HAL
open science

MatrixExplorer: Un système pour l'analyse exploratoire de réseaux sociaux

Nathalie Henry, Jean-Daniel Fekete

To cite this version:

Nathalie Henry, Jean-Daniel Fekete. MatrixExplorer: Un système pour l'analyse exploratoire de réseaux sociaux. IHM 2006, AFIHM, 2006, Montréal, Canada. pp.67 - 74, 10.1145/1132736.1132745 . inria-00107308

HAL Id: inria-00107308

https://inria.hal.science/inria-00107308v1

Submitted on 18 Oct 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

MatrixExplorer: Un système pour l'analyse exploratoire de réseaux sociaux

Nathalie Henry

INRIA Futurs/LRI/University of Sydney
Bât 490, Université Paris-Sud
91405 Orsay Cedex
Nathalie.Henry@lri.fr

Jean-Daniel Fekete

INRIA Futurs/LRI
Bât 490, Université Paris-Sud
91405 Orsay Cedex
Jean-Daniel.Fekete@inria.fr

RESUME

Dans cet article, nous présentons le système MatrixExplorer destiné à explorer des réseaux sociaux. Il a été conçu pour des chercheurs en sciences sociales – sociologues et historiens – et propose une démarche itérative reposant sur des vues multiples.

MatrixExplorer est issu d'interviews avec des chercheurs et d'une session de conception participative qui ont permis de mettre en évidence des besoins tant en ce qui concerne les visualisations souhaitées que le processus d'exploration. MatrixExplorer fournit deux représentations pour les graphes : matricielle et en nœuds et liens. L'exploration part de la représentation matricielle qui reste toujours exploitable et offre une démarche itérative qui permet de filtrer et organiser le graphe afin d'en dégager une structure et une abstraction qui peut éventuellement se visualiser convenablement sous forme de nœuds et liens.

MOTS CLES : Réseaux Sociaux, Analyse Exploratoire, Visualisation de graphes, Représentation Matricielle, Réordonnancement, Clustering Interactif, Consensus.

ABSTRACT

In this article, we present the MatrixExplorer system, aimed at exploring social networks. It has been designed for researchers in social sciences – sociology and history – and is based on an interactive process with multiple views. MatrixExplorer design comes from several interviews with researchers and a participatory design session that helped us assess users need in term of visualization and exploration of social networks. MatrixExplorer provides two representations for graphs: matrices and node-link diagrams. Exploration starts from a matrix representation to preserve readability and provides an iterative process to filter and organize the graph to reveal a structure and abstract it so that it can eventually be visualized as a readable node-link diagram.

CATEGORIES AND SUBJECT DESCRIPTORS:
H.5 Graphical User Interfaces

GENERAL TERMS: Design.

KEYWORDS: Social networks, Exploratory Analysis, Graph Visualization, Matrix Visualization, Reordering, Interactive Clustering, Consensus.

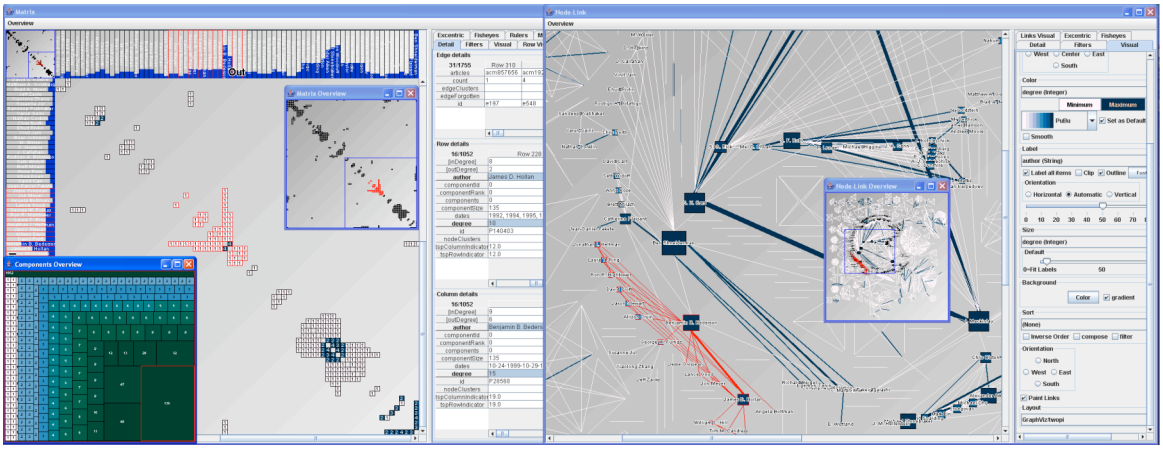

Figure 1 : MatrixExplorer

Réserver cet espace pour la notice de copyright

INTRODUCTION

La visualisation des réseaux sociaux est un domaine en pleine croissance depuis plusieurs années. La popularité des communautés en ligne et l'explosion de la quantité de données sont des facteurs qui ont générés cette effervescence. Cependant, si de nombreux systèmes et packages se sont développés dans le domaine de la visualisation d'information et du placement de graphes, aucun

n'est spécifiquement adapté aux besoins des chercheurs en sciences sociales et focalisé sur l'analyse exploratoire des réseaux sociaux. Dans cet article, nous présentons MatrixExplorer (*Figure 1*), un système conçu avec des sociologues et des historiens pour répondre à leurs besoins en terme d'exploration et d'analyse des réseaux sociaux. La philosophie du système est de construire interactivement de multiples points de vue sur les données. MatrixExplorer mixe donc les représentations et propose des outils d'interaction nouveaux pour explorer et confronter les multiples représentations d'un même réseau social. Nous présenterons un rapide panorama de la visualisation de réseaux sociaux avant de décrire nos préconisations pour un système exploratoire. Ces préconisations sont le résultat de séances de travail avec des utilisateurs réels. La deuxième partie de cet article décrit les caractéristiques de notre système MatrixExplorer. Nous expliquons pourquoi nous utilisons deux représentations de graphes : nœuds-liens et matrices et comment notre système répond aux besoins formulés par les utilisateurs avant de conclure sur de futurs travaux.

VISUALISATION DE RESEAUX SOCIAUX

La visualisation des réseaux sociaux est étroitement liée à la visualisation de graphes. Un réseau social est composé d'acteurs, en général des personnes, connectés par des relations. Il s'agit, par exemple, de communications téléphoniques ou bien dans le cas d'un arbre généalogique de liens de parenté. En théorie des graphes, les acteurs sont des sommets, les relations, des arêtes. Dans cet article nous utiliserons indistinctement réseau et graphe, acteur et sommet, relation et arête.

Il existe diverses techniques pour représenter les graphes, un panorama général est présenté dans [15]. Les deux visualisations les plus connues sont les représentations nœuds-liens et les représentations matricielles. Dans le domaine de l'analyse de réseaux sociaux, la visualisation a été employée comme outil d'analyse dès les années 1930. Un historique est présenté dans [11]. Les premières visualisations de réseaux sociaux ont été introduites par Moreno [16], il s'agissait alors de représentations nœuds-liens. Les matrices d'adjacence ont fait leur apparition par la suite [10]. [20, 23] exposent les concepts clés de l'analyse de réseaux sociaux et se base sur les deux représentations dont nous allons présenter un état de l'art: représentations nœuds-liens et matrices.

Représentation nœuds-liens

Un domaine entier de recherche est dédié aux représentations nœuds-liens : le domaine du dessin de graphe (graph drawing). Il s'agit de trouver de nouveaux algorithmes pour placer les nœuds et les liens d'un graphe dans l'espace ou le plan. Un panorama du domaine est présenté dans [4]. Aujourd'hui, un grand nombre de projets de visualisation de réseaux sociaux est dédié aux utilisateurs finaux. Il s'agit principalement d'outils de visualisation basés sur des graphes nœuds-liens. Ces projets visualisent, par exemple, les emails comme ContactMap [17] ou les communautés online comme Vizster [14]. Quelques visualisations mixtes apparaissent tel que Mat'Graph [18].

Dans cet article, nous nous focalisons sur des systèmes de visualisation et d'analyse des réseaux sociaux dédiés aux chercheurs en sciences sociales. Il s'agit de sociologues ou d'historiens par exemple.

De nombreux systèmes existent[1] mais l'état de l'art peut se diviser en deux catégories : les systèmes paramétrables et les systèmes « pousse-bouton ». Les systèmes paramétrables sont des systèmes perfectionnés dans lesquels l'utilisateur peut tout paramétrer du chargement du fichier à l'algorithme de placement en passant par l'algorithme de clustering. Ces systèmes ou packages proposent en général, un nombre de fonctionnalités impressionnant mais ne sont pas conçu pour faciliter l'interaction et ne guident pas l'utilisateur dans le processus d'exploration.

Pajek [3] est un très bon exemple de ces systèmes. Il est certainement le plus utilisé chez les sociologues mais demande beaucoup de connaissances pointues. Le processus de visualisation et d'analyse est laborieux, surtout pour des utilisateurs non spécialistes de la théorie des graphes ou des algorithmes de placement. UCINet[2] est un second exemple de ce type de systèmes, il propose des douzaines de mesures analytiques et statistiques. Ce système permet de valider plusieurs hypothèses et de répondre à un grand nombre de questions. Par contre, il n'est pas conçu pour explorer, c'est-à-dire susciter ces hypothèses et questions. Des systèmes commencent à favoriser le processus d'exploration. GUESS[3] par exemple, basé sur la boîte à outils JUNG[4], favorise l'exploration. Cependant, basé sur un langage de script (Jython), il est difficile à utiliser par un non-informaticien.

La deuxième catégorie de systèmes est la catégorie « pousse-bouton ». Il s'agit d'algorithmes qui proposent un placement ou une visualisation unique, sans paramétrage, ou bien de systèmes ou packages qui proposent un catalogue d'algorithmes ou de visualisations. Par exemple, GraphViz [12], JUNG ou TULIP [2] fournissent un grand nombre d'algorithmes de placement de graphes. Ces systèmes sont indispensables pour la visualisation de graphes, ils permettent l'exploration par essai-erreur. Par contre, ils ne guident pas l'utilisateur dans le processus d'exploration et proposent uniquement des représentations nœuds-liens.

Représentations matricielles de graphes

La représentation matricielle des graphes est une représentation beaucoup moins utilisée que les diagrammes nœuds-liens. Bien que moins intuitives, les matrices ont d'intéressantes propriété que Jacques Bertin a tenté de formaliser dans [5]. Elles permettent notamment de passer à l'échelle et de traiter de très grands graphes [1].

---

[1] http://www.insna.org/INSNA/soft_inf.html
[2] http://www.analytictech.com/
[3] http://www.hpl.hp.com/research/idl/projects/graphs/
[4] http://jung.sourceforge.net/

Pourquoi utiliser les représentations matricielles ?

Les deux représentations, nœuds-liens et matrices, ont été expérimentalement comparées dans Ghoniem et al. [13]. Ils ont défini un ensemble de tâches de bas niveau telles qu'évaluer le nombre de sommets du graphe, trouver un sommet, déterminer si deux sommets sont connectés, etc. En utilisant ces tâches, ils ont démontré expérimentalement que les matrices sont plus performantes sur les graphes dont le nombre de sommets est supérieur à 20 et/ou dense pour toutes les tâches de bas niveau excepté le suivi de chemin.

Outre l'absence d'occlusion des arêtes du graphe, qui reste un problème dans le dessin des représentations noeuds-liens, les représentations matricielles permettent également de détecter les informations qui n'existent pas ! Effectivement, pour les sociologues il est important de voir qu'une clique n'est pas totale et d'identifier rapidement les acteurs non communicants.

Dans le cadre de l'analyse exploratoire et interactive de graphes, le temps pour afficher la représentation à l'écran est un facteur déterminant. L'avantage des matrices sur ce point est que le temps pour les afficher est prédictible (contrairement aux algorithmes de placement de graphes masses-ressorts par exemple) et qu'il est possible de les afficher de façon acceptable sans calculer de placement pour chacun des sommets. Par acceptable, nous entendons qu'il est possible d'effectuer toutes les tâches de base même si certaines sont laborieuses.

Il est possible d'améliorer la lisibilité des matrices en les réordonnant, c'est-à-dire en permutant leurs lignes et leurs colonnes. Le temps de calcul du réordonnancement peut être instantané ou prendre quelques secondes selon la méthode utilisée.

Ordonner et réordonner une matrice

Jacques Bertin a montré dans [5] qu'une matrice dont les lignes et les colonnes sont réordonnées judicieusement permet d'identifier rapidement des groupes d'éléments tout en caractérisant des cas particuliers.

Par exemple, la *Figure 2* présente une table de valeurs numériques donnant la production de cinq viandes pour cinq pays ainsi que sa représentation visuelle, dont les colonnes et les lignes ont été réordonnées. Cette représentation permet de dégager d'un coup d'œil trois classes de pays ayant des profils similaires (A, B et C sur la *Figure 2*). Il aurait été beaucoup plus long et fastidieux de détecter ces groupes directement sur la table de valeurs alors qu'elle est pourtant très petite. Réordonner une table est donc une fonctionnalité indispensable pour la visualisation matricielle. Il s'agit d'un problème étudié depuis plusieurs années. Réordonner une matrice est évidemment similaire au réordonnancement de tables mais aussi à la sériation de graphe. De très nombreuses méthodes existent. Dans un premier temps, nous nous sommes intéressés aux réordonnancements de tables.

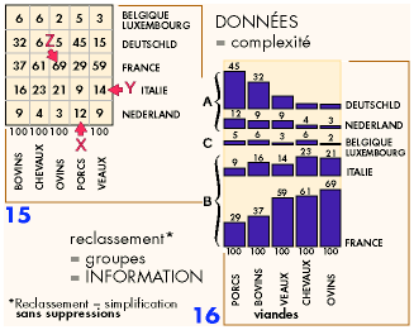

Figure 2 : Table de valeurs de production de viandes et la matrices réordonnée correspondante (Bertin 67)

L'état de l'art peut se diviser en deux grandes catégories : les méthodes automatiques et les systèmes interactifs. Parmi les systèmes interactifs, nous pouvons citer TableLens [19] ou InfoZoom [22]. Ces systèmes permettent de réordonner une table rapidement en les triant par une colonne. Par contre ils n'offrent pas de réordonnancement d'une table entière. Des méthodes de réarrangement automatique ont été explorées très tôt par Chauchat [6] et apparaissent à la fin de [5]. Il existe un certain nombre d'heuristiques [21] et également plusieurs méthodes en bioinformatique [7, 8]. En conclusion, ordonner manuellement une matrice est très laborieux tandis que l'ordonner automatiquement est généralement insuffisant et requiert une amélioration manuelle : nous ne disposons pas de formalisation de ce qu'est une configuration optimale !

BESOINS POUR L'ANALYSE EXPLORATOIRE

A l'issue de séances de conception participative et de plusieurs entretiens avec des sociologues, des historiens, et des enseignants en analyse des réseaux sociaux de l'EHESS[5] et de l'INED[6], nous avons dressé une liste des besoins pour un système d'analyse exploratoire des réseaux sociaux.

- *Multiples représentations* Un réseau social est un jeu de données complexe. En général, les acteurs ont plusieurs attributs et les relations également. Offrir plusieurs représentations permet d'avoir différents points de vue sur les données. En sciences sociales, les chercheurs sont familiers des représentations nœuds-liens et matricielles, il est donc intéressant de bénéficier des avantages de chacune. Visualiser le graphe entier en utilisant la matrice et chercher si deux nœuds sont connectés en utilisant la représentation nœuds-liens.
- *Organisation, Placement* Qu'il s'agisse de placer les sommets et les arêtes d'un nœuds-liens ou bien d'ordonner les lignes et les colonnes d'une matrice, cette fonctionnalité est un problème très complexe mais est essentielle pour l'exploration.
- *Gestion des composantes connexes* Un graphe réel est un ensemble de composantes connexes, c'est-à-dire qu'il est composé de sous-graphes non connec-

[5] Ecole des Hautes Etudes en Sciences Sociales
[6] Institut National des Etudes Démographiques

tés. Gérer ces composantes connexes est indispensable pour les sociologues afin qu'ils puissent charger leur réseaux sociaux réels.

- *Vue d'ensemble* La vue d'ensemble est un défi lorsqu'il s'agit d'un grand graphe, cependant elle est très importante dans le processus d'exploration. Tout d'abord, elle permet à l'utilisateur de commencer l'exploration et de se forger une première carte mentale du graphe. Ensuite, elle est un point d'ancrage lors de la navigation.
- *Attributs du graphe* Un des points revenant souvent au cours des séances de travail est le besoin d'avoir les données générales du réseau, i.e. quels attributs sont disponibles pour les acteurs et pour les relations. Par attribut, nous entendons, les informations sur les acteurs et les relations. Par exemple, le nom et l'âge pour les acteurs, le type et l'intensité pour une relation. En cela, la visualisation d'information diverge du placement de graphes. Effectivement, ce dernier s'intéresse à la topologie, la structure du réseau social, alors que la visualisation d'information se focalise également sur les attributs des éléments du réseau.
- *Données du graphe* Pour les chercheurs en sciences sociales, les données sont extrêmement importantes. Il ne s'agit pas seulement de dessiner un graphe et de montrer sa structure. Il faut leur permettre de consulter facilement toutes les informations sur les acteurs et les relations et de les visualiser de façon adéquate (choix des variables visuelles).
- *Données issues de l'analyse de réseaux* Proposer un ensemble de caractéristiques principales du réseau social est important. Visualiser et explorer n'excluent pas de proposer des données issues de l'analyse des réseaux. Ainsi, les sociologues se sont prononcés en faveur d'un ensemble de caractéristiques à afficher décrivant le réseau. Nous pouvons citer principalement le nombre d'acteurs et de relations, la distribution des degrés, le diamètre du réseau, les 5 acteurs les plus connectés et plusieurs mesures de centralité.
- *Interaction vs Paramétrage* Dans un système exploratoire, le paramétrage n'a pas vraiment sa place. Les sociologues, peu familiers des algorithmes de placement de graphes ou de clustering (regroupement des sommets similaires), ne savent pas les paramétrer. Mieux vaut alors s'orienter vers des algorithmes « pousse-bouton » et proposer de l'interaction directement sur le graphe. D'autant plus qu'interagir directement avec le graphe permet de le comprendre et de le mémoriser, d'après nos observations.
- *Filtrage* Pour de grands réseaux sociaux, le filtrage permet d'analyser des sous-parties du réseau. Il faut rester prudent et rappeler à l'utilisateur que les données filtrées existent toujours.
- *Cas Particulier* Les cas particuliers sont souvent le centre d'intérêt pour les sociologues. Par exemple, il s'agit de comprendre pourquoi deux acteurs de communiquent pas au sein d'une communauté presque totalement connectée (une quasi-clique).
- *Groupement* Dans l'analyse des réseaux sociaux, la détection des communautés (ou groupes, clusters) tient une grande place. Il existe de nombreuses méthodes pour regrouper des acteurs en communauté, cette fonctionnalité est indispensable pour l'exploration.
- *Agrégation* Agréger les communautés, c'est-à-dire transformer un ensemble d'acteurs en un seul, est une fonctionnalité intéressante qui pose plusieurs problèmes. Le plus épineux est de maintenir la cohérence entre plusieurs niveaux de détails et de garder un maximum d'informations en traitant correctement les attributs quand c'est possible.
- *Trouver un consensus* Les sociologues disposent souvent de plusieurs types de relations ou d'informations sur les acteurs. Lorsqu'il s'agit de grouper les acteurs en communauté, il est important de pouvoir trouver un consensus ou d'identifier que des groupes différents selon les attributs. C'est également important lorsqu'il s'agit de déterminer si un acteur est un cas particulier ou non. Cette problématique est liée aux multiples points de vue qu'un utilisateur peut avoir sur un jeu de données. Il doit être possible d'utiliser de multiples représentations, de les réorganiser, de les sauvegarder, de les confronter afin de déterminer celles qui semblent pertinentes.

Le but d'un tel système est de se forger rapidement une compréhension du réseau la plus proche de la réalité possible. Pour ce faire, il faut faciliter le processus d'exploration et guider l'utilisateur dans ses découvertes.

MATRIXEXPLORER

MatrixExplorer est un système exploratoire de réseaux sociaux qui répond aux besoins explicités plus haut. Nous avons conçu ce système en privilégiant le processus d'exploration et en proposant des outils d'interaction nouveaux et adaptés. Nous allons décrire dans cette section les fonctionnalités du système. Nous utiliserons deux jeux de données réels pour illustrer les fonctionnalités de MatrixExplorer : le graphe de collaboration scientifique issu du concours de la conférence InfoVis en 2004 et le graphe biparti des résultats du master recherche de l'Université Paris-Sud en 2005.

Multiples représentations

La philosophie de MatrixExplorer est d'offrir de multiples représentations d'un même réseau social. MatrixExplorer mixe les deux représentations nœuds-liens et matricielles. Les visualisations peuvent être synchronisées. Il est possible de les dupliquer à volonté et de les manipuler ensemble ou indépendamment.

Gestion des composantes connexes

La décomposition en composantes connexes est indispensable lorsque les données sont réelles. Elle fournit déjà des informations précieuses sur la structure du graphe et permet de guider l'utilisateur dans le processus

d'exploration. La *Figure 3* représente les composantes connexes du graphe de collaboration scientifique des chercheurs en visualisation d'information. Cette visualisation sous forme de Treemap permet d'identifier d'un coup d'œil le nombre de composantes connexes, leurs tailles et permet de débuter l'analyse du réseau social. Les composantes connexes sont analysables indépendamment les unes des autres, c'est pour cette raison que MatrixExplorer synchronise la vue d'ensemble des composantes avec les visualisations nœuds-liens et matricielles. Un clic sur une composante connexe filtre automatiquement les visualisations associées.

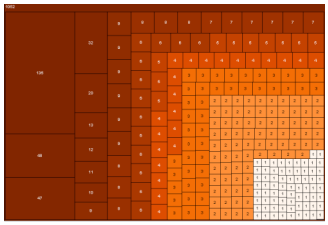

Figure 3: Représentation des composantes connexes du graphe de collaboration

Vues d'ensemble

La vue d'ensemble est un défi lorsqu'il s'agit d'un graphe. Pour cette raison, MatrixExplorer en propose plusieurs et mixe nœuds-liens et matrices.

La première constatation après observation de la *Figure 4* est que les vues d'ensemble, qu'elles soient nœuds-liens ou matricielles, sont très différentes alors qu'il s'agit d'un même graphe. Pour les représentations nœuds-liens, l'algorithme de placement de graphe est déterminant. Pour les représentations matricielles, l'ordre des lignes et des colonnes est déterminant. Dans les deux cas, déterminer quel algorithme est le plus intéressant pour l'utilisateur est un problème complexe. Alors qu'il est rare qu'une représentation nœuds-liens soit immédiatement exploitable après chargement d'un grand graphe, la représentation matricielle permet elle de rester lisible et fournit des informations de base sur le nombre de sommets et la densité du graphe. Par contre, après filtrage et groupement ou, lorsqu'il s'agit d'un petit graphe, la vue d'ensemble nœuds-liens est souvent préférée par l'utilisateur.

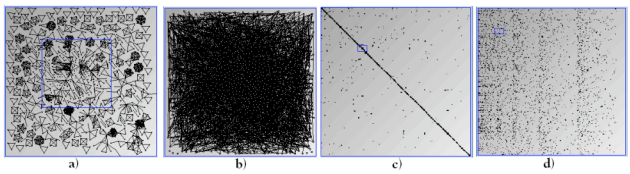

Figure 4: Vues d'ensemble nœuds-liens et matricielles du graphe d'InfoVis. a) Neato de GraphViz, b)Algorithme masses-ressorts, c)Ordre de construction, d)Ordre alphabétique en lignes, ordre croissant des degrés en colonnes.

Nous avons fait le choix, dans MatrixExplorer, de toujours proposer les deux représentations. Par défaut, l'algorithme de placement de graphe utilisé pour la représentation nœuds-liens est un algorithme qui gère au mieux l'affichage des différentes composantes connexes et qui ne nécessite un temps de calcul raisonnable. Nous avons choisi Twopi, algorithme de complexité linéaire fournit par GraphViz. Pour la représentation matricielle, la stratégie est d'ordonner les lignes et les colonnes selon l'ordre de chargement des sommets du graphe. Cet ordre est généralement intéressant pour l'utilisateur, préféré en tout cas à un ordre aléatoire ou bien basé sur un attribut déterminé par le système. L'utilisateur peut ensuite débuter le processus d'exploration. MatrixExplorer lui permet de dupliquer à sa convenance les représentations (nœuds-liens ou matricielles), de choisir des algorithmes de placement ou réordonnancement, de filtrer et grouper des éléments de son graphe.

Visualisation et Interaction de base

MatrixExplorer est basé sur la toolkit InfoVis [9]. Cette toolkit permet de bénéficier d'un certain nombre de fonctionnalités de base mais néanmoins indispensables à l'utilisation d'un système de visualisation d'informations par des utilisateurs réels.

Le premier point est de présenter les données sur le réseau social chargé. InfoVis permet de faire cela automatiquement, qu'il s'agisse des sommets ou des arêtes d'un graphe. Les attributs sont présentés sous la forme d'un tableau. Lorsqu'un acteur ou une relation est sélectionné (par un clic) alors les informations sont affichées dans ce même tableau. Le second point concerne l'attribution des variables visuelles (couleur, forme, taille,...) aux différents attributs du graphe. InfoVis construit automatiquement des éléments d'interface (liste déroulante, cases à cocher, …) permettant d'effectuer cette attribution. Enfin, le dernier point important concerne le filtrage, fournit lui aussi par InfoVis. Il est ainsi possible de filtrer selon n'importe quel attribut calculé ou non, d'un acteur ou d'une relation.

Placement de graphe

Il existe un très grand nombre d'algorithmes de placement de graphes. Plusieurs packages existent, InfoVis utilise principalement GraphViz mais d'autres packages ou algorithmes sont faciles à ajouter. MatrixExplorer se base donc sur quelques algorithmes de base pour afficher les représentations nœuds-liens. Le placement du graphe est sélectionné dans une liste déroulante par l'utilisateur et une interaction par glisser-déposer est proposée pour déplacer les nœuds du graphe.

Réordonner une matrice automatiquement

Il existe un large panel de méthodes de réordonnancement automatique que nous pouvons appliquer aux matrices. La première catégorie concerne les méthodes appliquées sur des tables, la seconde concerne les méthodes de sériation de graphes. Nous nous sommes intéressés à deux méthodes appliquées sur des tables. Ces méthodes sont issues de la bioinformatique et sont utilisées pour réordonner des données de puces ADN. Nous les avons choisi pour deux raisons : leur quasi-absence de paramétrage et leur rapidité d'exécution. Ceci nous permet de les utiliser dans un processus exploratoire, de façon interactive. Les deux méthodes calculent initialement une matrice de similarité entre les lignes et les colonnes. La première méthode (CL) est basée sur un clus-

tering hiérarchique à partir de ces similarités, suivi d'une linéarisation des feuilles selon la méthode décrite dans [8]. La seconde méthode est basée sur la résolution du problème du voyageur de commerce (TSP), consistant à trouver un cycle passant par toutes les lignes (resp. colonnes) et dont la somme des distances entre deux lignes (resp. colonnes) consécutives est minimale. La méthode est présentée dans [7]. Afin de pouvoir intégrer ces méthodes dans MatrixExplorer, nous les avons adaptées. Tout d'abord, nous les avons appliquées aux matrices d'adjacence, ensuite nous avons modifié leurs paramètres pour les appliquer dans les deux dimensions et finalement nous avons rendu le processus applicable sur des sous-matrices afin de pouvoir utiliser le réordonnancement automatique de façon interactive. La matrice d'adjacence ne contient pas assez d'informations pour que les algorithmes de réordonnancement soient efficaces car elle est généralement creuse. Dans une première étape, nous ajoutons de l'information en utilisant la matrice des distances, qui donne les plus courts chemins entre chacun des sommets du graphe. La deuxième étape consiste à trouver une similarité ou métrique qui permette d'ordonner la matrice dans les deux dimensions. Il faut une métrique de distance ou de similarité pour les lignes (resp. les colonnes) qui soit indépendante de l'ordre des colonnes (resp. des lignes). Nous utilisons la distance de Manhattan par défaut.

Réordonner une matrice interactivement

En plus des deux méthodes de réordonnancement automatique qui s'appliquent à la matrice entière, MatrixExplorer propose également un ensemble d'outil pour que l'utilisateur améliore interactivement le réordonnancement de la matrice.

Les interactions de base sont le déplacement d'une ligne, d'une colonne ou d'un groupe de lignes/colonnes. Ces opérations s'effectuent par glisser-déposer. L'utilisateur sélection directement les lignes, les colonnes ou un groupe de cases sur la matrice. La seconde fonctionnalité de base est inspirée des logiciels de traitement de tables. Elle permet de trier les lignes selon un attribut. Contrairement à InfoZoom ou TableLens, MatrixExplorer trie la matrice en deux dimensions : un attribut pour les lignes, un pour les colonnes. Il peut s'agir d'un attribut de base des acteurs ou bien d'un attribut calculé. Il est possible de choisir un ordre différent pour les lignes et les colonnes afin d'explorer rapidement l'influence d'un attribut sur un autre. Par exemple, la matrice des co-auteurs en visualisation d'information peut être réordonnée en lignes par le nom des acteurs (attribut de base) et en colonnes par le degré des acteurs (attribut calculé). Ainsi, il sera possible de rapidement trouver un acteur en utilisant les lignes triées par ordre alphabétique et d'identifier également les acteurs ayant beaucoup de relations en utilisant les colonnes triées par degré. Notons que si un attribut n'est pas défini pour certains des acteurs, ceux-ci sont alors placés en fin de matrice. Nous avons également développé un outil spécifique à la méthode d'ordonnancement utilisant TSP. Il s'agit d'un outil de permutation particulier: l'outil « main ». Etant donné que les heuristiques de TSP donnent en général des boucles et non des chemins, cet outil permet de permuter cycliquement les lignes ou les colonnes.

Un premier outil utilise une technique automatique de réordonnancement sur une sous-partie de la matrice. Un utilisateur peut alors sélectionner soit un groupe de lignes, soit un groupe de colonnes, soit un groupe de cellules et demander le réordonnancement de cette partie de la matrice uniquement. En utilisant TSP, la sous-matrice est alors réordonnée en lignes et en colonnes. Ceci permet d'explorer interactivement les parties de la matrice qui intéressent l'utilisateur. L'inconvénient est que certaines parties ordonnées peuvent être modifiées en fonction de la sous-matrice sélectionnée (*Figure 5*).

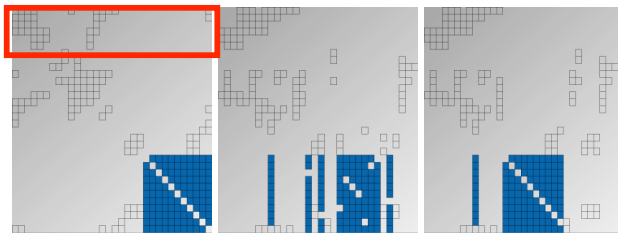

Figure 5 : A gauche matrice initiale. En rouge sous-matrice à réordonner, en bleu groupe à conserver. Au milieu sans blocage, à droite avec

Le résultat obtenu après réarrangement dans la *Figure 5* (milieu) est un artefact de TSP mais aisément explicable. La sous-matrice sélectionnée contient un grand nombre de colonnes vides. La métrique qu'utilise TSP ne permet pas de les distinguer, donc l'algorithme les affiche dans un ordre aléatoire. Pour palier ce problème, MatrixExplorer propose un système de masque ou de verrou. Ainsi il est possible de bloquer un certain nombre de lignes ou de colonnes ensembles. Il est envisageable d'apposer cette contrainte plus ou moins fermement : soit l'ordre donné par TSP est modifié dans son ensemble, soit il est modifié partiellement, uniquement pour les lignes et colonnes qui n'ont aucun impact sur la sous-matrice. Nous avons choisi la seconde solution car nous privilégions l'ordre optimal de la sous-matrice. Les lignes/colonnes non différenciables par TSP sont alors ordonnées pour conserver au maximum les lignes/colonnes bloquées ensembles (*Figure 5* droite).

Le second outil permet de réordonner la matrice « sans tenir compte » d'une sous-matrice donné. Cet outil filtre une partie de la matrice (lignes, colonnes ou groupes de lignes/colonnes) sans ôter les informations du champ de vision de l'utilisateur et réordonne la matrice sans en tenir compte. L'utilisateur peut ainsi garder ses points de repère et visualiser le réseau en supprimant l'influence de tel acteur, telle relation ou tel communauté.

Grouper automatiquement

Plusieurs algorithmes de regroupement ou clustering existent. Le but de MatrixExplorer n'est pas d'être le plus complet possible ou de proposer un paramétrage fin de ces algorithmes. Nous avons donc opté pour une découverte interactive des groupes. Etant donné que nous

utilisons une méthode de réordonnancement matricielle basée sur un clustering hiérarchique, nous proposons à l'utilisateur de visualiser l'arbre, ce qui peut fournir des informations complémentaires pour choisir ses groupes.

Grouper interactivement

Il ne s'agit pas seulement de réordonner pour grouper, il faut ensuite donner la possibilité à l'utilisateur d'identifier les groupes, de les visualiser et de leur donner un nom. MatrixExplorer permet d'entourer des groupes directement sur la matrice en utilisant deux modes : le mode sélection de cellules et le mode lasso.

Le mode sélection de cellules utilise les fonctionnalités classiques de sélection (clic et glisse, shift, contrôle) afin de sélectionner un groupe de lignes (ou de colonnes) en cliquant sur leurs étiquettes ou un ensemble de cellules (cases de la matrice). L'utilisateur attribue ensuite une couleur au groupe et peut saisir un nom.

Le mode sélection par lasso permet à l'utilisateur un groupement plus spontané et moins précis. Il convient mieux au processus d'exploration qui consiste à repérer des groupes potentiels sans réelle précision. Le principe est d'utiliser un lasso, c'est-à-dire un outil de sélection à main levée qui permet à l'utilisateur de tracer une boucle pour désigner un nouveau groupe. Les éléments à l'intérieur de la boucle sont considérés comme éléments effectifs du groupe alors que les éléments traversés par le lasso ont une appartenance moins marquée au groupe.

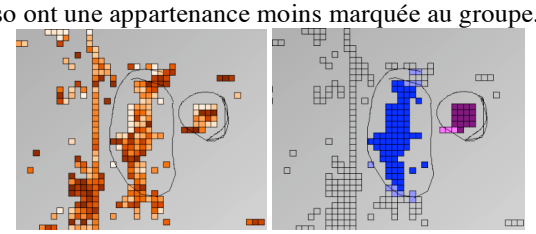

Figure 6 : Sélection approximative avec un lasso, à droite les valeurs, à gauche les clusters.

Une fois les groupes identifiés, se pose le problème de les visualiser sans complexifier la représentation. Nous avons opté pour une visualisation similaire au mode X-Ray de Vizster. L'utilisateur demande temporairement d'afficher l'appartenance des éléments aux groupes. Prenons l'exemple de la matrice des masters recherche (*Figure 6*). L'utilisateur a choisi d'afficher les notes des étudiants sur les relations en utilisant la variable visuelle des couleurs. Afficher les groupes ne nécessite pas d'utiliser un autre attribut visuel, ce qui pourrait d'une part surcharger la visualisation, d'autre part ne pas être facilement identifiable. L'utilisateur choisit simplement de passer dans le mode visualisation de groupes (X-Ray), la variable visuelle des couleurs est alors utilisée pour identifier les groupes. Le mode est activé lorsque l'utilisateur maintient un bouton ou une touche du clavier enfoncée. Ainsi, l'utilisateur conserve la carte mentale de la représentation, les informations sur les notes et peut rapidement et facilement identifier les différents groupes. Enfin, MatrixExplorer permet de nommer ces groupes. A chaque création d'un nouveau groupe, un élément est ajouté dans le tableau des groupes. Cet élément a un nom et une couleur par défaut mais est éditable par l'utilisateur. Celui-ci peut alors lui choisir un nom plus approprié et la couleur qui lui convient.

Confronter les groupes

Lors du processus d'exploration, un utilisateur identifie plusieurs groupes et plusieurs cas particuliers. MatrixExplorer permet de confronter ces découvertes. Il s'agit de déterminer si il y a des invariants pour trouver un consensus. Dans le cas des matrices, il est intéressant de vérifier qu'un groupe identifié par l'utilisateur pour un réordonnancement donné, reste un groupe dans une autre configuration. Prenons de nouveau l'exemple du réseau des master recherche. Dans la *Figure 7*, un utilisateur a utilisé le réordonnancement automatique clustering+linéarisation et a identifié 5 groupes (de gauche à droite) : étudiants de l'école A, étudiants du cursus réseaux, tronc commun, étudiants du cursus sciences cognitives et étudiants de l'école B.

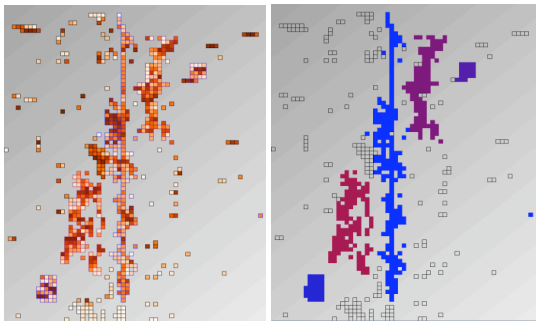

Figure 7 : Matrice des Master Recherche ordonnée en utilisant CL . A gauche les valeurs, à droite les groupes

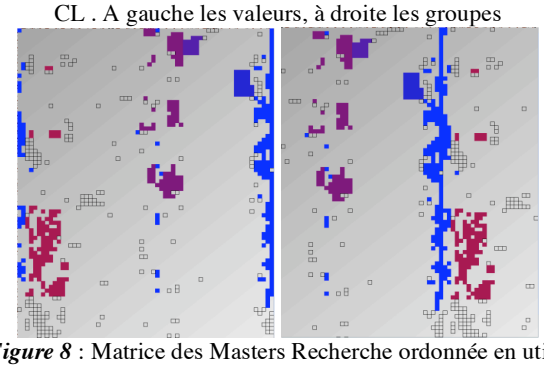

Figure 8 : Matrice des Masters Recherche ordonnée en utilisant TSP. A droite, l'outil Main a été utilisé pour mettre en évidence le groupe bleu.

En réordonnant la matrice d'une deuxième façon, en utilisant TSP, l'utilisateur veut vérifier qu'il retrouve bien ces mêmes groupes. Il veut donc savoir s'il y a consensus entre les deux méthodes automatiques de réordonnancement. TSP est présenté dans la *Figure 8*. En un coup d'œil l'utilisateur identifie qu'un groupe est coupé en deux, ce qui n'est pas étonnant car la solution proposée est un cycle. Il utilise donc l'outil Main et permute les colonnes de la matrice. Finalement, l'utilisateur retrouve ses 5 groupes de départ en remarquant cependant que le groupe des étudiants en sciences cognitives a éclaté dans cette représentation. Ayant trouvé un consensus entre deux représentations, l'utilisateur peut être plus confiant sur le fait que le tronc commun et les étudiants du cursus réseaux sont une tendance réelle de son jeu de

données. En construisant de nouvelles représentations et en confrontant les découvertes, MatrixExplorer permet de découvrir les multiples facettes de son jeu de données.

CONCLUSION

Nous avons présenté un système d'analyse exploratoire des réseaux sociaux : MatrixExplorer. Ce système a été développé pour répondre aux besoins que nous avons définis avec des utilisateurs réels : des sociologues, des historiens et des enseignants en analyse des réseaux sociaux. Dans cet article, nous avons décrit le processus d'exploration et nous avons spécifié chacune des fonctionnalités nécessaires à l'élaboration d'un système répondant aux besoins des chercheurs en sciences sociales. Nous envisageons de poursuivre notre collaboration avec ces chercheurs afin de valider et d'améliorer MatrixExplorer. Une étude longitudinale permettra de comprendre comment ils exploitent le système et ses fonctionnalités sur leurs propres données. Cela permettra également de collecter des informations sur le processus d'exploration dans un cadre réel. Concernant le système, nous allons implémenter les fonctionnalités d'agrégation des acteurs dans les deux représentations. Nous envisageons également d'ajouter à MatrixExplorer les mécanismes d'animation et d'évaluer leurs impacts sur le processus d'exploration.

BIBLIOGRAPHIE

1. Abello, J. and Ham, F.v. Matrix Zoom: A Visual Interface to Semi-External Graphs in Proceedings of the IEEE Symposium on Information Visualization (INFOVIS'04) - Volume 00 IEEE Computer Society, 2004 183-190
2. Auber, D. Tulip: A huge graph visualization framework. in Mutzel, P. and Jünger, M. eds. Graph Drawing Softwares, Springer-Verlag, 2003, 105-126.
3. Batagelj, V. and Mrvar, A. Pajek - Analysis and Visualization of Large Networks. in Jünger, M. and Mutzel, P. eds. Graph Drawing Software, Springer, Berlin, 2003, 77-103.
4. Battista, G.D., Eades, P., Tamassia, R. and Tollis, I.G. Graph Drawing: Algorithms for the Visualization of Graphs. Prentice Hall PTR, 1998.
5. Bertin, J. Sémiologie graphique : Les diagrammes - Les réseaux - Les cartes. Editions de l'Ecole des Hautes Etudes en Sciences, Paris, France, 1967.
6. Chauchat, J.H. and A., R., AMADO, a new method and a software integrating Jacques BERTIN's Graphics and Multidimensional Data Analysis Methods. in International Conference on VIsualization of Categorical Data, (Cologne, R.F.A, 1995).
7. Climer, S. and Zhang, W., Take a walk and cluster genes: a tsp-based approach to optimal rearrangement clustering. in Proceedings of the twenty-first international conference on Machine learning, (Banff, Alberta, Canada, 2004), ACM Press, 22.
8. Eisen, M.B., Spellman, P.T., Brown, P.O. and Botstein, D. Cluster Analysis and Display of Genome-Wide Expression Patterns. Proc. Natl. Acad. Sci. USA, 95. 1998, 14863-14868.
9. Fekete, J.-D. The InfoVis Toolkit in Proceedings of the IEEE Symposium on Information Visualization (INFOVIS'04) - Volume 00 IEEE Computer Society, 2004 167-174
10. Forsyth, E. and Katz, L. A Matrix Approach to the Analysis of Sociometric Data, Preliminary Report. Sociometry, 9. 1946, 340-347.
11. Freeman, L. Visualizing Social Networks. Journal of Social Structures, 1. 2000, http://cmu.edu/joss.
12. Gansner, E.R. and North, S.C. An open graph visualization system and its applications to software engineering. Software - Practice and Experience, 30 (11), 2000, 1203-1233.
13. Ghoniem, M., Fekete, J.-D. and Castagliola, P. On the readability of graphs using node-link and matrix-based representations: a controlled experiment and statistical analysis. Information Visualization, 4 (2), 2005, 114-135.
14. Heer, J. and Boyd, D., Vizster: Visualizing Online Social Networks. in InfoVis, (Minneapolis, MN, 2005).
15. Herman, I., Melançon, G. and Marshall, M.S. Graph Visualization and Navigation in Information Visualization: A Survey IEEE Transactions on Visualization and Computer Graphics 6 (1). 2000, 24-43
16. Moreno, J. Who shall survive ? Foundations of Sociometry, Group Psychotherapy and Sociodrama. in, Nervous and Menal Disease Publishing Co., 1934.
17. Nardi, B., Whittaker, S., Isaacs, E., Creech, M., Johnson, J. and Hainsworth, J. ContactMap: Integrating Communication and Information through Visualizing Personal Social Networks. Communications of the ACM, 45(4) 2002,89-95
18. Otjacques, B., Feltz, F., Halin, G. and Bignon, J.-C., Mat'Graph : Transformation matricielle de graphe pour visualiser des échanges électroniques. in Actes de la 17e conférence francophone sur l'Interaction Homme-Machine (IHM'05), (Toulouse, France, 2005), ACM Press.
19. Rao, R. and Card, S.K. The table lens: merging graphical and symbolic representations in an interactive focus + context visualization for tabular information in Proceedings of the SIGCHI conference on Human factors in computing systems: celebrating interdependence ACM Press, Boston, Massachusetts, United States 1994, 318-322
20. Scott, J.P. Social Network Analysis: A Handbook. SAGE Publications, 2000.
21. Siirtola, H. and Mäkinen, E. Constructing and Reconstructing the Reorderable Matrix. Information Visualization, 4, 2005. 32-48.
22. Spenke, M., Beilken, C. and Berlage, T. FOCUS: the interactive table for product comparison and selection in Proceedings of the 9th annual ACM symposium on User interface software and technology ACM Press, Seattle, Washington, United States 1996 41-50
23. Wasserman, S. and Faust, K. Social Network Analysis. Cambridge University Press, 1994.