

Analyse des réseaux : Une introduction à *Pajek*

Laurent Beauguitte

(UMR Géographie-cités, groupe fmr - flux, matrices, réseaux)

Pierre Mercklé

(Centre Max Weber, équipe « Dispositions, pouvoirs, cultures, socialisations »)

L'analyse des réseaux sociaux, longtemps réservée à quelques happy few en raison à la fois de la difficulté à constituer des corpus de données, et de la technicité des algorithmes à mettre en œuvre, s'est largement diffusée et démocratisée au cours de la dernière décennie. L'apparition d'outils de plus en plus faciles à manier, et l'explosion des nouvelles technologies de communication numérique, véritable eldorado statistique, y ont largement contribué.

Quanti souhaite participer à sa modeste manière à ce mouvement, en vous faisant découvrir un de ces outils : il s'agit de [Pajek](#), un logiciel d'analyse et de visualisation de réseaux pour Windows. Développé par deux chercheurs slovènes, Vladimir Batagelj et Andrej Mrvar, il a le grand mérite d'être gratuit et puissant, ce qui lui a permis de s'imposer comme un des outils les plus utilisés en analyse des réseaux¹.

Pour citer ce billet :

Beauguitte Laurent et Pierre Mercklé (2011), « Analyse des réseaux : une introduction à Pajek », *Quanti*, mis en ligne le 6 juin 2011, URL : <http://quanti.hypotheses.org/512>.

¹ D'autres outils, comme [NetDraw](#) (gratuit mais non libre), [Gephi](#) (libre et gratuit) ou [Tulip](#) (libre et gratuit) pour la visualisation, et pour l'analyse [Ucinet](#) (payant) ou certains packages pour [R](#) comme SNA ou iGraph (libres et gratuits), peuvent fournir des compléments intéressants à *Pajek*, voire constituer des alternatives pertinentes pour certains aspects. Nous en disons un mot très rapide à la fin de cet article, en attendant de leur consacrer de plus longs développements.

Table des matières

| | |
|--|-----------|
| Introduction : qu'est-ce que l'analyse des réseaux ? | 3 |
| Installer Pajek | 4 |
| Découvrir Pajek | 5 |
| Les objets..... | 6 |
| Les commandes..... | 7 |
| Préparer et importer des données relationnelles dans Pajek | 7 |
| Décrire les liens | 8 |
| Décrire les sommets..... | 11 |
| Un fichier pour les gouverner tous : le fichier de projet « .paj » | 12 |
| Pour aller plus loin dans la préparation des données..... | 13 |
| Visualiser des réseaux | 15 |
| Analyser des réseaux | 16 |
| Les propriétés locales d'un réseau..... | 17 |
| Les propriétés globales d'un réseau..... | 18 |
| Le blockmodeling | 20 |
| Travailler avec Pajek : un exemple pour aller un peu plus loin... | 22 |
| Préparer et importer les données..... | 22 |
| Visualiser le réseau..... | 24 |
| Manipuler et modifier les données..... | 27 |
| Conclusion : les atouts et les limites de Pajek | 29 |
| Liens utiles | 30 |
| Références bibliographiques | 31 |

Introduction : qu'est-ce que l'analyse des réseaux ?

L'analyse des réseaux sociaux s'intéresse aux relations entre unités sociales, ces unités pouvant être des individus, des groupes d'individus, des entreprises, des Etats, etc. En ce sens, on peut considérer que l'analyse des réseaux se présente comme un ensemble de méthodes, de concepts, de théories, de modèles et de techniques, mobilisables dans les différentes disciplines des sciences sociales comme dans d'autres domaines (par exemple en littérature, en biologie, en sciences physiques...), et qui consistent à prendre pour objets d'étude non pas les propriétés individuelles des unités étudiées (quand il s'agit d'individus : leur âge, leur sexe, leur niveau de diplôme, leur profession, etc.), mais les relations entre ces unités (quelle que soit la nature de ces relations) et les régularités qu'elles présentent. L'objectif est de les décrire, de rendre compte de leur formation et de leurs transformations, et d'analyser les effets de ces propriétés, parfois appelées « structurales », sur leurs comportements (Mercklé, 2011).

Pour celles et ceux qui souhaitent se familiariser avec les principes et les méthodes de l'analyse des réseaux sociaux, de nombreux ouvrages facilitent l'initiation : en français, les ouvrages de Degenne et Forsé (2004), Lazega (2007) ou Mercklé (2011) fournissent une introduction solide ; en langue anglaise, le manuel très complet de Wasserman et Faust (1994) reste la référence. En outre, vous trouverez une bonne introduction au vocabulaire de l'analyse des réseaux, par Laurent Beauguitte, à cette adresse : http://halshs.archives-ouvertes.fr/docs/00/54/18/98/PDF/fmr1_vocabulaire_notation.pdf.

Le présent article² a pour objectif de permettre, à celles et ceux qui souhaitent mettre en œuvre les techniques de l'analyse des réseaux, une prise en main rapide du logiciel [Pajek](#).

² La version initiale de ce document a été produite dans le cadre des activités du groupe fmr (flux, matrices, réseaux) animé par Laurent Beauguitte et César Ducruet depuis octobre 2010 au sein de l'UMR Géographie-cités. Elle est disponible à l'adresse suivante : <http://cel.archives->

Celui-ci présente plusieurs avantages : il est gratuit³, il fonctionne aussi bien sous Windows, Mac OSX⁴ ou Linux, il permet d'étudier des graphes de grande taille (comportant plusieurs milliers de « nœuds »⁵), et il est compatible avec les autres logiciels existants (notamment *Ucinet* et *R*). En outre, son site internet (<http://pajek.imfm.si>) propose une documentation fournie, et de nombreux corpus de données de démonstration...

...Mais *Pajek* a un inconvénient de taille : sa prise en main n'est pas totalement intuitive, et sa maîtrise demande quelques heures de persévérance... Heureusement, ce guide⁶ est là pour vous aider !

Installer *Pajek*

Commençons par le commencement... Pour télécharger *Pajek*, il faut aller sur cette page :

<http://pajek.imfm.si/doku.php?id=download>

Vous y trouverez des liens pour télécharger la dernière version en date de *Pajek*. Dans notre cas, et à la date où nous écrivons ce guide, nous allons installer la **version 2.03** pour Windows 32 bits :

<http://vlado.fmf.uni-lj.si/pub/networks/pajek/pajek203.exe>

Cliquez sur ce lien, puis cliquez sur « Exécuter⁷ » (ou sur « Enregistrer » si vous voulez d'abord le sauver sur votre ordinateur avant de l'installer), et suivez les instructions : en quelques secondes, *Pajek* sera installé sur votre ordinateur. Mais attention : le programme

ouvertes.fr/docs/00/56/44/14/PDF/outils_pajek.pdf. Des documents de synthèse ont également été produits sur différents aspects (vocabulaire, mesures locales et globales, cliques et clusters, blockmodeling), ils sont accessibles à l'adresse <http://halshs.archives-ouvertes.fr/FMR/fr/>. La version présentée dans ce document a été revue et adaptée par Pierre Mercklé à partir de ses propres expériences d'utilisation de *Pajek*.

³ Mais il n'est pas « libre », autrement dit son code n'est pas accessible et ne peut pas être modifié.

⁴ Tutoriel en ligne pour Mac OSX : <http://vlado.fmf.uni-lj.si/pub/networks/pajek/howto/PajekOSX.pdf>.

⁵ Pour les mathématiciens, la taille d'un graphe revoit au nombre de liens, le nombre de sommets ou de « nœuds » correspondant à « l'ordre » du graphe.

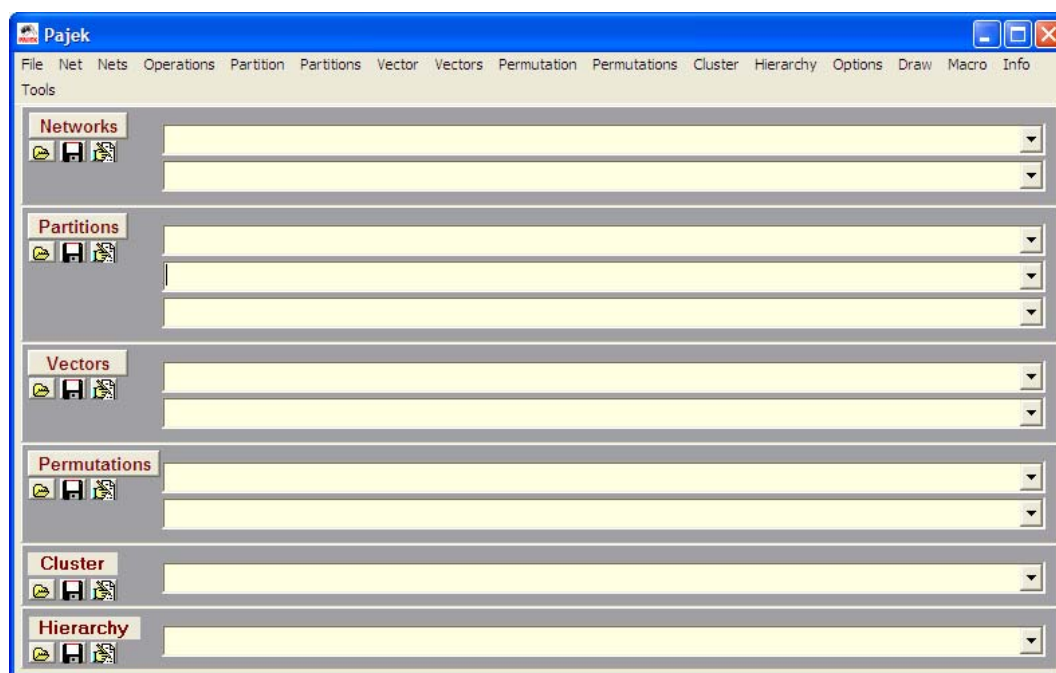
⁶ Si vous repérez des erreurs ou des approximations, n'hésitez surtout pas à nous en faire part !

⁷ En tant qu'administrateur dans les dernières versions de Windows.

d'installation ne crée pas de raccourci vers le logiciel dans le menu « Démarrer » de votre ordinateur, mais seulement dans le répertoire d'installation, qui est ouvert à la fin de l'installation : il vous suffit donc de copier ce raccourci et de le coller sur votre bureau, par exemple, pour pouvoir ensuite accéder facilement au logiciel.

Découvrir *Pajek*

Pour démarrer *Pajek*, il suffit de double-cliquer sur le raccourci évoqué ci-dessus. Vous découvrez alors l'interface du logiciel, et le moins qu'on puisse dire, c'est que ça n'est pas totalement « *User Friendly* » : difficile de comprendre comment on peut utiliser le logiciel simplement en contemplant cet écran d'accueil⁸...



Il faut donc s'aider du manuel de *Pajek*, disponible en ligne en anglais au format PDF. Vous pouvez le télécharger à l'adresse suivante :

<http://pajek.imfm.si/lib/exe/fetch.php?media=dl:pajekman203.pdf>

⁸ Ajoutons qu'il est paradoxal que, pour un logiciel gratuit, le seul véritable mode d'emploi digne de ce nom (De Nooy, Mrvar et Batagelj, 2005) soit payant (30 euros couverture souple, 62 euros couverture rigide !). Le [manuel de référence](#) librement téléchargeable est d'une utilité relative.

Les objets

Les trois principaux objets manipulables avec *Pajek* sont les réseaux (« Networks »), les partitions (« Partitions ») et les vecteurs (« Vectors »), qui correspondent aux trois premières sections de la fenêtre principale :

- un réseau est constitué d'une liste de sommets et d'une liste de liens entre eux (éventuellement signés ou valués) ;
- une partition permet de classer les sommets en différents ensembles distincts les uns des autres et, plus généralement, d'attribuer un nombre entier à chaque sommet. Par exemple, il est possible de créer une partition codant des villes en 1, 2 et 3 selon leur statut administratif (préfecture, sous-préfecture, autre) ;
- un vecteur est un nombre (entier ou décimal) associé à un sommet, sur lequel il est possible d'effectuer des opérations arithmétiques.

Pajek utilise des menus déroulants à la mode Windows ou Mac. Le fonctionnement de *Pajek* obéit à une logique générale de fonctionnement extrêmement simple, même si elle n'est pas totalement habituelle :

- Pour chaque opération que l'on veut effectuer, il faut d'abord sélectionner dans les différentes sections de la fenêtre principale les objets sur lesquels doivent porter ces opérations ;
- Ensuite, il faut dérouler les menus de la barre supérieure pour trouver les commandes que l'on souhaite effectuer ;
- A chaque fois que vous procédez ainsi, et qu'une opération est effectuée, *Pajek* crée un objet (réseau, partition ou vecteur), et le rajoute dans la section correspondante de la fenêtre principale.

Ensuite, pour voir ces objets, il suffit de double-cliquer dessus, et pour avoir des informations supplémentaires sur ces objets, par exemple pour connaître les effectifs et pourcentages bruts et cumulés des différents indicateurs recherchés (partition ou vecteur), il

faut utiliser le menu « **Info** ». Et pour sauvegarder, transformer ou effacer un de ces objets, cela se fait à partir des différentes commandes du menu « **File** ».

Les commandes

En dehors des menus « **Info** » et « **File** », il faut bien reconnaître que la ligne de commandes en haut de la fenêtre principale est un peu obscure au premier abord, aussi quelques repères peuvent être utiles avant de commencer à manipuler des données :

- Si on souhaite agir sur un réseau (pour l'ouvrir, le transformer...), c'est le menu « **Net** » qu'il faut dérouler ;
- Si les opérations concernent deux réseaux (ou plus), il faut utiliser le menu « **Nets** » ;
- Pour dessiner des représentations graphiques des réseaux : « **Draw** » ;
- Pour faire des opérations entre différents objets, c'est le menu « **Operations** »...

Mais pour nous lancer plus avant dans la compréhension de ces différentes fonctionnalités, il nous faut tout d'abord des données...

Préparer et importer des données relationnelles dans *Pajek*

Puisque l'analyse des réseaux porte moins sur les propriétés des individus que sur les propriétés des relations entre eux, elle a besoin de corpus de données fournissant des informations sur ces relations, en plus des données qui fournissent des informations sur les individus : c'est la grande différence entre les corpus de données qu'on manipule traditionnellement en sciences sociales (et qui croisent individus en lignes et variables en colonnes), et les corpus de données relationnelles.

Ceux-ci peuvent généralement se présenter sous deux formes :

- soit sous la forme d'une matrice carrée (qu'on appelle une « matrice d'adjacence »), avec les individus disposés à la fois en lignes et en colonnes dans le même ordre, et

des 1 et des 0 à l'intersection des lignes et des colonnes pour figurer l'existence ou l'absence de relations entre eux ;

- soit, dès que le nombre d'individus dans le réseau devient un peu élevé, sous la forme d'une liste de liens comportant au moins deux variables : l'identifiant de l'émetteur ou de la source du lien (son « prédécesseur » dans le langage des graphes), et l'identifiant du récepteur ou du destinataire du lien (son « successeur »). D'autres variables peuvent venir compléter ces deux informations, pour préciser la nature du lien, son intensité, sa date, etc.

Décrire les liens

La meilleure façon de comprendre à quoi ressemblent des données relationnelles est encore d'examiner des fichiers d'exemples. *Pajek* en met plusieurs à votre disposition, que vous pouvez télécharger aux deux adresses suivantes :

<http://vlado.fmf.uni-lj.si/pub/networks/data/> (pour les plus anciens)

<http://pajek.imfm.si/doku.php?id=data:pajek:index> (pour quelques nouveaux fichiers)

N'hésitez pas à en ouvrir quelques uns avec un éditeur de texte pour les examiner ! Vous verrez que la structure minimale d'un fichier de « réseau » (ce sont les fichiers avec l'extension « .net ») utilisable par *Pajek* contient une liste de sommets (*vertices*), et, à la suite, soit une matrice d'adjacence (*matrix*), soit une liste de liens orientés (*arcs*) ou non-orientés (*edges*). *Pajek* accepte donc les données sous les deux formes⁹, mais dans la mesure où tout l'intérêt de *Pajek* est de traiter des graphes comprenant plusieurs centaines ou milliers de nœuds, l'intérêt de manipuler une matrice de cette taille n'apparaît pas évident. C'est pourquoi dans la suite, nous privilégierons plutôt le format des listes de liens.

Le fichier minimal doit donc avoir la structure suivante :

⁹ En réalité, comme *Pajek* est également capable de lire les fichiers de données relationnelles au format DL, il peut accepter de très nombreuses autres façons de formater les données relationnelles. Vous pouvez découvrir une longue présentation du format DL dans le [guide d'utilisation du logiciel Ucinet](#).


```

*Vertices nombre_de _noeuds
1 "label" x y z
2 "label" x y z
...
*Arcs (en cas de graphe orienté)
numéro_origine numéro_destination valeur
...
*Edges (en cas de graphe non orienté)
numéro_origine numéro_destination valeur
...

```

Pour les sommets, les rubriques x, y et z sont optionnelles, elles permettent d'assigner aux sommets des coordonnées spatiales en x (axe horizontal), y (axe vertical) et z (hauteur) comprises entre 0 et 1. L'origine des axes horizontal et vertical (0,0) est en haut à gauche de l'écran, les coordonnées (1,1) désignant le coin inférieur droit.

En ce qui concerne *arcs* et *edges*, seuls le numéro d'origine et le numéro de destination sont obligatoires. Mais vous pouvez affecter les liens d'une « valeur », qui peut être positive ou négative, entière ou décimale (dans ce dernier cas, le point, et non la virgule, doit être utilisé comme séparateur de décimales). Il est également possible de créer des boucles (lien d'un sommet vers lui-même), qui ne seront pas représentées graphiquement mais prises en compte dans les calculs effectués.

Dans la constitution de vos propres fichiers, faites bien attention à utiliser des espaces et non des tabulations, sinon *Pajek* refusera obstinément d'ouvrir le fichier. Attention également à ne pas insérer de sauts de ligne dans le fichier source entre la liste des sommets et la liste des liens (et tant pis pour la lisibilité de l'ensemble).

Pour ce guide, nous vous proposons de commencer avec des données relationnelles volontairement très simples, tirées de l'exemple fictif utilisé par Pierre Mercklé dans le chapitre 2 de *Sociologie des réseaux sociaux* (pp. 21-35). Dans cet exemple, nous imaginons que nous voulons étudier le réseau des « citations » entre cinq sociologues, à partir de l'étude de leurs publications et des références bibliographiques qu'elles contiennent. En examinant les bibliographies figurant à la fin de leurs travaux, nous constatons que : le sociologue A cite le sociologue B et le sociologue C ; le sociologue B cite les sociologues A, D et E ; le sociologue C ne cite personne ; D cite E ; et enfin, E cite C.

Dans Pajek, on peut utiliser soit une matrice, soit une liste de liens pour décrire ce réseau. Suivant le mode de description retenu, cela donnera :

| Matrice d'adjacence | Liste de liens |
|--|--|
| <pre>*Vertices 5 1 "A" 2 "B" 3 "C" 4 "D" 5 "E" *Matrix 0 1 1 0 0 1 0 0 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0</pre> | <pre>*Vertices 5 1 "A" 2 "B" 3 "C" 4 "D" 5 "E" *Arcs 1 2 1 3 2 1 2 4 2 5 4 5 5 3</pre> |

Nous allons travailler dans la suite avec le second format, celui de la liste de liens. Le fichier pour *Pajek* décrivant ce « réseau » par une liste de liens a une extension « .net », et vous pouvez le télécharger en cliquant sur le lien ci-dessous :

citations.net

Une fois ce fichier téléchargé, vous pouvez l'ouvrir avec n'importe quel éditeur de texte.

Voici ce que vous y trouverez :

```
% Réseau de citations
% P. Mercklé 2011

*Vertices 5
1 "A"
2 "B"
3 "C"
4 "D"
5 "E"
*Arcs
1 2
1 3
2 1
2 4
2 5
4 5
5 3
*Edges
```

Et pour ouvrir ce fichier dans *Pajek*, il suffit de sélectionner « **Network > Read** » dans le menu « **File** » : une fenêtre « **Report** » apparaît pour vous indiquer le (bon) déroulement de l'opération, et le fichier apparaît dans la partie « **Network** » de la fenêtre principale de *Pajek*.

Décrire les sommets

Les fichiers avec l'extension « .net » servent à désigner les sommets (et éventuellement leur emplacement dans l'espace) et à décrire les liens. Pour décrire les sommets eux-mêmes (au-delà de leur position), il faut fournir à *Pajek* des fichiers avec l'extension « .clu » ou « .vec » :

Les fichiers « .clu » contiennent des « partitions » : elles permettent d'assigner à chaque sommet une valeur numérique entière, correspondant le plus souvent à une classe d'appartenance.

Les fichiers « .vec » contiennent des vecteurs : ils permettent d'assigner aux sommets des valeurs décimales, sur lesquelles il est possible de faire des opérations arithmétique (comme par exemple calculer une moyenne).

Nous pouvons ainsi imaginer que les sociologues de notre réseau ont un sexe (valeur discrète, 1 pour homme et 2 pour femme) et un âge (valeur continue), que nous pouvons stocker dans les fichiers « .clu » et « .vec » correspondant :

[citations.clu](#)

[citations.vec](#)

Par exemple, le fichier « citations.clu » contenant l'année de naissance de chaque sociologue se présente ainsi (les fichiers « .clu » sont construits exactement sur le même modèle) :

```
*Vertices 5
1938
1968
1975
1961
1953
```

Vous pouvez ouvrir ces fichiers en sélectionnant « **Partition > Read** » ou « **Vector > Read** » dans le menu « **File** ».

Par ailleurs vous pouvez également ajouter une nouvelle partition pour un réseau existant et déjà ouvert dans *Pajek*. Pour cela, sélectionnez « **Create constant partition** » dans le menu « **Partition** », puis sélectionnez « **Partition > Edit** » dans le menu « **File** » pour coder manuellement les différents sommets afin de les répartir dans les classes désirées.

Un fichier pour les gouverner tous : le fichier de projet

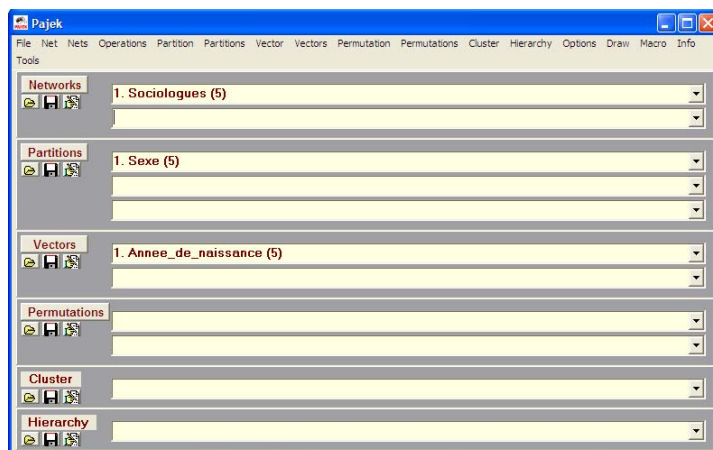
« .paj »

Toutes les informations permettant de décrire un réseau (liste et position des sommets, liste et propriétés des liens, propriétés des sommets) peuvent être stockées dans des fichiers différents, avec les extensions « .net », « .clu » et « .vec ». Mais elles peuvent également être stockées toutes ensemble dans un fichier de projet unique, avec l'extension « .paj ».

Lorsqu'on débute avec *Pajek*, les erreurs sont fréquentes et il est plutôt conseillé d'utiliser des sous-fichiers pour chaque type d'informations. Mais si vous vous sentez près à travailler avec un fichier unique sur notre exemple de réseau de sociologues, n'hésitez pas à télécharger le fichier suivant :

[citations.paj](#)

Pour l'ouvrir, sélectionnez « **Pajek Project File > Read** » dans le menu « **File** ». Et au total, que vous ayez ouvert successivement les trois fichiers « .net », « .clu » et « .vec », ou bien seulement le fichier « .paj », votre fenêtre principale de *Pajek* devrait maintenant ressembler à cela :



Pour celles et ceux qui veulent en savoir plus sur la préparation des données, et notamment sur les données de réseaux bipartites (*two-mode networks*) et sur les graphes dynamiques, vous pouvez lire ce qui suit immédiatement. Sinon, nous vous conseillons de passer directement à la partie suivante : « Visualiser des réseaux ».

Pour aller plus loin dans la préparation des données

Les réseaux bipartites (two-mode networks)

Un réseau bipartite (*two-mode network*, en anglais), ou « réseau d'affiliations », est un réseau dans lequel les liens relient des sommets d'un premier ensemble et les sommets d'un second ensemble. Par exemple, les réseaux « acteurs-événements » correspondent à des réseaux bipartites, comme dans l'exemple classiques où l'on utilise l'entrecroisement des conseils d'administration d'entreprises (*corporate interlocking*) à partir de données où les individus figurent en ligne, et les entreprises en colonnes.

Pour créer un réseau bipartite dans *Pajek*, le seul changement concerne la déclaration des sommets. Il convient en effet d'indiquer après le nombre total de sommets combien font partie du premier sous-ensemble de sommets. Si jamais il y a une erreur dans la liste des liens, c'est-à-dire si on a déclaré un lien à l'intérieur du même sous-ensemble de sommets, *Pajek* le signale à l'ouverture du fichier (mais l'ouvre malgré tout).

Le fichier .net suivant crée un réseau bipartite entre deux ensembles de sommets [a;b] et [1;2;3] :

```
*Vertices 5 2
1 "a"
2 "b"
3 "1"
4 "2"
5 "3"
*Edges
3 1 1
3 2 1
4 1 1
5 1 1
```

Pour transformer ce réseau en réseau unipartite (*one-mode network*), il faut utiliser la commande « **Transform > 2-Mode to 1-Mode** » du menu « **Net** », et on choisit ensuite si on

souhaite utiliser les lignes ou les colonnes pour transformer la matrice de départ. On obtient ainsi un réseau dans lequel les unités sont considérées comme liées si elles sont liées au même sommet de l'autre sous-ensemble. Ainsi, dans le cas des conseils d'administration, on obtiendra un réseau reliant les individus appartenant aux mêmes conseils d'administration, ou bien un réseau reliant les entreprises ayant des membres de conseil d'administration en commun.

Les graphes dynamiques

L'un des intérêts de *Pajek* est de permettre de façon relativement simple de décrire les états d'un réseau à différents moments, et de représenter graphiquement les évolutions d'un état à l'autre. Le texte ci-dessous montre les éléments indispensables pour créer un tel fichier, il est suivi d'un exemple montrant un graphe évoluant au fil du temps.

```
*Vertices nombre_de_vertices
1 "label" [x-y] # signale que le sommet 1 est présent du temps x
au temps y}
2 "label" [y] # le sommet 2 est présent seulement au temps y
3 "label" [x] # le sommet 3 est présent seulement au temps x
4 "label" [x,z] # le sommet 4 est présent au temps x et au temps z
*Edges
1 3 1 [x] # il existe un lien entre le sommet 1 et 3 au temps x
1 2 1 [y] # il existe un lien entre 1 et 2 au temps y
```

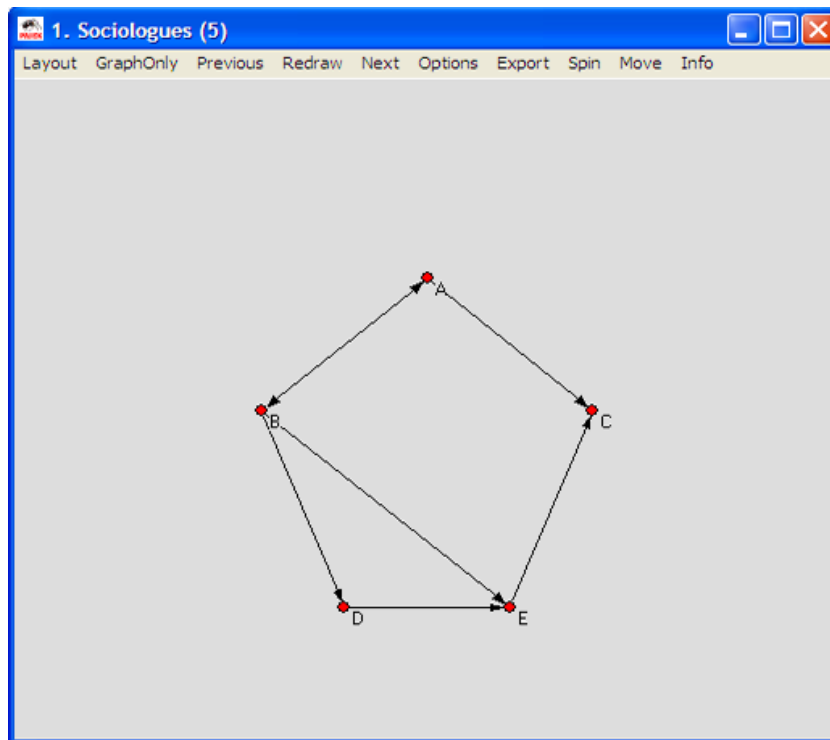
Voici un exemple de graphe évolutif au format .net :

```
*Vertices 4
1 "A" [1-4]
2 "B" [1-4]
3 "C" [1-4]
4 "D" [2,4]
*Edges
1 2 1 [1-4]
2 3 1 [2,4]
2 4 1 [2]
3 4 1 [4]
```

Créez le fichier qui correspond, ouvrez-le avec *Pajek*, puis sélectionnez l'option « **Transform > Generate in time > All** » du menu « **Net** », indiquez 1 comme *first time point* et 4 comme dernier, 1 comme *step*. *Pajek* produit les 4 moments du graphe. Pour visualiser l'évolution, choisissez « **Draw** » (ou le raccourci **Ctrl+G**), le *layout* (algorithme de visualisation) qui vous convient puis « **Options > Previous/Next > Apply to > Network** » et, en cliquant sur « **Next** », les étapes devraient apparaître les unes après les autres.

Visualiser des réseaux

La visualisation des graphes, des partitions et/ou des vecteurs se fait en sélectionnant « **Draw** » dans le menu « **Draw** » (ou en utilisant le raccourci CTRL+G) dans la fenêtre principale de *Pajek*. Si vous utilisez toujours le projet « citations » qui nous a servi d'exemple jusqu'à présent, vous allez voir apparaître la fenêtre suivante :



Il est possible de zoomer sur une partie du graphe, en maintenant enfoncé le bouton droit de la souris et en dessinant la zone à agrandir (cliquez sur « **ZoomOut** » dans le menu pour revenir au graphe initial). Il est aussi possible de déplacer des sommets, en cliquant dessus avec le bouton gauche de la souris.

La fenêtre « **Draw** », grâce à ses différents menus, permet ensuite de modifier le graphe de multiples façons (en déplaçant les sommets, en modifiant leur apparence, en modifiant celle des liens...), que vous pourrez découvrir plus en détail dans la dernière partie de cette introduction, consacrée à l'exploration d'un exemple de réseau de plus grande taille.

Pour faire varier la taille et/ou la couleur des sommets en fonction de leurs propriétés (que celles-ci aient été indiquées au moment de la création du graphe ou qu'elles aient été

calculées avec *Pajek* comme nous l'expliquons un peu plus loin), il suffit ensuite de sélectionner le graphe, la partition et le vecteur souhaités dans la fenêtre principale de *Pajek*, puis de choisir une des options du menu « **Draw** ». Par exemple, si les liens sont « valués » (par des valeurs indiquées dans le fichier « .net » ou dans la partie « *Network » du fichier « .paj »), la représentation des intensités peut se faire de trois manières différentes dans la fenêtre « **Draw** » : en indiquant la valeur du lien (« **Options > Lines > Mark Lines > With Values** ») ; en faisant varier l'épaisseur des traits en fonction de l'intensité du lien (« **Options > Lines > Different Widths** ») ; ou en gardant la même épaisseur mais avec un dégradé du blanc au gris foncé (« **Options > Lines > GreyScale** »).

Et une fois que le graphe vous convient, vous pouvez en sauvegarder l'image dans différents formats (.svg .ps .eps ou bitmap) en sélectionnant « **Export** » dans le menu « **Options** ».

Analyser des réseaux

L'analyse des réseaux offre beaucoup plus qu'une technique de visualisation de systèmes de relations : en s'appuyant sur les apports de la théorie des graphes, elle permet de mesurer un certain nombre de propriétés structurales « locales » des individus et des liens qui composent ces réseaux, lesquelles peuvent ensuite être mobilisées sous forme de variables dans le cadre d'analyses bi-variées et multi-variées, de procédures de classification, etc. Elle permet également de mesurer un certain nombre de propriétés « globales » des réseaux étudiés, ou de sous-parties de ceux-ci, par exemple à des fins de comparaisons. Elle permet enfin de réaliser des partitionnements de réseaux en sous-composantes en fonction des propriétés structurales des éléments qui composent le réseau, par exemple en s'appuyant sur des approches en termes d'équivalence structurale (Mercklé, 2011, p. 65-72).

Attention ! Chaque fois qu'une manipulation sur le réseau étudié est effectuée, *Pajek* demande si on souhaite créer un nouveau réseau (*Make new Network ?*). Répondre « **No** » entraîne l'écrasement du réseau de départ et son remplacement par le résultat obtenu... Il est donc préférable de répondre « **Yes** » pour pouvoir conserver une trace de toutes les étapes successives d'un traitement, ou au moins de conserver des copies de secours du réseau initial, et des étapes les plus importantes du traitement.

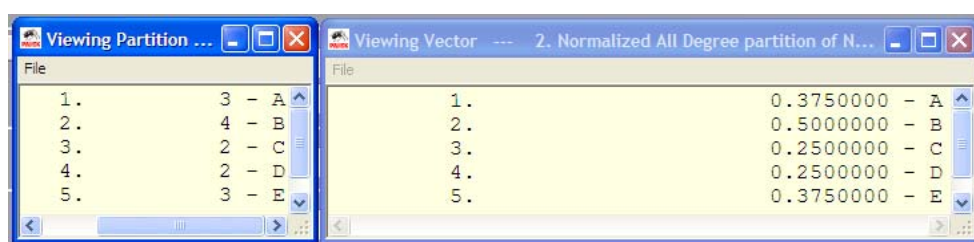
Les propriétés locales d'un réseau

Les degrés

Pour obtenir les « degrés » des sommets (c'est-à-dire le nombre de liens qui leur sont reliés), il faut sélectionner « **Partition > Degree** » dans le menu « **Net** », puis sélectionner « **Input** » pour le demi-degré intérieur (nombre de liens arrivants), « **Output** » pour le demi-degré extérieur (nombre de liens partants), et « **All** » pour le degré (somme des deux demi-degrés).

Comme à chaque commande de ce type, le comportement de *Pajek* est alors le suivant : le logiciel indique les opérations effectuées dans la fenêtre « **Report** », calcule les résultats demandés, puis les range dans une nouvelle partition ou un nouveau vecteur, en fonction de la nature de ces résultats (nombre entiers ou nombre décimaux).

Dans le cas présent, *Pajek* a créé à la fois une partition pour les degrés (les degrés sont des nombres entiers), et un vecteur pour les degrés normalisés (autrement dit les rapports entre les degrés et le degré maximum possible, qui est ici de 8, puisque chaque sociologue peut citer et être cité par au maximum 4 autres sociologues¹⁰). Pour afficher ces résultats, il suffit alors de double-cliquer, dans la fenêtre principale, sur la partition ou le vecteur souhaité, ce qui donne ici :



| File | | |
|------|-------|--|
| 1. | 3 - A | |
| 2. | 4 - B | |
| 3. | 2 - C | |
| 4. | 2 - D | |
| 5. | 3 - E | |

| File | | |
|------|---------------|--|
| 1. | 0.3750000 - A | |
| 2. | 0.5000000 - B | |
| 3. | 0.2500000 - C | |
| 4. | 0.2500000 - D | |
| 5. | 0.3750000 - E | |

Pour connaître le degré moyen d'un réseau, il faut faire une petite manipulation supplémentaire : sélectionnez la partition créée à l'étape précédente, qui doit s'appeler « **All Degree partition of N1 (5)** », puis transformez-là en vecteur avec la commande « **Make**

¹⁰ L'auto-citation, qui correspondrait en analyse de graphe à une boucle (lien d'un sommet vers lui-même), n'est pas considérée ici.

Vector » du menu « **Partition** ». Ensuite, sélectionnez le vecteur obtenu, puis exécutez la commande « **Vector** » du menu « **Info** » (en conservant les options par défaut). Le degré moyen, qui est de 2.8 dans notre exemple, apparaît avec d'autres statistiques descriptives élémentaires de la distribution des degrés, dans la fenêtre « **Report** ».

| Vector Values | Frequency | Freq% | CumFreq | CumFreq% |
|--------------------|-----------|----------|---------|----------|
| (... 2.000] | 2 | 40.0000 | 2 | 40.0000 |
| (2.000 ... 3.000] | 2 | 40.0000 | 4 | 80.0000 |
| (3.000 ... 4.000] | 1 | 20.0000 | 5 | 100.0000 |
| Total | 5 | 100.0000 | | |

Les mesures de centralité

Les mesures de centralité d'intermédiarité (*betweenness*) et de proximité (*closeness*)¹¹ sont accessibles via la commande « **Vector > Centrality** » du menu « **Net** ». Il est possible de tester les corrélations (Spearman ou Pearson) entre deux partitions (ou entre deux vecteurs) à l'aide de la commande **Partitions > Info**.

Les propriétés globales d'un réseau

Pour connaître les caractéristiques de base du réseau, il faut utiliser la commande « **Network > General** » du menu « **Info** ». On obtient alors un récapitulatif avec le nombre de sommets, d'arcs, d'arêtes, de boucles et de liens multiples ainsi que deux mesures de densité (avec et sans boucle).

¹¹ Pour plus d'explication sur la signification de ces mesures, voir Mercklé, 2011, p. 53-54, ou Ducruet, 2010, « Les mesures locales d'un réseau », http://halshs.archives-ouvertes.fr/docs/00/54/69/73/PDF/fmr3_mesures_locales.pdf.

Composantes connexes

Une composante connexe est un sous-ensemble de sommets entre lesquels il existe toujours au moins un chemin composé d'un nombre variable de liens. L'information sur le nombre de composantes connexes concerne un réseau dans son ensemble, elle se détermine donc dans le menu « **Net** », avec la commande « **Components** » (les options « weak » et « strong » concernent les graphes orientés, et indique s'il faut traiter les liens comme orientés ou non).

Distances et diamètre

La distance dite « géodésique » entre deux sommets est le nombre de liens dont est composé le plus court chemin entre eux. Obtenir la matrice des distances entre les sommets du graphe est possible grâce à la commande « **Paths between 2 vertices > Geodesics Matrices*** »¹² du menu « **Net** ». Il est également possible d'obtenir tous les chemins d'une longueur maximale k entre deux sommets avec la commande « **Paths between 2 vertices > Walks with Limited Length** » du menu « **Net** ».

Le diamètre du graphe, qui est la distance géodésique entre les deux sommets les plus éloignés l'un de l'autre¹³, peut être calculé à l'aide de la commande « **Paths between 2 vertices > Diameter** » du menu « **Net** »¹⁴. Il est sans doute plus judicieux d'utiliser la commande « **Paths between 2 vertices > Distribution of Distances > From All Vertices** » du menu « **Net** », qui fournit, en plus du diamètre, la distance moyenne entre sommets.

Cycles

La structure globale du graphe concernant la transitivité (si A est lié avec B et C, B et C sont-ils liés ?) - appelée *clustering* par les physiciens - s'obtient avec la commande « **Network > Triadic Census** » du menu « **Info** ». Le nombre de triades est fourni par catégories.

¹² Une commande Pajek suivie d'une * signale qu'elle est gourmande en ressources et en temps, il est donc conseillé de la réserver aux graphes ne dépassant pas la centaine de sommets.

¹³ Ou le plus long des plus courts chemins.

¹⁴ Si le graphe n'est pas connexe, les données fournies concernent le sous-graphe connexe où ces valeurs sont les plus élevées.

K-cores

Pajek est limité pour la recherche des sous-graphes fortement connexes. La recherche des *k-cores*¹⁵ se fait à l'aide de la commande « **Partitions > Core** » du menu « **Net** ».

Pour en savoir plus sur les propriétés globales d'un réseau :

http://halshs.archives-ouvertes.fr/docs/00/54/69/73/PDF/fmr3_mesures_globales.pdf

Pour en savoir plus sur les cliques, communautés et dérivées :

http://halshs.archives-ouvertes.fr/docs/00/55/68/67/PDF/fmr4_cliques_communautes_et_derivees.pdf

Le blockmodeling

Deux algorithmes de partitionnement de graphe basés sur l'équivalence sont disponibles, l'un concerne l'équivalence structurale et l'autre l'équivalence régulière. Des sommets sont dits structurellement équivalents s'ils entretiennent exactement les mêmes relations avec les mêmes autres sommets. L'équivalence régulière affaiblit cette condition, des sommets sont équivalents s'ils appartiennent à un *cluster*¹⁶ dont tous les éléments ont une relation avec au moins un élément d'un autre *cluster* d'équivalence régulière donnée¹⁷.

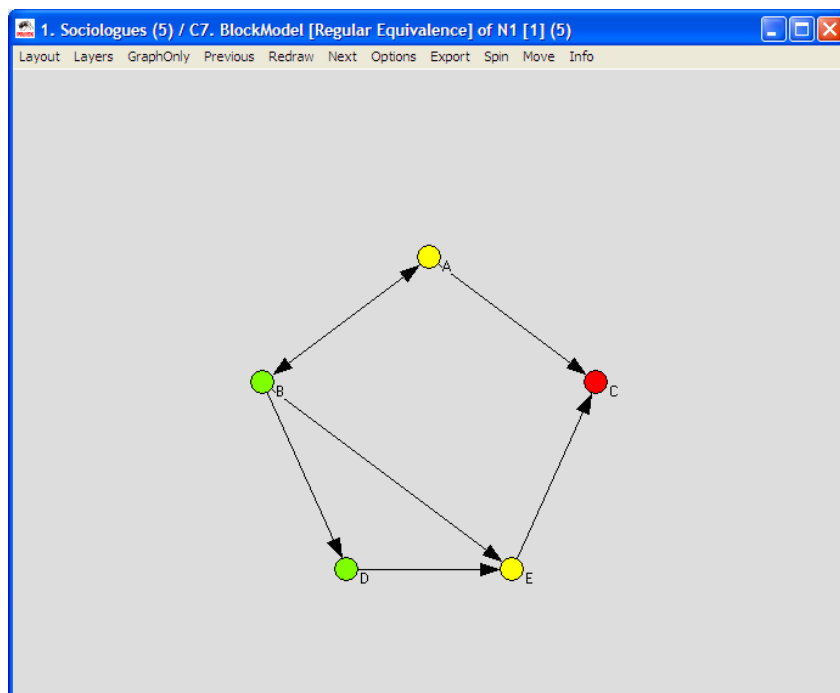
Les algorithmes de *blockmodeling* dans *Pajek* sont accessibles à partir de la commande « **Blockmodeling*** » du menu « **Operations** ». Les seules options proposées par défaut concernent le type d'équivalence, le nombre d'itérations et le nombre de blocs souhaités. Dans notre exemple de réseau de citations entre 5 sociologues, l'algorithme de *blockmodeling* par équivalence régulière propose trois *clusters* : le premier comporte le seul sociologue C, qui ne cite personne ; le deuxième inclut les sociologues A et E, qui ont pour point commun de citer tous les deux C, et d'être cités par B ou par D ; et enfin, ces deux derniers, B et D, forment le troisième *cluster*, celui des sociologues qui citent les autres mais

¹⁵ Un *k-core* est un ensemble d'au moins 3 sommets qui tous sont voisins d'au moins k autres sommets.

¹⁶ Ce terme est ici utilisé comme synonyme de sous-graphes.

¹⁷ Pour en savoir plus sur les théories de l'équivalence structurale, voir Mercklé, 2011, p. 65-72, et http://halshs.archives-ouvertes.fr/docs/00/56/64/74/PDF/fmr5_blockmodeling_equivalence.pdf.

que personne ne cite (à part eux-mêmes). Le résultat peut être représenté sur le graphe à l'aide de couleurs différentes, en sélectionnant la partition contenant le résultat du *blockmodeling*, puis en sélectionnant la commande « **Partition** » du menu « **Draw** » :



Ajoutons enfin qu'il est possible de visualiser la matrice d'adjacence du graphe étudié, dans sa forme originale ou dans la forme bloquée¹⁸. Les commandes respectives sont « **Network > Export Matrix to EPS > Original** » et « **Network > Export Matrix to EPS > Using Permutations** » dans le menu « **File** ». Si une partition a été effectuée, la matrice image (*image matrix*) correspondante peut également être visualisée.

¹⁸ Une matrice bloquée permute lignes et colonnes afin d'obtenir des sous-graphes homogènes.

Travailler avec Pajek : un exemple pour aller un peu plus loin...

L'exemple fictif utilisé jusqu'à présent a le mérite de la simplicité, due à son faible nombre de sommets et de liens. Mais c'est évidemment quand la taille du graphe augmente notablement que l'analyse des réseaux peut déployer toute sa puissance et ses vertus heuristiques. Dans la dernière partie de ce guide, nous souhaitons donc vous présenter, même rapidement, les principes les plus généraux de l'analyse de plus grands réseaux avec *Pajek*.

L'exemple que nous allons utiliser dans cette dernière partie est celui de ce qu'on va appeler « l'Hyosphère », autrement dit le graphe des relations entre les différents carnets de recherche hébergés sur la plateforme Hypotheses.org, développée par Revues.org. La plateforme accueille actuellement près de 200 blogs de chercheurs ou d'équipes, qui sont constitués d'articles dans lesquels il peut arriver à leurs auteurs de citer un article d'un autre carnet de recherche d'Hypothèses : les liens hypertextes ainsi établis entre carnets de recherche constituent, ensemble, une sorte de réseau des circulations possibles à l'intérieur de « l'Hyosphère »¹⁹.

Préparer et importer les données

Nous avons choisi ici de prendre pour exemple non pas ce réseau des circulations possibles, mais celui des circulations réelles, en mesurant le trafic des internautes empruntant ces liens à partir des enregistrements (*logs*) du serveur d'Hypothèses. Le fichier ci-dessous a été constitué à partir des *logs* du trafic sur Hypothèses sur une période de quatre mois, entre le 1^{er} janvier et le 30 avril 2011 :

¹⁹ Pour une première étude de ce réseau des citations entre carnets de recherche d'Hypothèses, voir ce billet de Josquin Debaz sur SocioArgu : <http://socioargu.hypotheses.org/1921>. Et pour une étude plus détaillée de l'Hyosphère à partir des données utilisée ici, dans cette introduction à *Pajek*, voir également ce billet sur le blog de Pierre Mercklé : <http://pierremerckle.fr/2011/06/lhyosphere/>. Notez d'ailleurs qu'en citant ce billet de SocioArgu dans Quanti, je viens de rajouter un nouveau lien dans l'Hyosphère !

hyposphere.net

Dans ces *logs*, à chaque fois qu'une page d'un carnet est consultée, une nouvelle ligne s'ajoute, qui contient en particulier le nom du carnet consulté et la page d'où arrivait le visiteur : nous n'avons donc conservé que les lignes correspondant à des visites de pages de carnets d'Hypothèses en provenance d'autres carnets d'Hypothèses, et nous avons comptabilisé sur cette période de quatre mois le nombre de visites entre chaque paire de carnets.

En réalité, ce que tout étudiant ou chercheur qui se lance dans l'analyse de données quantitatives (qu'elle soient relationnelles ou non) finit par découvrir assez rapidement, c'est que ce n'est pas la mise en œuvre des logiciels et des méthodes d'analyse qui pose les plus grandes difficultés, mais bien plutôt ce travail, souvent ingrat, de recueil et de constitution des corpus de données. Dans ce domaine, chacun se constitue progressivement un répertoire de techniques, d'astuces, d'habitudes, relevant souvent plus du bricolage que de la méthode. Il serait probablement difficile de formaliser complètement ces répertoires de techniques dans un manuel, mais le livre de Claire Lemerrier et Claire Zalc (*Méthodes quantitatives pour l'historien*, La Découverte, 2008) fournit tout de même quelques aperçus intéressants. Ici, nous avons utilisé un mélange de formules dans Excel, de macros dans Batch Replacer et surtout de scripts de « data step » dans SAS pour parvenir à nos fins. Il faut signaler aussi l'existence de deux petites applications, appelées [Excel2pajek](#) et [Txt2pajek](#), qui permettent de transférer facilement des données relationnelles pour pouvoir les utiliser dans *Pajek*.

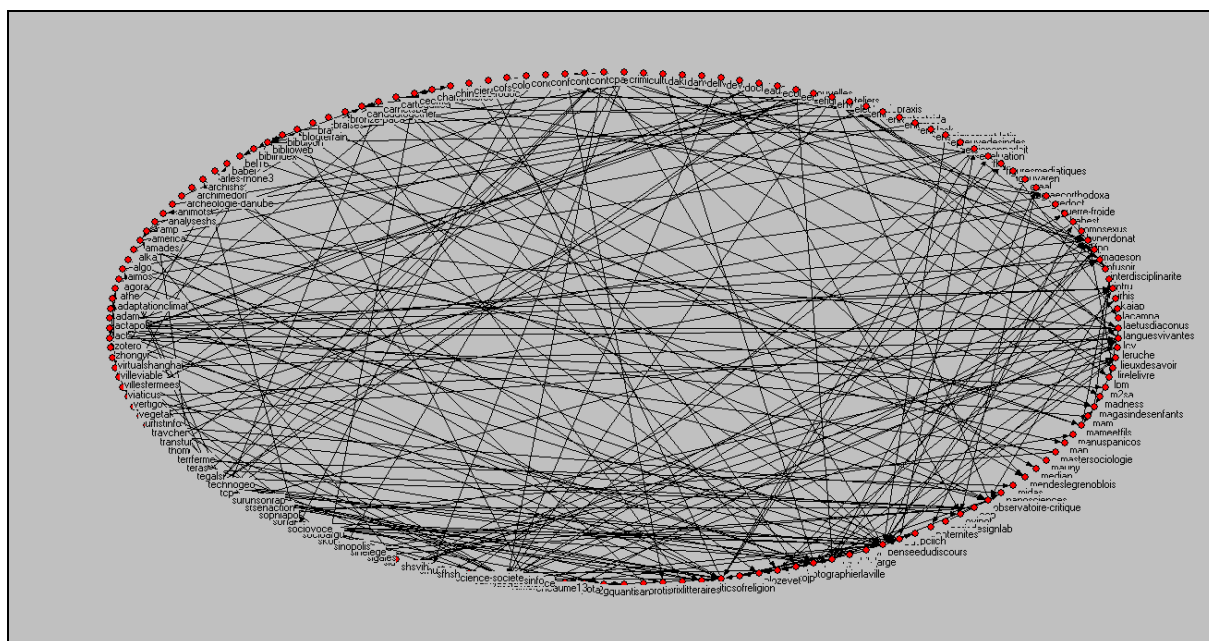
Le fichier « hyposphere.net » ainsi obtenu est donc constitué de la façon suivante :

- La partie « ***Vertices** » contient les identifiants et les noms des 191 carnets d'Hypothèses mentionnés dans les *logs* du serveur ;
- La partie « ***Arcs** » contient la liste de tous les liens entre carnets, chaque lien étant caractérisé par l'identifiant du carnet émetteur du lien, l'identifiant du carnet récepteur, et le nombre de visites ayant emprunté ce lien entre le 1^{er} janvier et le 30 avril 2011 (en moyenne mensuelle).

Pour l'ouvrir, sélectionnez « **Network > Read** » dans le menu « **File** », puis sélectionnez le fichier « *hyposphere.net* » que vous avez enregistré sur votre ordinateur.

Visualiser le réseau

Il est possible de visualiser immédiatement le réseau « Hyposphere » en sélectionnant « **Draw** » dans le menu « **Draw** ». La première représentation graphique ainsi produite dispose de façon aléatoire les sommets du réseau sur le périmètre d'une ellipse :

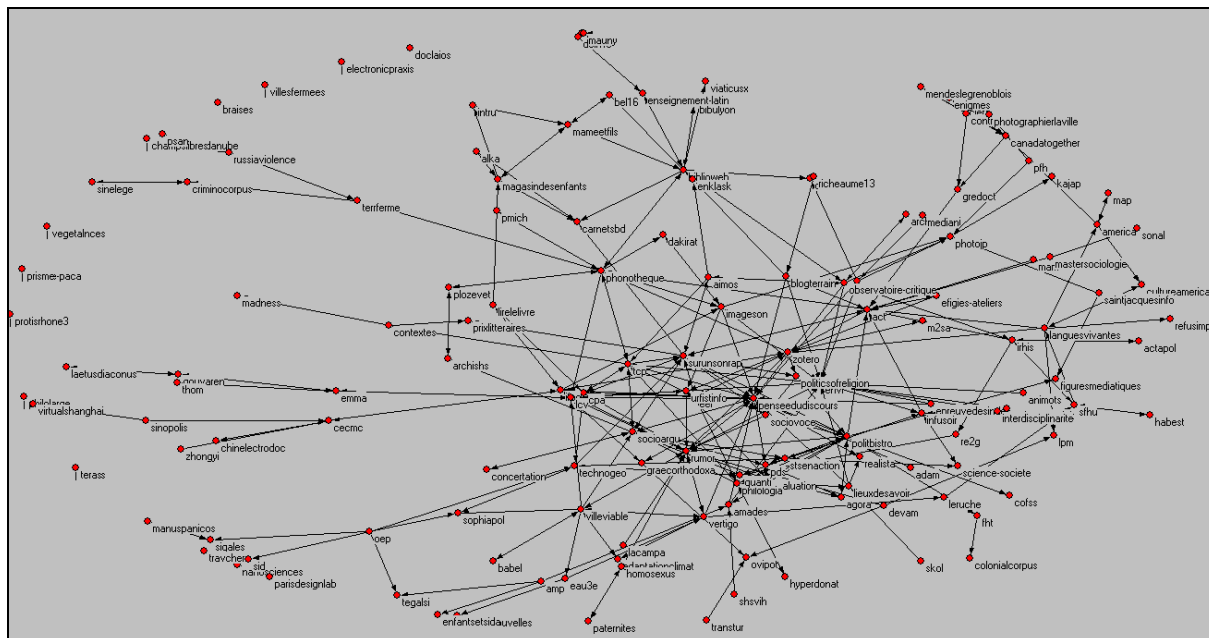


Il faut bien reconnaître que cette représentation graphique n'est pas facile à analyser... Heureusement, *Pajek* propose tout un ensemble de solutions...

La distribution des sommets

Se sont développés en analyse des réseaux des algorithmes de distribution des sommets permettant d'améliorer la visualisation des réseaux. Ces algorithmes cherchent en particulier à minimiser l'entrecroisement des liens. Le menu « **Layout** » de la fenêtre « **Draw** » vous permet d'explorer quelques uns de ces algorithmes proposés par *Pajek*. Ci-dessous, vous pouvez par exemple voir le résultat produit par le très classique algorithme de Fruchterman-

Reingold²⁰ (« **Layout > Energy > Fruchterman Reingold > 2D** »), qui traite le graphe comme si les sommets se repoussaient les uns les autres tout en étant reliés par des ressorts :



L'apparence des sommets

On a vu plus haut, à partir d'un exemple simple, qu'il était possible de faire varier la couleur des sommets en fonction de leurs propriétés, que celles-ci aient été indiquées au moment de la création du graphe ou qu'elles aient été calculées avec *Pajek*. Le vecteur ou la partition contenant ces propriétés peut aussi être récupéré ultérieurement. Dans l'exemple de l'Hyosphère, nous pouvons ainsi utiliser la fréquentation de chaque carnet pour faire varier la taille des sommets qui les représente. Cette information se présente sous la forme d'un « vecteur » :

[visites.vec](#)

Le fichier « visites.vec » contient pour chaque carnet le nombre de visites par mois entre janvier et avril 2011.

²⁰ Cet algorithme est implémenté dans la très grande majorité des logiciels d'analyse de graphes. Pour connaître les détails de son fonctionnement, voir Fruchterman et Reingold, 1991.

Après avoir ouvert ce fichier dans *Pajek* avec la commande « **Vector > Read** » du menu « **File** », vous pouvez en examiner les principales propriétés en sélectionnant la commande « **Vector** » du menu « **Info** » (ne tenez pas compte des options, cliquez sur « OK » à chaque fois) : vous verrez ainsi que la valeur maximum est de 25337 visites par mois, ce qui est une valeur beaucoup trop élevée pour pouvoir être utilisée directement comme diamètre des sommets. Il faut donc réduire ces valeurs, ce qui peut être obtenu par exemple en les divisant par 1000 : sélectionnez la commande « **Transform > Multiply by** » du menu « **Vector** », avec le coefficient multiplicateur 0.001.

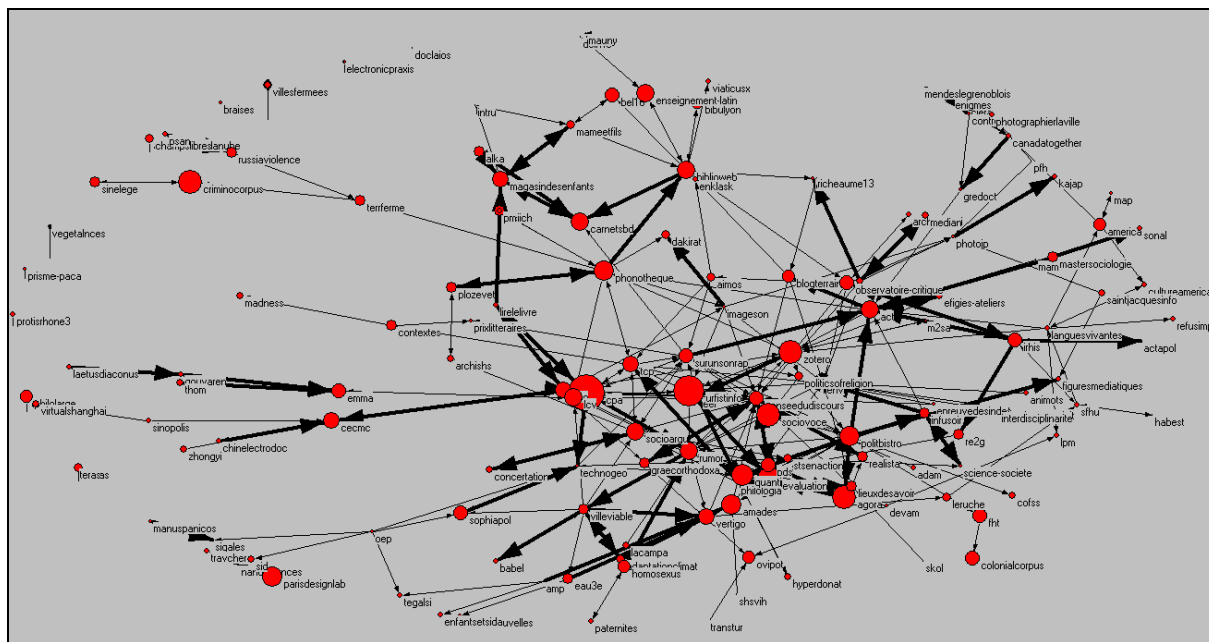
Une fois ce nouveau vecteur affiché dans la section « **Vectors** » de la fenêtre principale de *Pajek*, il ne vous reste alors plus qu'à redessiner le graphe avec la commande « **Draw-Vector** » du menu « **Draw** ».

L'apparence des liens

Dans notre exemple, en plus d'une variation de la taille des sommets en fonction de la fréquentation des carnets, nous pouvons faire varier l'apparence des liens entre eux en fonction du trafic qu'ils supportent, tel qu'il est indiqué en intensité des liens dans le fichier « *hyposphere.net* ». Pour cela, rendez-vous dans la fenêtre de représentation graphique où est déjà affiché le graphe avec la taille des sommets variable en fonction de la fréquentation (voir étape précédente, « L'apparence des sommets »), puis sélectionnez par exemple « **Lines > Different Widths** » dans le menu « **Options** »...

Le résultat n'est pas satisfaisant, parce que les trafics utilisés comme valences sont trop élevés pour être directement utilisés comme largeurs des liens. Nous choisissons ici de représenter par un trait plus épais les liens ayant une valence supérieure à 9 (autrement dit, supérieure ou égale à 10). Pour y parvenir, il faut d'abord transformer les valences avec la commande « **Line Values > Transform > Recode** » du menu « **Net** », avec en option « 10 » comme seule valeur de partition. On obtient ainsi un nouvel objet « Network », dans lequel environ 25% des liens (ceux qui avaient une intensité d'au moins 10) ont désormais une valence de 2, et les autres une valence de 1. Et si on utilise la commande « **Line Values > Transform > Power** » avec la valeur 2, on obtient finalement un réseau dans lequel les liens

ont désormais une valence de $2^2=4$ pour les trafics supérieurs ou égaux à 10, et $1^2=1$ pour les autres. Il ne reste plus qu'à le dessiner avec la commande « **Draw-Vector** » du menu « **Draw** », puis éventuellement à déplacer manuellement quelques sommets pour une meilleure lisibilité, et vous obtiendrez quelque chose comme ça :



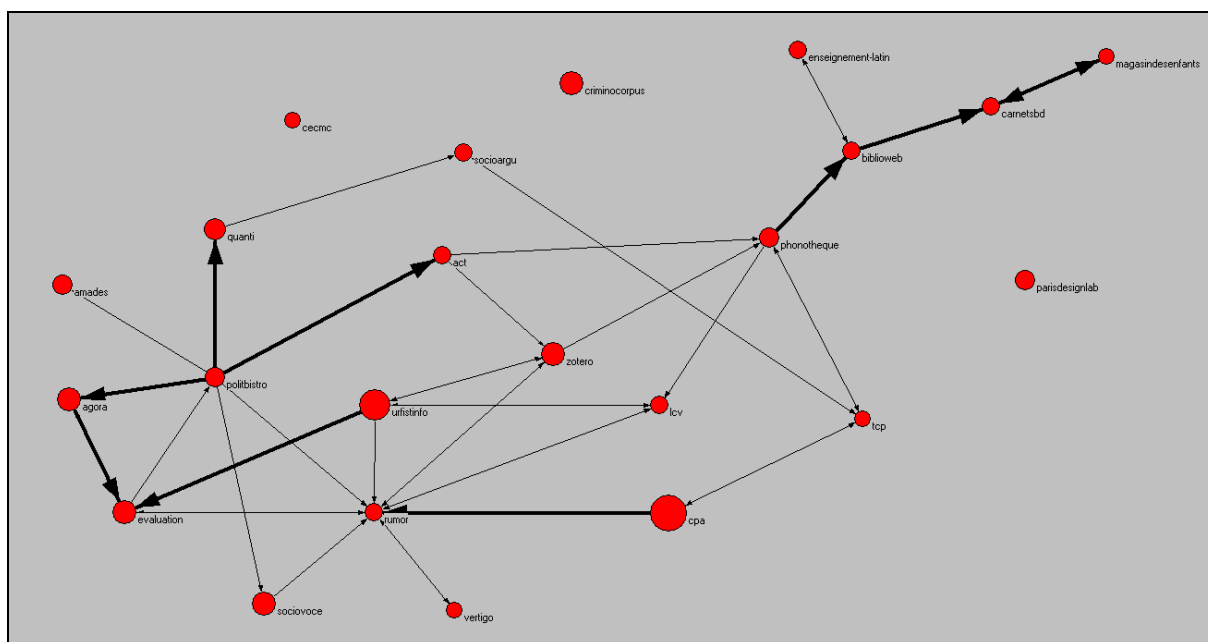
Manipuler et modifier les données

Notre exemple, celui de l'Hyposphère, contient 191 sommets et presque 350 liens. Même à l'aide des différentes possibilités offertes par *Pajek* en matière de visualisation, la représentation graphique de ce réseau reste assez complexe. On peut donc souhaiter modifier les données pour simplifier le graphe. *Pajek* offre plusieurs possibilités dans ce domaine, et permet notamment d'extraire un sous-graphe en fonction de certains critères, ou de remplacer un graphe par le graphe des liens entre sous-graphes constitués également en fonction de critères que vous pouvez déterminer.

Vous pouvez par exemple réduire le graphe en ne conservant que les liens dont la valence dépasse un certain seuil, avec la commande « **Transform > Remove > Lines with Values > Lower than** » du menu « **Net** ».

Une autre possibilité consiste à ne conserver que certains sommets, ce qui se fait à l'aide de la commande « **Extract from Network** » du menu « **Operations** ». Dans notre exemple, imaginons que nous souhaitons ne conserver que les carnets dont la fréquentation est supérieure à 5000 visites par mois (soit environ 23 carnets). Comme la commande « **Extract from Network** » ne reconnaît pas les vecteurs, il faut d'abord créer une partition contenant le critère qui nous intéresse, en appliquant au réseau de départ la commande « **Make Partition > By Intervals > Selected Thresholds** » du menu « **Vector** », en entrant « 5000 » comme valeur dans la fenêtre « **Dividing values or #Clusters** ». Ensuite, il suffit de sélectionner la partition obtenue, puis donc d'exécuter la commande « **Extract from Network** » du menu « **Operations** », avec « 2 » comme valeur dans la fenêtre « **Select Clusters** » (pour extraire les sommets appartenant à la classe 2 de la partition, celle regroupant les carnets ayant reçu plus de 5000 visites par mois).

Vous obtenez alors un nouvel objet « Network », qui ne contient plus que 23 sommets. Le problème, c'est que *Pajek* ne réduit pas automatiquement les partitions et les vecteurs correspondants. Par conséquent, si vous souhaitez également récupérer les informations qu'ils contiennent pour les sommets qui ont été conservés, il faut une opération supplémentaire. Par exemple, si vous voulez récupérer la fréquentation de ces 23 sommets (ou bien la fréquentation divisée par 100 dans la partie précédente, pour faire varier la taille des sommets), vous devez sélectionner la partition des 191 sommets produite à l'étape précédente (celle qui contient la valeur 1 pour les carnets à faible fréquentation, et 2 pour les carnets à forte fréquentation), le vecteur de la fréquentation, puis vous devez utiliser la commande « **Extract Subvector** » du menu « **Vector** », avec à nouveau « 2 » comme valeur dans la fenêtre « **Select Clusters** ». Et voilà le résultat, obtenu avec la commande « **Draw-Vector** »...



Conclusion : les atouts et les limites de *Pajek*

Pajek est un logiciel plutôt rapide et, une fois assimilée la logique de manipulation des trois principaux types d'objets (réseaux, partitions et vecteurs), son utilisation est plutôt simple. Toutefois, il n'est évidemment pas exempt de défauts, et on peut en pointer trois limitations principales :

- Le travail avec *Pajek*, qui repose essentiellement sur la manipulation de commandes et de menus, en fait une sorte de « clicodrome » très satisfaisant pour une initiation à l'analyse des réseaux, mais qui peut devenir rapidement très laborieux quand on souhaite appliquer les mêmes séries d'opérations complexes à plusieurs réseaux, même si *Pajek* offre une possibilité (un peu rudimentaire) d'enregistrer des séries d'opérations sous forme de macros ;
- Les algorithmes de visualisation disponibles sont peu nombreux : c'est la raison pour laquelle nous vous suggérons assez rapidement d'utiliser, en complément de *Pajek*,

d'autres applications de visualisation de réseaux, comme par exemple [NetDraw](#), [Gephi](#) ou en encore mieux [Tulip](#), qui sont du reste tous les trois gratuits ;

- Enfin, que ce soit en termes de mesures ou de recherche de structure, le choix des opérations et des algorithmes est également limité, excepté en ce qui concerne les mesures de distance.

Au final, *Pajek* demeure toutefois un outil très pratique pour les graphes de grande taille, même s'il demande parfois un travail assez long de mise en forme des données²¹ et ne permet que des analyses relativement standards. Quand vous sentirez que vous en atteignez les limites mentionnées ci-dessus, il sera alors toujours temps de mobiliser, en complément, d'autres applications, comme par exemple les différents packages de *R*²², auxquels nous essaierons de consacrer un prochain guide. En attendant, bon travail avec *Pajek* !

Liens utiles

Pajek

<http://pajek.imfm.si>

Guide d'utilisation de Pajek (PDF)

<http://pajek.imfm.si/lib/exe/fetch.php?media=dl:pajekman204.pdf>

Excel2pajek

<http://vlado.fmf.uni-lj.si/pub/networks/pajek/howto/excel2Pajek.htm>

Txt2pajek

<http://pajek.imfm.si/doku.php?id=faq:text2pajek>

Les guides du groupe fmr (flux, matrices, réseaux) sur HAL-SHS

<http://halshs.archives-ouvertes.fr/FMR/fr/>

²¹Apprendre à utiliser la fonction RECHERCHERV() disponible dans les tableurs permet un recodage relativement rapide des données. Sur ce sujet, voir par exemple l'article « Une fonction bien pratique pour fusionner des fichiers de données avec Excel » sur Quanti : <http://quanti.hypotheses.org/254/>.

²²Comme par exemple *SNA*, *StatNet*, *iGraph*, *Blockmodeling*...

Références bibliographiques

De Nooy Wouters, Mrvar Andrej et Batagelj Vladimir, 2005, *Exploratory social network analysis with Pajek*, New York, Cambridge University Press, coll. « Structural analysis in the social sciences »

Degenne Alain et Forsé Michel, 2004 [1994], *Les réseaux sociaux. Une approche structurale en sociologie*, Paris, Armand Colin, coll. « U », 2ème éd., 296 p.

Fruchterman T. et Reingold E., 1991, « Graph Drawing by Force-directed Placement », *Software - Practice and Experience*, 21(11), pp. 1129-1164

Lazega Emmanuel, 2007 [1998], *Réseaux sociaux et structures relationnelles*, Paris, Presses universitaires de France, coll. « Que sais-je ? »

Lemercier Claire et Zalc Claire, 2008, *Méthodes quantitatives pour l'historien*, Paris, La Découverte, coll. « Repères »

Mercklé Pierre, 2011 [2004], *Sociologie des réseaux sociaux*, Paris, La Découverte, coll. « Repères », 3ème éd., 128 p.

Wasserman Stanley et Faust Katherine, 1994, *Social Network Analysis. Methods and Applications*, Cambridge (Mass.), Cambridge University Press, coll. « Structural analysis in the social sciences », 825 p.