VISUGRAPH : UN OUTIL POUR L'ANALYSE DU RELATIONNEL

Eloïse LOUBIER (*), **Bernard Dousset** (*)
loubier@irit.fr; dousset@irit.fr

(*) IRIT, UPS, 118 route de Narbonne
31062 Toulouse Cedex 9

Mots clefs :
Modélisation des connaissances, algorithme de placements dirigés par des forces, exploration de graphe, interactivité, outil de veille stratégique, visualisation de données relationnelles temporelles, structure.
Keywords:
Knowledge modeling, force directed placement algorithm, graph exploration, interactivity, strategic watch tool, relational and temporal data visualization, structure.
Palabras clave :
Formalización del conocimiento, algoritmo de colocación dirigida, exploración gráfica, herramienta interactiva para la inteligencia empresarial, visualización de datos relacional estructura temporal

Résumé

Dans cet article, l'outil de visualisation de données relationnelles est présenté, ainsi que toutes ses fonctionnalités interactives. Basé sur des matrices de cooccurrences 2D, dans le cas statique, ou 3D dans le cas évolutif, les graphes réalisés facilitent l'exploration de données et la prédiction dans un contexte de veille stratégique. Ainsi la visualisation interactive de données relationnelles apporte à l'utilisateur un substrat artificiel qui transcrit un grand nombre d'informations, faisant ainsi fonction de support à ses connaissances et à son intuition pour lui permettre de découvrir de nouvelles relations, l'aider à la prise de décision et permettre l'anticipation, quant à l'évolution de ces données.

1 Introduction

Dans un contexte d'extraction de connaissance, l'objectif de la représentation graphique est de faciliter la recherche des structures, caractéristiques, motifs, tendances, anomalies et des relations entre les individus [14]. La visualisation apporte une valeur ajoutée. Les travaux de [10] caractérisent cet apport comme étant l'augmentation des capacités de perception de l'humain, en fournissant une vision perspicace des données.
La base de toute proposition d'outil de visualisation de données relationnelles suppose de s'intéresser aux trois aspects fondamentaux suivants :

- la nature des données représentées,
- la manière dont les composantes du graphe sont exploitées pour transcrire ces données,
- la perception de ces composantes par l'utilisateur.

Tout l'art de la conception de graphe consiste alors à passer de l'espace des informations à une représentation visuelle qui traduise, grâce aux composantes utilisées, l'information originelle. C'est cette étape de traduction ou de transcription de l'information vers un espace de représentation visuel qui nous intéresse. Le rôle de l'utilisateur dans les outils de visualisation de données est un sujet de préoccupation majeure [14], [10], [23]. Ainsi la visualisation interactive de données relationnelles apporte à l'utilisateur un substrat artificiel qui transcrit un grand nombre d'informations, faisant ainsi fonction de support à ses connaissances et à son intuition pour lui permettre de découvrir de nouvelles relations, l'aider à la prise de décision et permettre l'anticipation, quant à l'évolution de ces données. En fouille visuelle de données, l'interaction matérialise la boucle de rétroaction entre l'utilisateur et le support visuel [22]. L'objet de la visualisation n'est pas simplement limité à la production de représentations graphiques prédéfinies, non modifiables par l'utilisateur. En effet, un critère important d'un bon outil de visualisation de données est la possibilité pour celui qui le manipule, de le contrôler et le maîtriser pleinement afin de comprendre l'espace des informations ou d'interagir avec lui. La visualisation rejoint sur ce point les préoccupations qui sont du domaine de l'*interaction* homme-machine. Dans cet article, nous présentons VisuGraph, un outil de visualisation de données relationnelles, composé de fonctionnalités permettant l'analyse de structure graphiques d'entités, pouvant prendre en compte la dimension temporelle. Dans un premier temps, nous présentons les représentations possibles sous VisuGraph. Puis dans un second temps, nous exposons les fonctionnalités permettant d'analyser les graphes, qu'ils soient temporels ou non.

2 Visualisation de données

2.1 Approche

Les techniques de visualisation, et en particulier l'outil que nous proposons, viennent en aval des étapes de traitement automatiques. Suite à l'application d'algorithmes de découverte des structures, elles permettent de représenter les résultats sous des formes intelligibles facilitant leur interprétation. VisuGraph, l'outil que nous proposons, se base sur des matrices de cooccurrences 2D ou 3D, selon la prise en compte, ou non, du temps dans les croisements des entités.
Dans ce contexte de proposition d'outil d'aide à l'analyse de données relationnelles, deux axes majeurs sont pris en compte :

- une représentation des données, dans un espace défini, en utilisant la notion de métrique pour caractériser les composants du graphe, représentés et placés spécifiquement ;
- une possibilité pour l'utilisateur de naviguer librement, à travers des méthodes d'exploration de graphe.

Dans tout ce qui suit, nous utilisons les notations suivantes.
Un graphe simple G est un couple formé de deux ensembles :

- $X = \{x_1, x_2, ..., x_n\}$ dont les éléments sont appelé sommets ou encore nœuds, n étant fini ;
- $A = \{a_1, a_2, ..., a_m\}$, partie de l'ensemble $P_2(X)$ des parties à deux éléments de X, dont les composants sont appelés arêtes. Lorsque $a = \{x,y\} \in A$, on dit que a est l'arête de G d'extrémités x et y, ou que a joint x et y, ou encore a passe par x et y. Les sommets x et y sont dit adjacents dans G. Dans la cas des graphes orientés, si $a = (x,y)$ est un arc du graphe G, x est l'extrémité initiale de a, ou encore appelée *extrémité initiale* et y est l'*extrémité terminale* de a, ou bien *origine* et *destination.* L'arc a part de x et arrive à y.

2.2 Métriques

Les graphes servent à modéliser des structures relationnelles comportant un ensemble d'entités et des relations liant ces entités entre elles.
Afin de faciliter la compréhension des graphes, le concept de métrique permet la comparaison des différents éléments d'un graphe par affectation de valeur des différentes entités (sommets, arêtes) composant ce dernier.
Les travaux de [34] mènent au concept de *nœud métrique* comme une quantité numérique associée aux nœuds et aux arêtes du graphe.
Nous ciblons, ici, la métrique basée sur le contenu, c'est-à-dire sur les valeurs des données. Dans VisuGraph, les métriques sont associées aux sommets mais aussi aux arêtes [32].
Les mêmes fonctions sont utilisées pour la coloration des arêtes afin d'identifier les liens forts et faibles dans le graphe. Elle est utilisée à la fois pour définir l'épaisseur et l'intensité de couleur des arêtes [18]. Ces variations sont ordonnées mais elles ne sont pas quantitatives puisque les différents niveaux de couleurs utilisés, que ce soit pour la coloration des arêtes ou encore des sommets, peuvent être classés mais il est impossible de chiffrer la différence entre deux valeurs.

2.3 Placement des sommets

La position individuelle des sommets, sans prendre en compte l'aspect structural constitué avec leur voisinage, n'est pas significative. Dans un contexte non évolutif, elle ne traduit pas la valeur d'attribut des données, mais celle relative aux liens entre les sommets. Ainsi, le positionnement des sommets est exclusivement calculé de manière à satisfaire un certain nombre de critères esthétiques ou pratique de construction de graphe, tels que la minimisation des croisements d'arêtes, l'optimisation de la surface de représentation,… La visualisation graphique des données nécessite l'attribution de coordonnées x et y pour chaque nœud visualisé dans l'espace de représentation.
Dans le cas d'un graphe statique représenté par VisuGraph, les sommets sont initialement placés de manière circulaire. Dans le cas de graphe biparti, les sommets sont placés sur deux cercles concentriques en fonction de leur type. Les sommets qui correspondent aux lignes de la matrice sont situés sur le cercle extérieur et ceux associés aux colonnes sont sur le cercle intérieur. Cette représentation par défaut permet de visualiser sans grand effort cognitif, la répartition des deux ensembles de sommets, dans le cas biparti.
Dans le cas évolutif, le positionnement des données s'effectue de façon circulaire, selon le principe de l'horloge. Dans un premier temps, chaque période considérée est assimilée à un sommet nommé « *repère temporel* ». Chacun de ces repères est placé de façon circulaire près des bords de la fenêtre, tous comme le sont les heures sur un cadran d'horloge [33].
Une fois ces repères positionnés, chaque donnée est placée à leur proximité de façon proportionnelle à leur appartenance à chaque période. Ainsi, si une donnée à une valeur de métrique nulle pour une période, son positionnement n'est pas proche du repère de cet instance. Inversement, plus la valeur de sa métrique est importante pour une ou plusieurs périodes, plus la donnée est proche des repères correspondant.

2.4 Algorithmes de représentation de graphe

Afin d'améliorer la représentation de graphe et d'obtenir une visualisation la plus planaire possible, c'est-à-dire minimisant le nombre d'entrecoupements d'arêtes, nous nous basons sur l'analogie « *arc = ressort* ». Notre modèle s'inspire des travaux de [12]. Le système, ainsi considéré, engendre des forces entre les sommets, ce qui provoque naturellement des déplacements de ces derniers. La notion d'attraction entre les sommets s'effectue par leur rapprochement pour ceux fortement liés et la répulsion s'établit par éloignement des nœuds. La condition d'arrêt initialement proposée pour un tel système est un nombre maximum d'itérations selon l'évolution du graphe dans le temps. L'utilisateur laisse les forces agir jusqu'à ce qu'il obtienne satisfaction des résultats visuels. Dans notre proposition nous prenons en compte plusieurs paramètres, à savoir :

- Le dosage, par l'utilisateur, de l'attraction et de la répulsion,
- La distance minimale entre les deux sommets,
- L'aire de représentation du graphe (fenêtre de représentation).

Dans un premier temps, nous proposons un algorithme général [18], [30] permettant un meilleur rendu pour la représentation graphique, quelque soit le type de données (temporelles ou non).

La force d'attraction entre deux sommets *u* et *v* est donnée par :

$$f_a(u,v) = \frac{\beta \times d_{uv}^{\alpha_a}}{K} \qquad [1]$$

β est une constante. d_{uv} est la distance entre *u* et *v* dans le dessin. α_a sert à augmenter/diminuer l'attraction entre deux sommets.

Le facteur *K* est calculé en fonction de l'aire du dessin et du nombre de sommets du graphe et permet de s'assurer du non dépassement par les sommets, des bords de la fenêtre de représentation. Pour cela, *L* représente la longueur de la fenêtre, *l* la largeur et *N* correspond au nombre de sommets visibles du graphe.

$$K = \sqrt{\frac{L \times l}{N}} \qquad [2]$$

Si les sommets *u* et *v* ne sont pas reliés par une arête alors $f_a(u, v) = 0$.

La force de répulsion entre deux sommets *u* et *v* est définie par :

$$f_r(u,v) = \frac{\alpha_r \times K^2}{d_{uv}^c} \qquad [3]$$

α_r sert à augmenter/diminuer la répulsion entre deux sommets *u* et *v* ; *c* est, dans ce cas là, une constante.
Afin d'obtenir davantage d'interactivité entre le système et l'utilisateur et surtout pour permettre à ce dernier de contrôler pleinement sa représentation graphique, l'attraction ou/et la répulsion entre les sommets peuvent manuellement être modifiées.
Pour cela, des sliders[1] sont mis à disposition, dans le menu, permettant l'augmentation ou la diminution de ces deux types de forces. Le système dispose ainsi d'un slider spécifique aux forces d'attractions et un pour les forces de répulsion. Chacun des sliders est composé de dix graduations et la valeur d'initialisation est par défaut à 5.

Nos expérimentations nous mènent à préconiser un ordonnancement spécifique pour obtenir un meilleur résultat visuel, en ce qui concerne le paramétrage de ces trois forces.

Etape 1) Appliquer une très forte valeur d'attraction, via le slider spécifique, jusqu'à obtenir un regroupement concentré des données, permettant de distinguer la structure globale du graphe. Mettre la température au maximum via le slider, afin de permettre le déplacement rapide et efficace des sommets.

Etape 2) Réduire cette force d'attraction et augmenter la répulsion, via les deux sliders, afin d'obtenir un graphe lisible. Réduire la température pour éviter un mouvement trop brutal des sommets.

Etape 3) Ajuster sensiblement les trois sliders, en baissant la température, jusqu'à obtention d'un résultat satisfaisant.

Ces principes sont illustrés dans la

Figure 1, décomposant les différentes étapes du dessin de graphe.

[1] Règle graduée, dont le seuil initialement fixé peut être changé.

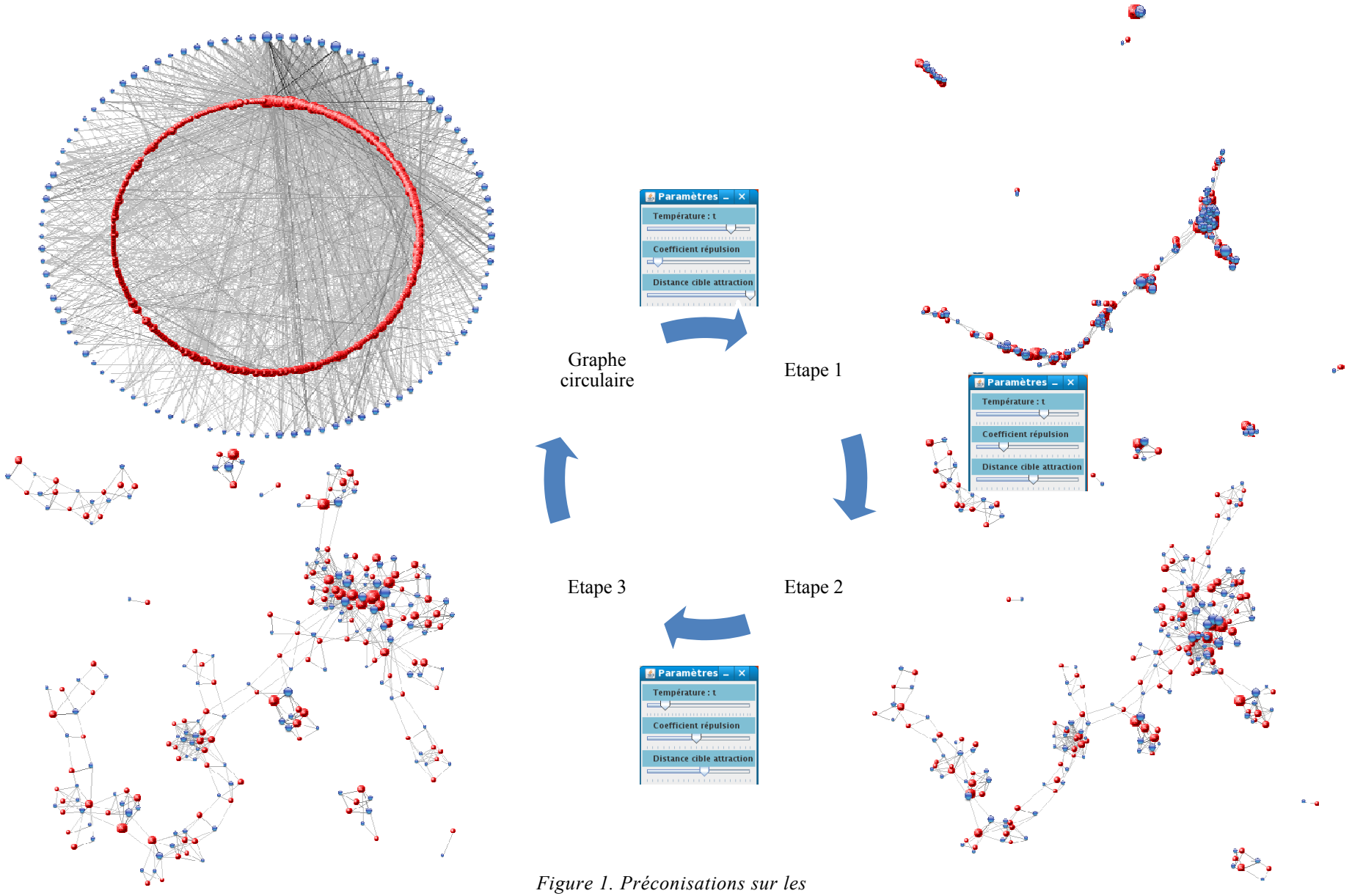

Figure 1. Préconisations sur les différentes étapes de réglage des FDP.

2.5 Graphes orientés

Par convention, nous appelons *prédécesseur* le sommet x_i, à partir duquel l'arc est tracé, et *successeur* le sommet d'arrivée x_j. Dans l'exemple de la Figure 2, x_1 est le *prédécesseur* et x_2, le *successeur*.

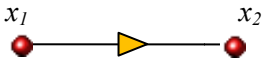

Figure 2. Prédécesseur et successeur.

A l'origine d'un graphe orienté, sous VisuGraph, se trouve une matrice croisant les prédécesseurs et les successeurs. Le graphe obtenu permet alors de distinguer clairement qui est à l'origine de qui. Cependant, il est possible d'ajouter des informations à un graphe orienté afin d'enrichir son potentiel informatif. Dans la Figure 3, Pn représentent les prédécesseurs et Sn les successeurs.

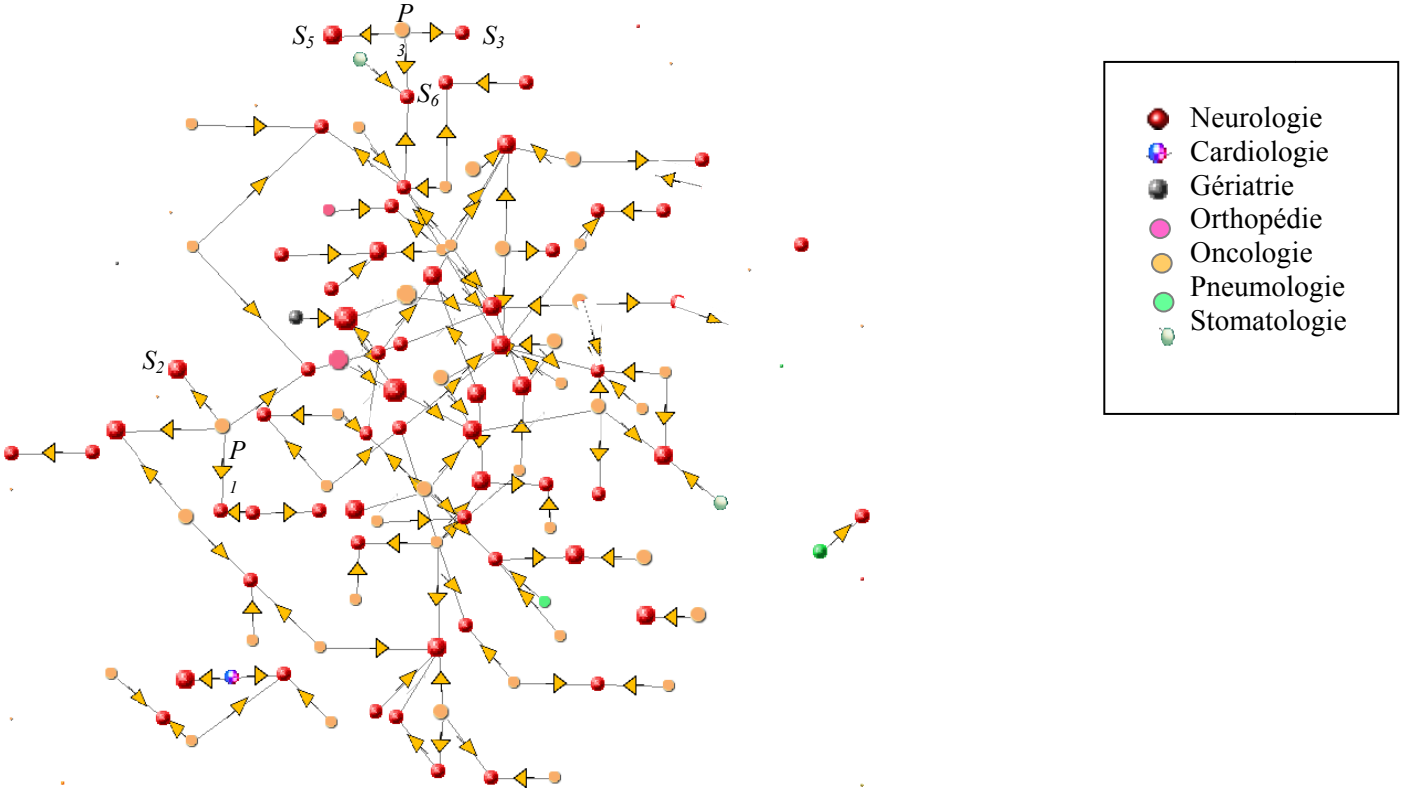

Figure 3. Graphe *orienté biparti.*

Par exemple, intéressons nous au cas d'un service de veille d'une entreprise fournissant des licences et cherchant à étudier les forces actuelles du marché dans son domaine. Un graphe orienté croisant les entreprises vendant les licences et celles les achetant dans le domaine pharmaceutique est très utile pour le veilleur. Cependant, ce dernier, peut vouloir disposer sans grand effort cognitif du domaine médical spécifique des entreprises afin de mieux cibler le domaine étudié. Pour cela, VisuGraph permet de compléter le graphe orienté par analyse d'une matrice asymétrique croisant toutes les entreprises, qu'elles soient prédécesseurs ou successeurs avec les domaines médicaux.

3 Les fonctionnalités

3.1 Identification et analyse de structure de graphe

La visualisation et l'étude de graphe consistent à analyser le réseau formé par ces entités et la combinaison de leurs relations, afin de comprendre la façon dont la structure contraint les comportements individuels tout en faisant émerger des interactions. Un graphe se caractérise d'abord, très simplement, par son ordre, c'est-à-dire par le nombre de ses sommets indiqué dans la console. Le concept dominant de l'analyse structurale est moins celui de lien ou de relation que celui de système, c'est-à-dire qu'il s'agit de rechercher les formes structurelles du système [9]. L'analyse structurale tente de trouver les régularités de comportement. Plus un individu est proche des autres, plus il est susceptible d'avoir d'informations [26], d'accéder à un plus haut statut social [21], d'avoir du pouvoir [6], de l'influence [4], [11], du prestige [5]. Pour étudier cette proximité, plusieurs critères sont alors utilisables.

- La *connexité* consiste à repérer des groupes dont les membres sont liés de façon directe ou indirecte.
- La *cohésion* s'appuie plutôt sur la densité des relations dans le groupe.
- L'*équivalence* introduit un autre point de vue en permettant de rassembler les individus en fonction de leur similitude.

On peut aussi vouloir caractériser chaque acteur d'après sa position dans le graphe, par exemple selon sa centralité. Les études qui utilisent ces notions relèvent de la théorie des graphes [38]. Si l'on dispose seulement de données décrivant les réseaux personnels d'un échantillon d'individus, généralement choisis pour être représentatifs d'une population plus large, il n'est pas impossible de tester l'influence de certaines caractéristiques structurales sur le problème traité.
Le travail de description consiste à inventorier la diversité des régimes d'action et des entités mises en relation dans le réseau. Ainsi, les individus les plus centraux dans un graphe occupent des relations privilégiées dans les échanges, notamment par rapport à ceux qui sont situés plus à la périphérie.
L'analyse de la structure d'un graphe mène à étudier la centralité des sommets les plus importants. Dans VisuGraph, ces notions peuvent être étudiées, par l'affichage d'un sommet et de son entourage direct, mais aussi par l'étude spécifique d'une partie de la structure du graphe. La proximité entre les sommets permet de révéler l'intensité de leur relation. Ces notions sont par la suite approfondies à travers l'usage des fonctionnalités de VisuGraph permettant l'exploration du graphe. L'analyse de la structure d'un graphe sert à la construction d'indicateurs prometteurs pour caractériser la dynamique d'un ensemble de données relationnelles. Elle sert à caractériser individuellement les données représentées qui assurent ou au contraire réduisent la cohésion globale.

3.2 Morphing de graphe : du graphe global au graphe de période

3.2.1 Définition

La représentation graphique facilite l'exploration des données et des différentes tendances par analyse de la structure du graphe et particulièrement par voisinage des sommets. Cependant, dans le cas temporel, il est important de considérer dans un premier temps la visualisation globale des données, puis, dans un second temps, la représentation individuelle de chaque période, avec la possibilité de revenir à tout moment à n'importe quel type de graphe, général ou non.

Nous définissons le ***morphing de graphe*** comme la transformation géométrique T_g d'une représentation graphique, permettant le passage d'une visualisation de donnée au temps *t-1* à celle de *t* et inversement. Il s'agit d'une déformation de graphe continue. Le morphing [30] consiste à fabriquer une animation qui transforme de la façon la plus naturelle et la plus fluide possible un graphe initial vers un graphe final.

L'objectif est de réaliser une lecture intuitive de l'évolution en répartissant séquentiellement les périodes de façon cyclique, permettant, à partir de la représentation du graphe global, de visualiser successivement chaque graphe de période, de façon animée et fluide. En se basant sur l'analogie espace/temps, il permet ainsi de détecter, comprendre et même prévoir les tendances significatives, au travers de la visualisation de l'évolution des données.

- Soit $G_g = (X,A)$ le graphe global ;
- Soit P_t un sous ensemble de X, noté $P_t = \{x_1, \ldots, xt\}$ *et* caractérisé par l'appartenance à une période spécifique.

On dit que P_t est un ***graphe de période*** issu de G_g.

Un sous-ensemble de X, élément de P_t peut être vide, dans le cas d'une période étudiée durant laquelle aucune des données n'a de valeur de métriques positive. Ce cas là ne présente alors que peu d'intérêt.

Les sous-ensembles de X, éléments de P_t ne sont pas obligatoirement disjoints deux à deux, dans le cas où des données sont valuées durant plusieurs périodes.

$$\forall (i,j) \in \{1, \ldots, y\}, si\ \ i = j, V_i \cap V_j \neq \emptyset$$
$$ou\ si\ i \neq j, V_i \cap V_j = \emptyset$$

3.2.2 Principe

Le morphing de graphe se base sur une représentation globale des données temporelles, en utilisant une structure permettant de détecter rapidement les caractéristiques temporelles des données, à savoir quelles sont les données persistantes ? Celles apparaissant ?

L'animation des visualisations successives des différentes périodes, dans le sens chronologique similaire à l'analogie espace/temps d'une horloge, permet de créer une certaine dynamique, révélant l'évolution des données au cours du temps, trouvant ainsi un bon compromis entre la préservation de la carte mentale de l'utilisateur et la lisibilité du tracé. Le morphing de graphe repose sur deux types de visualisation.

La représentation globale sert de carte mentale à l'utilisateur, elle est à l'origine de toute visualisation temporelle dans VisuGraph. Les données sont placées spécifiquement selon leurs spécificités temporelles.

Afin de maintenir une bonne interactivité avec l'utilisateur, il faut préserver au mieux la stabilité des tracés. « *La stabilité est une notion complexe qui dépend des caractéristiques géométriques et combinatoires du tracé, mais aussi des facultés de perception et de mémorisation de l'utilisateur* » [36].

Pour cela, l'utilisateur doit garder en mémoire le graphe global et pouvoir s'y référer, c'est à dire sa carte mentale, en limitant les perturbations apportées sur le nouveau tracé de graphe par période, par rapport aux précédents. A travers le graphe global, l'utilisateur visualise toutes les données, toutes périodes confondues,
Il est donc indispensable que ce dernier soit intelligible et clair, permettant, sans grand effort cognitif, de situer chaque graphe période dans le graphe global. Il doit reposer sur un placement fixe par défaut des sommets, selon des repères spatiaux précis, permettant une mémorisation simple de la représentation.
Afin de pouvoir étudier chaque période individuellement, nous proposons, suite à la visualisation du graphe global, de réduire ce dernier à des graphes temporels appelés « graphes de périodes ». Pour une instance spécifique, tous les sommets et toutes les arêtes n'appartenant pas à la période d'étude sont masquées. Il ne reste alors que les données propre à la période analysée. Le choix de l'instance à visualiser s'effectue par le biais d'un slider, actionné par l'utilisateur.
Le passage d'un graphe à l'autre s'effectue par disparition progressive des sommets appartenant au graphe d'origine mais pas au final, par apparition progressive des éléments naissant et par évolution des persistants.

3.3 Filtrage

Une première méthode pour analyser plus simplement un graphe est le filtrage. Un nettoyage préalable du graphe, basé sur une technique de filtrage appropriée [16] permet de révéler une structure et des motifs intéressants. La difficulté est de proposer un filtre qui révèle des caractéristiques sans pour autant dénaturer le graphe. Filtrer un graphe revient à filtrer ses sommets ou ses arêtes selon certains critères. Ces derniers sont basés sur les propriétés quantitatives ou qualitatives des sommets ou arêtes [15], [16]. Dans VisuGraph, le filtrage dynamique, basé sur les valeurs de la métrique utilisée, consiste à ne conserver que les sommets et les arêtes du graphe associés aux valeurs supérieures ou égales à un seuil. La dynamique du filtrage est une idée clé de la visualisation de l'information. Grâce au filtrage, l'utilisateur peut contrôler le volume des contenus à afficher pour se concentrer sur ce qui l'intéresse. Cette procédure fait apparaître les sommets les plus représentatifs, ainsi que les composantes importantes de la structure. Dans notre cas, le filtrage s'effectue par masquage des arêtes ayant une valeur de métrique inférieure au seuil fixé par l'utilisateur. Cela implique le masquage des sommets isolés. Le filtrage permet d'éliminer les objets inintéressants en favorisant la visualisation des liaisons les plus importantes. Ainsi, le filtrage d'un graphe complet permet d'obtenir un sous-graphe, comportant des sommets isolés. Le masquage de ces derniers permet d'obtenir un graphe partiel. Le filtrage permet aisément de détecter la nature des liens entre les acteurs du graphe. Ainsi, un graphe composé majoritairement de liens unitaires peut donner des indications riches sur les échanges entre les données.

3.4 *K*-Core

Une autre fonctionnalité permettant l'étude de la structure d'un graphe est le *k*-core. Cette décomposition [3] consiste à identifier des sous ensembles particuliers du graphe appelés *k-core*.
Un *k-core* est défini comme suit :

- Un sous-graphe $S = G(C, A|C)$ induit par l'ensemble $C \subseteq X$ est un *k-core* ou un *core* d'ordre *k* si et seulement si $\forall v \in C : degre_H(v) \geq k$, et *S* est un sous ensemble maximal avec cette propriété.
- Notons que le *k-core* est unique [2].
- Un nœud a un *coreness c*, s'il appartient au *core* d'ordre *c* et s'il n'appartient pas au *core* d'ordre C*c + 1)*.
- Un ensemble connexe de *coreness* c_E forme un cluster, ou encore une communauté, au sens de [2].

Le *k-core* est obtenu par élagage récursif des nœuds qui ont un degré plus petit que *k*. Le graphe restant ne contient que des sommets de degré ≥ *k*.
Appliqué à VisuGraph, le *k-core* est calculé à partir d'un seuil fixé par l'utilisateur. Plus ce seuil augmente plus le *coreness* est élevé. Le *k*-core permet de cibler le cœur du graphe, au détriment de sa périphérie.
Ainsi, l'analyse du graphe, via des techniques précises telles que les *k-cores* permettent de comprendre l'objet dans son ensemble.

3.5 Transitivité

La transitivité s'exprime très simplement et assez justement par « *les amis de mes amis sont mes amis* ».Dans VisuGraph, cet algorithme s'applique à partir d'un sommet sélectionné par l'utilisateur et par la fixation d'un seuil à l'aide d'un curseur, la fermeture transitive correspond à un rang alors égale au seuil choisi. L'analyse de graphe est le moyen d'élucider des structures et de s'interroger sur leurs rôles [35]. Au-delà de la méthodologie [25], il s'agit de comprendre en quel sens une structure contraint concrètement des comportements, tout en résultant des interactions [7] entre les éléments qui la constituent. L'étude d'un sommet particulier et de ses relations avec d'autres entités par le biais de ses connexions permet de définir son rôle au sein de la structure. La donnée visualisée peut alors apparaître comme un acteur majeur du domaine ou encore comme un chaînon d'une grande équipe. La forme du réseau a une incidence sur les ressources qu'un individu peut mobiliser et sur les contraintes auxquelles il est soumis. Elle ne le détermine pas, mais elle explique que tout ne soit pas possible pour lui et que dès lors certains comportements ou stratégies sont, en raison de la position occupée dans le graphe, plus probables que d'autres. Si des sommets A et C sont liés au sommet B, c'est peut-être qu'ils détiennent des caractéristiques communes, mais aussi des comportements proches ou « compatibles ».
Le voisinage d'un acteur, ou réseau égo-centré, est l'instrument majeur permettant d'observer les formes de rapprochement que l'individu opère entre des relations, des ressources et des références différentes. En ce centrant sur l'analyse des réseaux égo-centrés, le chercheur peut restituer la diversité des relations et préserver le caractère local de l'espace dans lequel elles se développent.
Ainsi, pour expliquer la configuration d'une structure comme une organisation, il faut aussi tenir compte des caractéristiques de l'individu en dehors de cette structure. « Nous ne pouvons pas comprendre les interactions d'un groupe donné d'individus si nous ne les considérons pas à la lumière de l'ensemble des liens que chaque acteur entretient en dehors de l'espace commun » [13].
Nous les caractérisons en plusieurs catégories :

- Les sommets individuels, liés à aucun autre mais caractérisés par une forte valeur de métrique, symbolisant son importance dans le domaine étudié.
- Les sommets isolés de faibles valeurs, n'appartenant à aucune structure. Nous les qualifions « d'électrons libres ».
- Les sommets appartenant à une structure et se trouvant en bout de structure. Ces éléments sont caractérisés comme membre d'une équipe mais pas comme leader. Ces sommets sont caractérisés comme communiquant peu avec le reste du graphe, avec un faible nombre de liens, même si ces derniers peuvent être de forte valeur. L'étude de leurs transitivités est intéressante puisqu'elle permet de reconstituer l'ensemble de l'équipe et de connaitre le nombre d'intermédiaire entre deux extrémités de la structure. Dans cet exemple, la structure est reconstruite par transitivité en sept pas. Le premier seuil révèle une liaison qu'avec un seul autre sommet. Puis, par voisinages successifs l'architecture du sous-graphe s'amplifie.
- Les sommets au cœur de la structure sont caractérisés par de nombreux liens avec les autres membres. Leur suppression entraine la rupture en deux de cette dernière. La fermeture transitive permet d'une part de reconstituer l'équipe, tout en étudiant le nombre prédominant de liens avec les autres sommets. [5] nomme ces rotules des « trous sociaux », c'est-à-dire la théorie selon laquelle deux acteurs ne peuvent communiquer entre eux que par l'intermédiaire d'un troisième acteur, qui occupe ainsi une position avantageuse.

Le graphe de la Figure 4 illustre le principe de ce sommet qui, dès le premier seuil, est lié à de nombreux autres. Par les différents seuils de transitivité, il est constable que toute la structure repose sur ce sommet, puisque c'est autours de lui que viennent se lier les autres éléments.

Figure 4. Calcul de la transitivé d'une rotule.

3.6 Retour aux documents

VisuGraph est doté d'une fonctionnalité permettant de favoriser l'exploration locale d'un sommet. Suite à la sélection d'un nœud spécifique, un pop up[2] apparait indiquant l'intitulé du sommet, la valeur de sa métrique, ainsi que le libellé des nœuds auxquels il est lié, ainsi que la valeur des liens. Cet affichage est complété par le retour aux notices. Cette fonctionnalité permet, à partir d'un sommet sélectionné, d'afficher dans un éditeur de texte, les documents contenant l'item choisi. Ainsi, si l'utilisateur clique sur un sommet, tous les documents du corpus initial concernant ce nœud s'afficheront dans un éditeur de texte [27], [29]. Ainsi, pour un sommet sélectionné, tous les synonymes ou variation orthographique de ce dernier seront retournées. Un autre aspect important de la synonymie, dans un contexte de « retour aux notices », est la proximité des termes. En effet, un même terme peut être écrit sous des formes différentes, telles que les erreurs de saisie, abréviations. Par exemple, le nom de notre institut peut être écrit sous la forme « IRIT » ou « Inst. de Rech. en Inf. de Toulouse » ou « Institut de Recherche en Informatique de Toulouse » ou encore « UMR 5505 ». Or, dans le cas d'un croisement entre auteurs et documents, lors de la recherche des documents dans lesquels un auteur particulier apparaît, le système doit retourner ceux comportant au moins une parmi toutes les formes possibles d'écriture de son nom.

[2] Nouvelle fenêtre s'ouvrant automatiquement au dessus de la fenêtre de navigation actuelle.

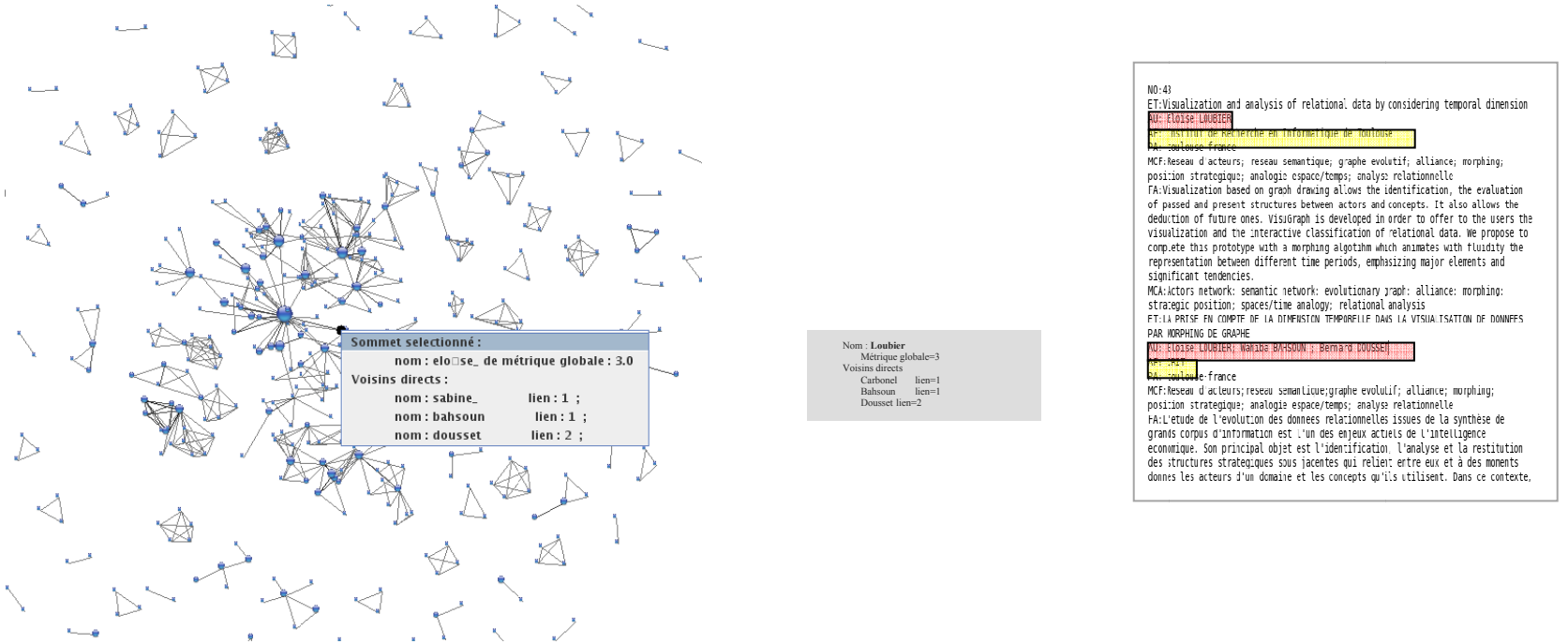

Figure 5. Exploration d'un sommet spécifique et retour aux notices.

3.7 Partitionnement de graphes

Dans un contexte d'analyse de graphe, la lisibilité et l'interprétation deviennent de plus en plus complexe, lorsque le volume de données augmente. Il est indispensable d'avoir à disposition des techniques permettant de réduire le nombre de données représentées, lorsque ce dernier devient trop important. Le partitionnement de graphe, intégré à VisuGraph, est une solution à ce problème.

Créer une partition consiste à répartir un ensemble d'objets en plusieurs sous-ensembles. Il est utile de pouvoir comparer ces sous-ensembles entre eux. Ainsi, chaque objet va être associé à d'autres objets, et les associations résultantes, ou liens, doivent pouvoir être quantifiées. L'objectif du partitionnement d'un ensemble d'objets est en général de répartir ceux-ci en parties ayant de très forts liens internes et de faibles liens entre les groupes. Le but des méthodes de classification est de construire une partition d'un ensemble d'objets. La classification a pour hyponyme le partitionnement de données, qui se traduit en anglais par data clustering.

Une *k-partition* d'un ensemble *X* est définie par une famille de sous ensembles $\{x_1, ..., x_k\}$ vérifiant :

$$\bigcup_{i=1}^{k} V_i = V \; et \; V_i \cap V_j = \emptyset, \forall i \neq j \qquad [4]$$

Un graphe clusterisé est un graphe *G = (X, A)* pour lequel on dispose d'une partition *{x_1,…,x_k}* de l'ensemble des sommets où les x_i sont des clusters.
Afin de faciliter l'analyse, les données les plus fortement liées doivent être regroupées en classes homogènes. Parmi les travaux effectués sur le partitionnement de graphe, les travaux de [1], [24], [17] se basent sur des approches spectrales alors que les algorithmes de la famille METIS [20] se basent sur le partitionnement multi niveaux. Le Markov Clustering (MCL) consiste en l'alternance d'un mouvement en deux étapes -*expansion* et *inflation*- afin d'atteindre la convergence d'une matrice stochastique par laquelle un réseau entier est subdivisé en « clusters durs » sans aucun chevauchement. Ordinairement, le sous-réseau de chaque cluster de Markov est de type « étoile » dont le centre est le nœud de plus haut degré et les autres nœuds ne sont reliés qu'à celui-ci.
Ce type d'approche est basé sur la notion des déplacements aléatoires dans un graphe, selon un processus stochastique à temps discret par lequel on se déplace d'un sommet à un autre, choisi aléatoirement.
La méthode de partitionnement utilisée dans VisuGraph est inspirée du Markov Clustering [37] que nous avons aménagée pour pouvoir influencer le nombre de classes proposées [19].
Cette approche, dont l'algorithme se base sur deux opérations matricielles simples, successivement itérées :

- La première calcule les probabilités de transition par des marches aléatoires de longueur fixée *r* et correspond à une élévation de la matrice à la puissance *r*, visant à élargir la capacité de l'arc entre deux nœuds.
- La seconde consiste à amplifier les différences en augmentant les transitions les plus probables et en diminuant les transitions les moins probables. Les transitions entre sommets d'une même communauté sont alors favorisées et les itérations successives des deux opérations conduisent à une situation limite dans laquelle seules les transitions entre sommets d'une même communauté sont possibles.

La complexité totale de l'algorithme est en *O(n3)*. L'évaluation de la méthode MCL a montré la rapidité et la qualité de ses résultats [8]. Le graphe final est alors un graphe de classe, pour lequel chaque sommet est en fait une des classes obtenues, permettant de travailler alors sur un graphe réduit. Les liens entre les sommets sont assimilés à des liaisons interclasses [28], [31].
Dans un second temps, l'attribution d'une couleur spécifique à chaque classe permet de visualiser le graphe de départ, en figeant un représentant par classe et en distribuant les autres sommets sur une couronne centrée sur ce dernier, permettant ainsi une première vue intra classe.
L'avantage d'un tel procédé est de pouvoir travailler alternativement sur un graphe dit réduit, facilement manipulable et beaucoup plus lisible et sur le graphe initial avec un dessin initialisé par celui du graphe réduit. On peut ainsi passer d'une vue synthétique à des vue détaillées des classes redessinées autour de leurs centres et qui ne se recouvrent plus. Sur la fenêtre de visualisation, chaque classe obtenue par application de l'algorithme MCL apparait sous forme d'un sommet de couleur. Le contenu de chaque classe peut être obtenu en détail en cliquant sur ce sommet. Une nouvelle fenêtre apparait, dans laquelle chacun des sommets composant la classe est focalisé.
De plus, VisuGraph offre la possibilité d'obtenir sous forme de liste textuelle tous les composants d'une classe. Cette fonctionnalité, applicable à partir du menu, permet d'obtenir le fichier texte résultat.
Pour obtenir le détail de chaque classe, deux solutions sont appliquées, selon le point de vue global ou local.

Dans le *cas global*, le retour à un graphe complet permet d'obtenir autour de chaque représentant, l'ensemble des sommets constituant la classe, comme visualisé dans le quatrième graphe.
Dans le *cas local*, il est possible d'extraire la classe, en l'affichant dans une autre fenêtre de façon complète, c'est-à-dire en visualisant le représentant et les constituants de la classe, ainsi que les liens directs de la classe.

4 Conclusion

Dans cet article, nous avons présenté VisuGraph, un outil de représentation de graphes statiques mais aussi de graphes dynamiques. Cet outil se base sur une représentation de sommets et de liens symbolisant leurs relations, par le biais de techniques de sémiologie préconisées dans le chapitre précédent, tels que la couleur, la forme, la taille… L'outil développé permet de représenter des graphes basés sur les matrices de cooccurrences. Ces derniers peuvent être des graphes simples, bipartis, orientés ou/et temporels. L'originalité de notre proposition repose sur toutes ces solutions possibles au sein d'un même outil et sur le panel de méthodes spécifiques permettant l'exploration et l'analyse des graphes.
Les fonctionnalités d'exploration intégrées à VisuGraph permettent l'analyse de la structure, par étude locale ou globale du voisinage des sommets caractéristiques.
L'interactivité entre la représentation et l'utilisateur étant un critère majeur dans nos travaux, toutes les fonctionnalités présentées sont paramétrables via des sliders, afin que le manipulateur reste maître de sa visualisation. Ce dernier choisit les méthodes à appliquer sur le graphe, il règle via la fenêtre de paramètre le dosage d'action et cesse la fonctionnalité lorsque le résultat visuel le satisfait.
Les fonctionnalités d'exploration de graphe présentées dans ce chapitre sont la transitivité, permettant d'étudier le voisinage directe et indirect d'un sommet spécifique. Selon sa place dans la structure, le sommet peut alors être caractérisé comme étant un acteur majeur, dominant ou inversement sans influence. La *k-core* permet d'identifier les sous ensembles particuliers du graphe, il permet aussi d'obtenir un graphe dont les sommets ont un degré supérieur ou égal au degré fixé en paramètre et changeable à tout moment par l'utilisateur. Dans le cadre d'analyses temporelles, nous proposons le morphing de graphe, basé sur une représentation graphique semblable au principe des horloges, sur lequel chaque période étudiée est assimilée à un repère, positionné comme les heures sur une montre. Les sommets sont alors positionnés selon leur appartenance aux différentes périodes. Cette visualisation facilite la lecture des caractéristiques temporelles des entités.
Enfin, nous avons proposé une méthode de partitionnement par le Markov Clustering.
Ainsi VisuGraph permet de visualiser des volumes de données importants et permet d'analyser les structures de graphe afin de répondre à des questions, telles que « s'agit-il d'un ensemble de données relationnelles formant des ensembles extrêmement soudés ? Quel rôle a tel sommet ? Quel est son niveau d'implication au sein de la structure ? A-t-on plutôt affaire à plusieurs écoles nettement séparées les unes des autres, au sein desquelles on collabore, mais où l'on ne communique jamais avec les autres écoles ? »

5 Bibliographie

[1] **ALPERT C.J.,** Kahng A.B. *Recent developments in netlist partitioning : A survey*. The VLSI journal, vol. 19, pages 1-18, 1995.

[2] **ALVAREZ-HAMELIN J.I., DALL'ASTA L., BARRAT A., VESPIGNANI A**. *k-core decomposition: a tool for the visualization of large scale network*s. Cité pages 41, 52, 53, 54, 55, 2005.

[3] **BATAGELJ V., ZAVERSNIK M.** *Generalized cores*, 2002.

[4] **BAVELAS A**. *Communication patterns in task-oriented groups*. D. Cartwright et A.Zander (Eds), Groupe Dynamics, Nex York: Row-Peterson, pages 493-506, 1950.

[5] **BURT R.** *Structural Holes: The Social Structure of Competition.* Boston, MA: Harvard University Press, 1992.

[6] **COLEMAN, J. S.** *Loss of Power*, American Sociological Review 38, pages 1-17, 1973.

[7] **DEGENNE A., FORSE M.** *Les réseaux sociaux*, Paris, Armand Colin, 2004.

[8] **ENRIGHT A.J., VAN DONGEN S., OUZOUNIS C.A**. *An efficient algorithm for large-scale detection of protein families.* Nucleic Acids Research, vol. 30, pages 1575-1584, 2002.

[9] **EVE M.** *Deux traditions d'analyse des réseaux sociaux*. Réseaux, 20,115, pages185-212, 2002.

[10] **FAYYAD U., GRINSTEIN G.G., WIERSE A.** *Information visualization in data mining and knowledge discovery*. Morgan Kaufmann, 2002.

[11] **FRIEDKIN, N. E.** *Theoretical foundations for centrality measures*. American Journal of Sociology, 1991.

[12] **FRUCHTERMAN TMJ., REINGOLD EM.** *Graph drawing by force_directed placement*. Software – Practice and experience, 21, pages 1129-1164, 1991.

[13] **GRIBAUDI M.** *Espaces temporalités stratifications - Exercices sur les réseaux sociaux*, Ed. EHESS, Paris, 1998.

[14] **GRINSTEIN G., WARD M.** *Introduction to Data Visualization,* 2002.

[15] **HENRY T. R.** *Interactive Graph Layout: The Exploration of Large Graph*s. Department of Computer Science. Tucson, University of Arizona, 1992.

[16] **HUANG X., EADES P., LAI W.** *A Framework of Filtering, Clustering and Dynamic Layout Graphs for Visualization*. ACSC 2005, pages 87-96, 2005.

[17] **JOUVE B., KUNTZ P., VELIN F.** *Extraction de structures macroscopiques dans des grands graphes par une approche spectrale.* ECA, Hermès Science publication édition, vol. 1, pages 173-184, 2001.

[18] **KAROUACH S**. *Visualisations interactives pour la découverte de connaissances : concepts, méthodes et outils*. Thèse de Doctorat en informatique, Université Paul Sabatier, France, 2003.

[19] **KAROUACH S., DOUSSET B.** *Les graphes comme représentation synthétique et naturelle de l'information relationnelle de grandes tailles.* Dans : Workshop sur la recherche d'information, associé à INFORSID'2003, Nancy, 03/06/2003-06/06/2003, INFORSID, pages 35-48, juin, 2003.

[20] **KARYPIS G., KUMAR V.** *Multilevel k-way partitioning scheme for irregular graphs.* Journal of Parrallel and distributed Computing, vol. 48, pages 96-129, 1998.

[21] **KATZ L**. *A new status index derived from sociometric analysis.* Psychometrika, 18, pages 39-43, 1953.

[22] KEIM D. A., KRIEGEL H.P. *Using visualization to support data mining of large existing databases*. Lecture Notes in Computer Science, 871, pages 210 -229, 1994.

[23] **KUNTZ P.** *Découverte de règles d'association et de structures dans des réseaux de relations par des approches non supervisées automatiques et interactives.* Habilitation à diriger des recherches, Université de Nantes, 2003.

[24] **KUNTZ P., HENAUX F.** *Numerical comparaison of two spectral decomposition for vertex clustering*. Data Analysis, Classification and Related Methods, Proceeding Of IFCS'2000, Springer Verlag, pages 581-586, 2000.

[25] **LAZEGA E**. *Réseaux sociaux et structures relationnelles*, Paris, PUF, 1998.

[26] **LEAVITT H.J.** *Some effects of certain communication pattern on group performance*. Journal of abnormal and social psychology, 46, pages 38-50, 1951.

[27] **LOUBIER E**. *Analyse et visualisation de données relationnelles évolutives*. Rencontres Inter-Associations (RIA'S 2007), Toulouse, 12/03/2007-13/03/2007, IRIT, (en ligne), mars 2007. URL : http://www.irit.fr/RIA07/intervenants.html, 2007.

[28] **LOUBIER E.** *Visualization and analysis of large graphs*. Conference on Information and Knowledge Management (CIKM 2007), Lisbonne - Portugal, 06/11/2007-09/11/2007, ACM, (support électronique), 2007.

[29] **LOUBIER E., CARBONNEL S.** *Influence du prétraitement textuel sur la représentation graphique dans un contexte d'analyse de données relationnelles*. Colloque Veille Stratégique Scientifique et Technologique (VSST 2007), Marrakech, IRIT, support électronique, 2007.

[30] **LOUBIER E., BAHSOUN W., DOUSSET B.** *La prise en compte de la dimension temporelle dans la visualisation de données par morphing de graphe*. Colloque Veille Stratégique Scientifique et Technologique (VSST 2007), Marrakech, IRIT, (support électronique), 2007.

[31] **LOUBIER E., DOUSSET B**. *La prise en compte de la dimension temporelle dans la classification de données*. Journées Francophones Extraction et Gestion de Connaissances (EGC 2008), Sophia Antipolis, 2007.

[32] **LOUBIER E., DOUSSET B.** *Temporal and relational data representation by graph morphing*. Safety and Reliability for managing Risk (ESREL 2008), Hammamet, 2008.

[33] **LOUBIER E.** *Proposition d'un algorithme de placements temporels des sommets d'un graphe évolutif*. Congrès VSST'2009 : Veille Stratégique Scientifique & Technologique, Nancy, support électronique, 2009.

[34] **MELANÇON G., HERMAN I., DELAST M.** *Indices visuels et métriques combinatoires pour la visualisation de données hiérarchique*. Proceedings of the IHM'99 Workshop, Montpellier, pages 166-173, 1999.

[35] **MERCKLE P**. *Sociologie des réseaux sociaux*. La Découverte, Paris, 2004.

[36] **PINAUD B., KUNTZ P.** *Un guide sur la Toile pour sélectionner un logiciel de tracé de graphes*. Congrès VSST'2004 : veille stratégique scientifique & technologique : Systèmes d'information élaborée, bibliométrie, linguistique, intelligence économique , pages 546-540 , 2004.

[37] **VAN DONGEN S.** *Graph Clustering by Flow Simulation*. Thèse de doctorat, Université d'Utrecht, Allemagne, 2000.

[38] **WASSERMAN S., FAUST K.** *Social Network Analysis, Methods and Applications*, Cambridge, Mass., Cambridge University Press, 1994.