :

```
0
5
0 A 2 1
1 B 1 2
2 C 2 2
3 D 3 2
4 E 2 3
4
0 0 2
1 1 2
2 2 3
3 2 4
```



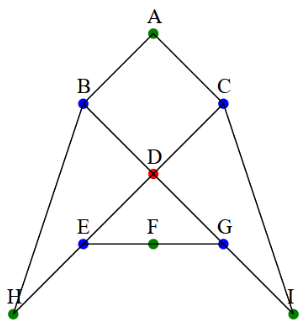
graphe_etoile1_topo.txt

```
0
9
0 A 2 1
1 B 6 1
2 C 1 2
3 D 3 2
4 E 4 2
5 F 5 2
6 G 7 2
7 H 2 3
8 I 6 3
8
0 0 3
1 1 5
2 2 3
3 3 4
4 4 5
5 5 6
6 7 3
7 8 5
```



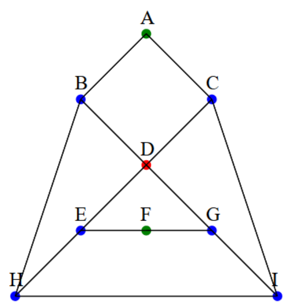
graphe_etoile3_topo.txt

```
0
9
0 A 3 1
1 B 2 2
2 C 4 2
3 D 3 3
4 E 2 4
5 F 3 4
6 G 4 4
7 H 1 5
8 I 5 5
12
0 0 1
1 0 2
2 1 3
3 2 3
4 3 4
5 3 6
6 1 7
7 2 8
8 4 5
9 5 6
10 6 8
11 4 7
```



graphe_cycle4_topo.txt

```
0
9
0 A 3 1
1 B 2 2
2 C 4 2
3 D 3 3
4 E 2 4
5 F 3 4
6 G 4 4
7 H 1 5
8 I 5 5
13
0 0 1
1 0 2
2 1 3
3 2 3
4 3 4
5 3 6
6 1 7
7 2 8
8 4 5
9 5 6
10 6 8
11 4 7
12 7 8
```



graphe_cycle5_topo.txt

Explication du format de fichier de pondérations :

Les arêtes sont repérées par leur indice à partir de 0. Les poids sont des nombres réels.

Dans le fichier sont donc indiquées dans l'ordre suivant, sur des lignes distinctes, les informations suivantes :

- Le nombre d'arêtes (taille du graphe)
- Pour chaque arête (une ligne par arête) : son indice suivi de son poids.

- ***Visualisation des graphes (1 point)***

La topologie du graphe chargé devra pouvoir être visualisée à partir d'un fichier au format svg (voir les TPs et le projet C++ du 1er semestre : vous pouvez réutiliser la classe Svgfile). Pensez à transformer les coordonnées des sommets pour tenir compte de la largeur et de la hauteur de la zone d'affichage.

- ***Calculs, comparaisons et analyses des indices de centralité des sommets (8 points)***

  - **Implémentez des méthodes pour calculer les indices (non normalisés et normalisés) de centralité de degré, de vecteur propre et de proximité présentés dans la partie 1.**

  - **Pour chaque algorithme, étudiez sa complexité. Dans votre rapport vous indiquerez dans un tableau la complexité de vos différents algorithmes.**

  - **Les résultats devront être :**
    - **affichés en console**
    - **sauvegardés dans un fichier texte**

    (format : 1 ligne par sommet et pour chaque sommet son indice suivi des valeurs des différents indices de centralité).

    Pour chaque indice de centralité, afficher et sauvegarder bien les deux valeurs non normalisées et normalisées. Les valeurs non normalisées vous permettront de vérifier vos résultats sur des petits graphes en faisant les calculs à la main.

- **Analyse et calcul de l'indice de centralité d'intermédiarité :**

  - **Concevez et implémentez un algorithme de calcul en supposant qu'il n'existe qu'un seul plus court chemin reliant chaque paire de sommets.**

  - **Faîtes le tourner sur un petit graphe dans lequel il existe plusieurs plus courts chemins entre différentes paires de sommets, et analysez les résultats obtenus : comparez les valeurs d'indices obtenues avec celles que donne la formule (2) (travail papier) pour mettre en évidence le biais introduit.**

  - **Trouvez et implémentez un algorithme correct pour calculer les indices de centralités intermédiaires dans le cas où plusieurs plus courts chemins peuvent exister entre deux mêmes sommets.**

  - **Etudiez la complexité de votre algorithme**

- **Comparaisons et analyses des différents indices de centralité :**
  - **Analyser vos résultats et comparer les différents indices de centralité obtenus.**
  - **Faites-le sur des graphes de topologie différente.**
  - **Que pouvez-vous en déduire sur la qualité et la signification de ces indices ? Les tableaux comparatifs et votre analyse des résultats devra figurer dans votre rapport.**

- ***Premier pas vers l'étude de la vulnérabilité (3 points)***

  - **Implémentez une méthode pour pouvoir tester la connexité (suite à la suppression d'une ou plusieurs arêtes)**
  - **Implémentez une méthode permettant de comparer les indices de centralité avant et après la suppression d'arêtes (calcul des différences). Les résultats devront être affichés en console.**

- ***Le programme final***

  Il devra donc au moins comporter un menu proposant de :
  - charger un graphe (donc le fichier de topologie et éventuellement un fichier de pondérations) en demandant les noms de fichiers à l'utilisateur.
  - changer le système de pondérations d'un graphe (en chargeant un nouveau fichier de pondérations)

- calculer, afficher et sauvegarder les différents indices de centralité.
- tester la vulnérabilité du graphe en :
  - permettant de supprimer une ou plusieurs arêtes choisies par l'utilisateur
  - testant la connexité du graphe suite à la suppression de ces arêtes
  - recalculant les indices de centralités pour les comparer avec ceux obtenus avant la suppression.

- ***Extensions (5 points)***

Seront retenus comme « extensions » et valorisés en fonction du niveau de difficultés estimé et de pertinence par rapport au sujet :

- Des améliorations en terme d'affichage : pouvoir visualiser les résultats au format vectoriel (svg) : valeurs de centralité affichées à coté des sommets, système de couleurs permettant de discrimer les sommets en fonction de leur importance mesurée par ces indices …

- Etudier dès le départ la k-connexité des réseaux.

- Etendre l'étude au cas des graphes orientés. Les réseaux initiaux seront supposés fortement connexes, et pour la vulnérabilité suite à la suppression d'arcs il s'agira donc de tester la forte-connexité.

- Calculer les indices de centralité d'intermédiarité des arêtes et pas seulement des sommets.

- Etudier des réseaux routiers selon une approche de flots. L'étude proposée ici se focalise sur l'accessiblité en termes de plus courts chemins, mais il peut être interessant d'étudier les flots de circulation. L' indice de centralité intermédiaire d'un sommet s_i peut se mesurer par rapport aux flots :

  $$\sum_{j,k \neq i} \frac{flot\ maximum\ entre\ \ s_j\ et\ s_k\ transitant\ par\ s_i}{flot\ maximum\ entre\ \ s_j\ et\ s_k}$$

  Suite à la rupture d'un tronçon, comment peut se re-répartir le flot ?
  …

- Approfondir l'analyse de la vulnérabilité. Cette extension comporte une partie documentation et recherche non négligeable.
  Quelques idées : Pour chacun des indices de centralité présentés dans le sujet, on peut définir un indice de centralité global du réseau. Par exemple on peut définir la centralité de proximité globale du réseau comme :

$$C_P = \frac{\sum_i [\max_i C_P(s_i) - C_P(s_i)]}{\frac{n^2 - 3n + 2}{2n - 3}}$$

  Il peut alors être interessant de voir l'effet de la suppression d'arêtes sur cet indicateur et surtout de trouver les arêtes dont la suppression a le plus d'impact …

- Travailler sur un exemple de réseau réel. Récupérer par exemple le réseau routier principal d'un pays ou d'une île. L'étude pourrait aussi être appliquée au réseau du métro parisien.

• *Consignes générales*

En dehors des exemples du cours et des codes officiellement fournis, toute détection de copie massive de code tiers sera considérée comme plagiat et très sévèrement sanctionné, avec circonstances aggravantes si :

- l'emprunt n'est pas cité
- l'emprunt vient du travail d'une autre équipe d'étudiants de l'ECE pour ce même projet
- il y a tentative de dissimulation (maquillage de code...)
- vous avez déjà été repéré dans les « cas suspects » des dépôts de TP

Il reste possible d'utiliser un code public (publié sur Internet) avec ou sans modifications mais l'emprunt doit être **très explicitement et très clairement cité** en tant que tel dans le code (source, début et fin de l'emprunt) et également **explicitement mentionné lors de la soutenance**.
Le code source doit être présentable : bien indenté, raisonnablement aéré et commenté (chapitré).
Les identifiants doivent être explicites, on doit comprendre de quoi on parle en lisant le nom des classes, des attributs, des méthodes. Le projet est structuré en .cpp et .h.

Bibliographie

[1] P. G. T. G.-P. W. G. Martinelli A., «Indicateurs d'accès pour une mobilité durable» Rapport du PNR 41 "Transport et environnement", rapport A11, Berne, 2000.

[2] M. Drevelle, «Site du groupe fmr (flux, matrices, réseaux)» [En ligne]. Available: https://groupefmr.hypotheses.org/.

[3] S. Borgatti, «Centrality and AIDS,» *University of South California, Connections n°18,* 1995.

[4] J.-F. Gleyze, «Réseaux, territoires et accessibilité» Institut Géographique national, laboratoire Cogit, 2001.

[5] U. Brandes, «A faster algorithm for betweenness centrality» *Journal of mathematical Sociology,* 2001.