

La visualisation des réseaux. Principes, enjeux et perspectives

Françoise Bahoken*, Laurent Beauguitte** et Serge Lhomme***

Juin 2013

*IFSTTAR ET UMR GÉOGRAPHIE-CITÉS **UMR IDEES ***EIVP



Introduction

La visualisation graphique des structures relationnelles est devenue un domaine de recherche foisonnant, rendu possible par l'ouverture de masses de données, en lien avec le développement de nouvelles méthodes d'analyse et l'augmentation des potentialités des supports-écran. Les objets/entités considérés dans la théorie des graphes sont une liste de sommets et de liens, leur représentation graphique consiste alors à visualiser la structure des relations entre ces entités, sous une forme ou sous un autre. Le graphique propose, impose une image structurée du phénomène étudié qui sera d'autant plus lisible et compréhensible qu'un effort aura été fait sur les aspects liés au design (visuel), à l'esthétique de la figure. Or, la réflexion sur la visualisation des structures relationnelles, si elle est centrale dans les travaux de chercheurs en informatique graphique (voir les travaux de Jean-Daniel Fekete à l'INRIA [24] [13], d'Ulrik Brandes [8] [7] ou la revue en ligne *Journal of Graph Algorithms and Applications*), reste peu développée. Les manuels classiques d'analyse de réseaux, tels que ceux de Wasserman et Faust [42] ou de Newman [31] ne proposent pas de chapitre consacré à la visualisation.

Et pourtant, certains logiciels proposent des plusieurs dizaines de modes de visualisations différents d'un même graphe. Si la tentation est forte de les tester, les uns après les autres, afin de sélectionner celui qui nous séduit, encore faut-il être en mesure de faire le choix adéquat. Aussi nous est-il apparu essentiel de présenter les principes généraux des algorithmes de génération de ces images qui, si elles sont souvent attractives et expressives, présentent un aspect boîte noire pour ce qui est de leur mise en œuvre, de leur intérêt dans l'analyse. Une réflexion conjointe sur les méthodes et les outils utilisés

est, en effet, le seul moyen de choisir l'algorithme le plus adapté aux données que l'on étudie, en fonction du processus que l'on souhaite donner à voir.

Nous rappelons dans une première section, les principes généraux relatifs à la visualisation spécifique des graphes, certains ayant été établis dès les années 1930 ; puis nous présentons les algorithmes les plus fréquemment utilisés ; enfin, les alternatives proposées au graphe pour visualiser les données relationnelles.

1 Principes de base

Visualiser des données quantitatives

D'après W. Tukey (1977 [39]), la représentation graphique est un outil puissant d'exploration et d'analyse des données numériques. Elle prend des formes variées qui sont plus complémentaires qu'antagonistes, elle autorise aussi leur perception selon différentes perspectives. L'intérêt du graphique est tel qu'il a souvent accompagné l'innovation scientifique (Börner, 2010 [6]). Complémentaire aux traitements des données réalisés en amont, la visualisation permet d'illustrer graphiquement les résultats, mais également de susciter des questionnements, de valider des hypothèses. Elle acquiert ainsi une importance particulière dans le domaine de l'analyse des structures relationnelles caractérisées par une complexité intrinsèque.

La visualisation de données sous la forme de diagramme lien-nœud - plus connu sous le terme de graphe - a donné lieu à de nombreuses réflexions, à des développements qui ont conduit à des recommandations générales qu'il serait judicieux de suivre. Il s'agit en particulier des trois principes de l'excellence graphique définis par Tufte [38] : « L'excellence graphique est une présentation bien conçue des données d'intérêt [...], [elle] vise à communiquer des idées complexes avec clarté, précision et efficacité ; [elle] est ce qui donne au lecteur le plus grand nombre d'idées dans le temps le plus court, tout en utilisant le moins d'encre sur un minimum d'espace » (Tufte, 1983, p. 51, adaptée par nos soins¹).

Les principes de sémiologie graphique rassemblés et formalisés par Jacques Bertin [4][5] méritent eux-aussi d'être rappelés : seule la variable visuelle taille permet de représenter des données quantitatives portant sur des stocks quand la valeur, la couleur ou le grain traduisent des relations quantitatives continues, segmentées en classes. Il est donc absurde de représenter des sommets de même taille avec une variation de couleurs pour exprimer des

1. La citation originale est la suivante : “*Graphical excellence is the well-designed presentation of interesting data [...]. Graphical excellence consists of complex ideas communicated with clarity, precision and efficiency. Graphical excellence is that which gives to the viewer the greatest number of ideas in the shortest time with the least ink in the smallest place*”.

écarts de degrés... Pour cela, il est nécessaire de transformer au préalable les données, en les normalisant par exemple.

Une contrainte graphique, parfois négligée, est le principe d'instantanéité de transmission, *a fortiori* de compréhension, du message transmis par l'image. Celui-ci nécessite la conception d'une figure qui permette de repérer instantanément, du premier coup d'œil, le type et la nature des relations représentées ; de discriminer les différents figurés ; de percevoir le sens des variations de forme, de couleur ou de valeur utilisées. Il est aussi impératif que cette image soit accompagnée d'une légende, tout comme il est recommandé de préciser les outils de visualisation utilisés (logiciel et algorithmes).

Du sociogramme au graphe : les règles élémentaires de la lisibilité

Dès les premiers sociogrammes dessinés manuellement par Moreno dans les années 1930, un certain nombre de conventions vont être utilisées pour représenter les graphes ; elles vont ensuite s'imposer et être intégrées dans la plupart des algorithmes existants. Trois grands principes de représentation s'imposent dès le départ :

- les sommets les plus connectés sont placés au centre de la figure ;
- les sommets les moins connectés sont placés en périphérie de la figure ;
- il faut limiter, autant que possible, le chevauchement des liens .

Si les deux premières règles sont facilement applicables, la troisième pose davantage de problèmes, la lisibilité des cliques et des sous graphes denses étant souvent délicate.

A noter que Moreno faisait le choix de représenter, sur le même graphe, les phénomènes d'attractions et de répulsions, à l'aide de symboles géométriques (ligne, cercle, triangle) de couleurs noires et rouges². Ce mode de représentation n'est plus utilisé aujourd'hui, non pas parce qu'il manque d'efficacité, mais parce qu'il ne semble pas faire partie des possibilités offertes par les logiciels actuels³...

Lisibilité, variables visuelles et progrès informatique

Plus récemment, certains auteurs ont proposé des améliorations graphiques mobilisant les variables visuelles. Dans le récent manuel de Hennig *et al.* [23], un chapitre entier est consacré aux variations de forme, de valeur, de trame et de grain pour améliorer l'efficacité des graphes. L'apport de la visualisation interactive⁴, et / ou en 3D sur le web, *via* des logiciels de

2. Voir les figures reproduites à l'adresse <http://groupefmr.hypotheses.org/1069>.

3. Ce type de représentation est cependant possible avec des SIG et avec le logiciel Tulip.

4. On peut notamment citer, comme algorithmes de dessin de graphes dynamiques : GraphAEL (C. Erten *et al.*, 2004 [12]) qui produit simultanément, en les recalculant, des images des différents états du graphe ; et l'algorithme de Y. Frishman et A. Tal (2008 [15])

plus en plus performants a été souligné par Freeman [14]. L'auteur propose d'ailleurs une intéressante rétrospective mêlant considérations scientifiques (quelle visualisation pour quel message?) et contraintes techniques. Un panorama sur les innovations informatiques en matière de visualisation avait été réalisé dès 1981 dans un des rares articles de la revue *Social Networks* dédiés à ce sujet (Klovdhal [27]).

La conclusion de Klovdhal mérite d'être citée dans la mesure où la visualisation est présentée avant tout comme un outil d'exploration, d'aide à la recherche, et non pas comme un outil de communication :

One very important feature of computer generated visual representations is that their use - by supplementing visual imagery and by permitting human pattern recognition capabilities to be more fully utilized - can help to stimulate the research questions, hypotheses, and theoretical insights - the new ideas - that are the driving force in any scientific enterprise.

Ces propositions sont généralement pertinentes. Cependant, si elles s'inspirent des travaux de Bertin, elles n'apparaissent pas suffisantes pour régler le problème relativement récent de la visualisation des grands graphes (parfois appelés denses ou graphes complexes).

Visualiser les grands graphes

Étant donné que la lisibilité et la fidélité aux données de départ sont deux contraintes essentielles pour la visualisation, que dire alors de la tendance récente à représenter des graphes comprenant plusieurs milliers ou plusieurs dizaines de milliers de sommets et de liens? ⁵ Représenter l'ensemble de ces liens est un défi technique et esthétique, il n'est pas évident qu'il présente un quelconque intérêt scientifique.

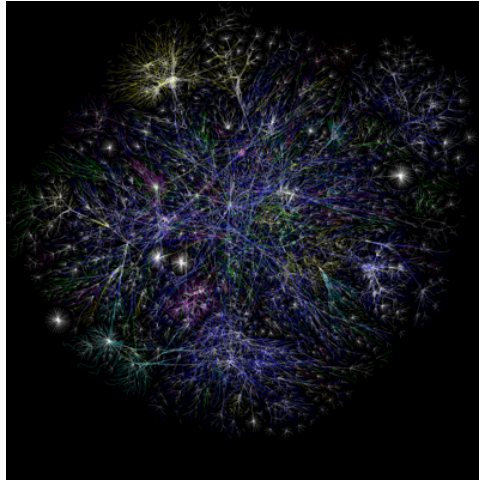
Il n'est, en effet, pas rare que le principal enseignement d'un graphe dense soit la mise en évidence d'une sélection de sommets très centraux (voir les travaux sur la structure du web). Or ces propriétés peuvent être révélées par des méthodes plus robustes telles que la distribution des degrés et par un mode de représentation adapté. Sa visualisation, si elle est parfois spectaculaire, n'apporte dans bien des cas qu'une valeur ajoutée très relative aux traitements des données réalisés en amont.

L'image 1 est sans doute superbe mais que nous apprend-elle?... Lire l'historique du projet (<http://www.opte.org/history/>) permet de comprendre que son enjeu principal était de relever un défi technique : produire en une journée une carte du web.

qui recalculent les dessins qui correspondent à différents états du graphe.

5. Le site *visual complexity* s'est fait une spécialité de ce type de représentation (<http://www.visualcomplexity.com/vc/>).

FIGURE 1 – L'esthétique avant tout, une visualisation du web



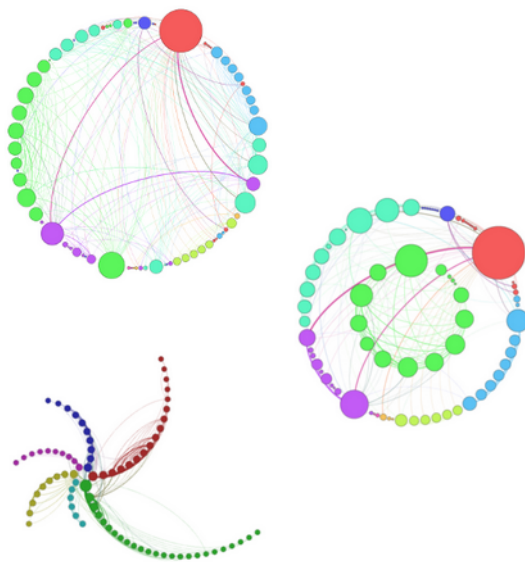
Source : <http://www.opte.org/maps/>

2 Quelques algorithmes courants

Les algorithmes classiques de la visualisation des graphes sont fondés sur différentes méthodes liées au placement des sommets pour minimiser les chevauchements des liens (Koren et Harel, 2002 [28]). On peut citer l'analyse spectrale (Harel et Koren, 2002 [21]); les graphes arborescents (sans cycle) ou sous la forme d'arbres hiérarchiques; les représentations orthogonales, orthogonales, symétriques ou encore circulaires (voir Bahoken, 2011 [2]). Dans les faits, l'ensemble des algorithmes de visualisation pourrait désormais être regroupé en deux grandes familles. La première famille concerne les algorithmes fondés sur des propriétés mathématiques. Plus précisément, ces algorithmes reposent sur les techniques issues de l'analyse de données. La deuxième famille est plus récente, elle concerne notamment les algorithmes de dessin fondés sur les forces (*Force-based* ou *Force-directed algorithms*). Cette famille est quant à elle fondée sur des analogies issues de la physique.

Les algorithmes issus de l'analyse de données déterminent la position des nœuds en appliquant un traitement statistique sur la matrice d'adjacence. Cette matrice n'est autre qu'une matrice de similarité (de proximités) ou de covariance (pour les analyses en composantes principales). Les algorithmes développés tentent alors de représenter au mieux cette matrice dans un espace en deux ou trois dimensions. Ainsi les distances observées doivent être aussi proches que possible de celles obtenues à partir de l'algorithme. L'objectif est donc de minimiser une fonction nommée : « fonction de stress ». Pour cela, les algorithmes développés déterminent les valeurs et les vecteurs

FIGURE 2 – Circular layout



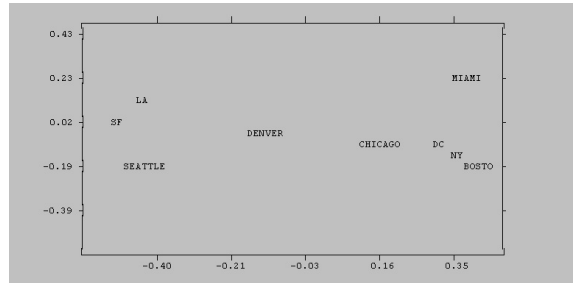
Source : Matt Groeninger (2010). Code source pour Gephi : <https://marketplace.gephi.org/plugin/circular-layout/>

propres des matrices. C'est pourquoi, il est souvent fait référence au terme d'analyses spectrales. Par exemple, le *MultiDimensional Scaling* (MDS) calcule les valeurs propres et vecteurs propres d'une matrice afin de déterminer la « meilleure » projection de l'ensemble d'objets dans un sous-espace de dimension donnée. Si la matrice de départ est une matrice de distances euclidiennes, le MDS fournit en résultat les coordonnées des n points, à une rotation près, dans un repère dont l'origine est le centre de gravité des points.

Plus récemment, des algorithmes de dessin fondés sur les forces ont été développés. Ils connaissent un succès indéniable. En effet, comme leur vocation première est de faciliter la visualisation des graphes et non l'analyse de données, ils peuvent apparaître plus efficaces que ces derniers. Ces algorithmes utilisent un système de forces appliquées entre les nœuds et les arcs (Fruchterman et Reingold, 1991 [16]). Ils peuvent ainsi être décrits à l'aide d'une analogie issue de la physique. Les nœuds peuvent être considérés comme des particules de même charge qui se repoussent et les arcs peuvent être assimilés à des ressorts qui tendent à rapprocher les nœuds voisins. Ces algorithmes cherchent alors à minimiser l'énergie du système en déplaçant les nœuds. Les déplacements effectués changent en effet les forces entre les nœuds.

Ces algorithmes sont réputés pour être plutôt flexibles. Ils permettent

FIGURE 3 – MultiDimensional Scaling



Source : Stephen P. Borgatti (1997)
<http://www.analytictech.com/borgatti/mds.htm>

FIGURE 4 – Tableau comparatif des algorithmes de dessin fondés sur les forces

Criteria	Di Battista et al. (1994)	Eades (1984)	Kamada and Kawai (1989)	Fruchterman and Reingold (1991)	Davidson and Harel (1996)	NicheWorks (1997)
Symmetric	✓	✓	✓			
Evenly distributed nodes	✓		✓	✓	✓	Clustered
Uniform edge lengths	✓	✓	✓	✓	✓	Weights
Minimized edge crossings	✓		✓	✓	✓	

Source : Chen (2006 [9])

d'obtenir des visualisations jugées « esthétiques » avec notamment l'apparition de symétries. Néanmoins, les temps de calcul peuvent être relativement longs. Les graphes doivent donc être d'une taille relativement réduite pour que ces algorithmes se révèlent pertinents (la limite se situant au environ de 1000 nœuds). Des algorithmes ont été mis en place pour traiter des graphes de plus de 10 000 nœuds, ils s'appuient sur des représentations multi-niveau allant du plus simple au plus complexe (Harel et Koren, 2002 [21]). Néanmoins au-delà de ces points communs, chaque algorithme présente ses particularités, car ils ne reposent pas tous sur les mêmes critères esthétiques.

Tous les algorithmes de dessin fondés sur les forces sont issus d'une idée initiale de Tutte (1963 [40]). Puis, une première famille d'algorithmes est définie par les travaux de Eades [11]. Cet algorithme définit en effet précisément l'idée de forces répulsives et de forces attractives. Les forces attractives sont

alors calculées pour les nœuds voisins et les forces répulsives sont calculées pour toutes les paires de nœuds. A chaque itération, tous les nœuds sont déplacés simultanément. L'algorithme fait la somme des forces appliquées puis déplace les nœuds suivant des règles de physique classique jusqu'à trouver un état stable (*e.g.* qui minimisent l'énergie de l'ensemble des ressorts), mais la force d'attraction définie par Eades ne correspond pas à celle d'un ressort, elle suit une loi utilisant un logarithme. C'est Fruchterman et Reingold [16] qui utiliseront les premiers l'analogie du ressort. Les forces sont alors calculées en ayant recours à la loi de Hooke qui permet de définir l'élasticité d'un ressort.

L'algorithme de Kamada-Kawai relève d'une logique un peu différente [26]. A partir d'une répartition initiale des nœuds (en anneau par exemple), les nœuds sont déplacés un par un. Cet algorithme cherche alors à minimiser l'énergie de l'ensemble des ressorts et repose sur la définition d'une distance idéale entre chaque nœud. C'est à Kamada et Kawai que l'on doit l'idée de remplacer les arcs par des ressorts. Ils introduisent ainsi la loi de Hooke dans les algorithmes de visualisation. Contrairement à d'autres algorithmes, celui-ci ne vise pas à limiter les croisements. D'autre part, cet algorithme peut rapprocher, voire superposer, des nœuds qui ne sont pas forcément proches trompant le lecteur en matière de centralité ou de partitionnement (communauté).

L'algorithme *LinLog* (Noack, 2007 [32]) permet de repérer visuellement plus facilement des communautés car la fonction ressort agit davantage sur les sommets qui présentent des degrés forts. Une classification des sommets (en graphe hiérarchique, par exemple, Noack et Rotta, 2009 [33]) est souvent rendue nécessaire au préalable de la représentation pour sélectionner ceux qui seront représentés. Le principe est dès lors celui de la représentation des graphes classifiés. Enfin, *Force-Atlas*, développé par les concepteurs de Gephi, repose sur le même principe. Dans ce cadre, les mécanismes de fonctionnement semblent plus facilement compréhensibles par les utilisateurs.

3 Les alternatives à la visualisation classique

Face à la difficulté de visualiser, mais aussi d'explorer aisément les structurelles relationnelles denses, à l'aide de sociogrammes et de graphes, le mode de représentation sous la forme de matrice de valeurs ordonnées s'est développé dans le champ d'analyse des réseaux et ce, dès les années 1960. La théorie matricielle, notamment les opérations d'ordonnancement, de permutation des lignes et des colonnes, ainsi que les questions liées à leur visualisation a ensuite su tirer parti des apports de l'informatique.

Les matrices bloquées

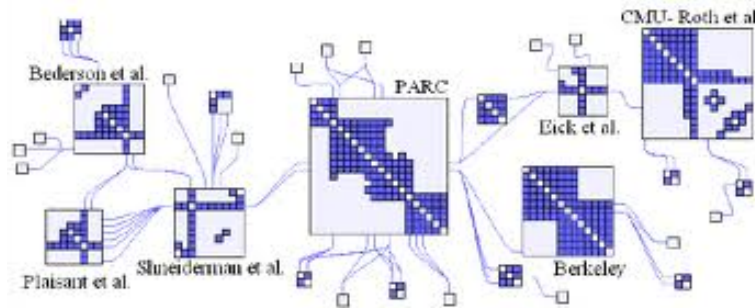
Le mode de représentation en matrice-ordonnée (ou matrice-Bertin) a été mis au point d’abord manuellement par J. Bertin ⁶, à la fin des années 1960. Il s’agit d’une méthode de traitement graphique et visuel de données quantitatives développée à l’heure où les outils statistiques de classification n’étaient ni généralisés ni informatisés. Ordonner une matrice consiste à recoder ses valeurs, en permutant ses lignes et ses colonnes. Ce recodage conduit à une transformation de la matrice de manière à aboutir à une partition qualifiée de matrice bloquée (*blocked matrix*). Plusieurs partitions de la matrice en (deux ou) plusieurs sous-matrices (et donc sous-graphes) homogènes, peuvent être générées. De la même façon que les sommets (ou les liens) peuvent être agrégés afin d’obtenir une partition de niveau supérieur qualifiée de matrice de densité (*density matrix*).

Les matrices hybrides

Le procédé de la matrice ordonnable a été transféré ces dernières années au champ de la visualisation des réseaux (Ghoniem *et al.*, 2004 [17] ; Henry et Fekete, 2008 [24] ; Guo, 2009 [18] ; Wilkinson, Friendly, 2009 [43]). Il est devenu un objet de recherche à part entière. Si N. Henry et J.-D. Fekete parlent de « matrices » ou de « matrice-Bertin », on notera que Guo utilise plutôt l’expression « matrice OD » ; on rappelle que J. Bertin (1967) utilise le terme de « matrice ordonnable ». Le potentiel exploratoire de ce mode de représentation a été démontré dans le cadre de la recherche de structures internes des grandes matrices, celle des communautés par exemple (Fekete et Henry, 2008). Guo (2008, 2009) l’a utilisé conjointement avec des traitements statistiques d’agrégation quand d’autres ont combiné matrice-ordonnée et graphe (Ghoniem *et al.*, 2004) pour proposer des formes hybrides de représentation matricielle. Les matrices hybrides proposées par Henry et Fekete (*op. cit.*) s’appuient sur différents algorithmes de ré ordonnancement optimisé des matrices d’adjacence et sur plusieurs modes de représentation. Elles permettent de visualiser les données, au sein d’une même figure, conjointement sous la forme de graphe et de matrice ordonnée « augmentée » où les liens sont placés sur les bords de la matrice, la matrice ordonnée « hybride », le graphe permettant ici la représentation détaillée d’une communauté précise (voir figure 5).

6. J. Bertin préconise ce recodage des valeurs en teintes de couleurs, car il considère que la difficulté de lecture des valeurs numériques de la matrice est préjudiciable à l’analyse. Elle n’autorise pas une vision globale de la structure des relations.

FIGURE 5 – Matrice hybride



Source : Henri et Fekete 2008 (*op. cit.*)

4 Représenter les graphes non simples

Si la réflexion relative à la visualisation des graphes simples et éventuellement valués est ancienne, la visualisation de graphes plus spécifiques (bipartis, multigraphe ou multi-niveaux) a été moins questionnée.

La visualisation des graphes bipartis

Le graphe biparti représentant les relations entre deux ensembles distincts de sommets (Beauguitte, 2012 [3]), la tendance la plus fréquente pour sa représentation est d'aligner en haut l'un des groupes, en bas l'autre groupe et de tracer les liens entre les deux. Cette habitude ne limite pas les chevauchements de liens qu'il serait facile d'éviter si ces contraintes - arbitraires - de position n'étaient pas respectées.

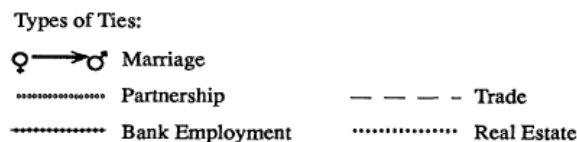
Dans le récent manuel de Brandes *et al.* où la visualisation de ces graphes est abordée, la seule proposition visant à améliorer la lisibilité des graphes bipartis consiste à réordonner les sommets pour limiter les chevauchements de liens.

Variation de la couleur et/ou la forme en fonction de l'appartenance à un des deux groupes de sommets est sans doute la solution la plus élégante pour éviter l'alignement et les chevauchements de lignes. Néanmoins, et contrairement à ce qui existe pour les graphes simples, il n'existe à notre connaissance pas d'algorithme de visualisation reposant sur des principes mathématiques ou de type *energy*.

Les défis du multigraphe

Les méthodes même d'analyse des multigraphes consistant le plus souvent à agréger les liens pour produire un graphe valué simple [10], il n'est guère étonnant que peu de propositions soient faites dans ce domaine. Représenter

FIGURE 6 – Multigraphe et variation des figurés



Source : Padgett et Ansell, 1993

deux relations (ou plus) simultanément multiplie en effet les problèmes de lisibilité et de compréhension. Un algorithme de positionnement des sommets peut prendre en compte une configuration donnée mais ne saurait s’appliquer à deux configurations différentes.

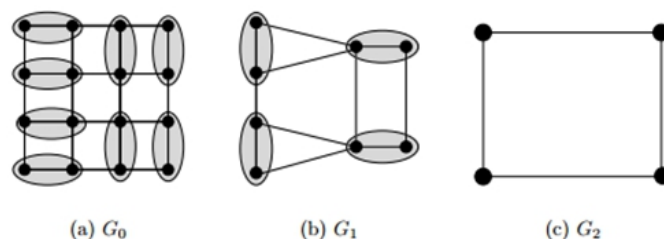
L’une des rares possibilités, mais elle n’est valable que pour des graphes peu denses et de taille réduite, consiste à utiliser des figurés (forme, couleur) différents pour des liens de nature différente. Ainsi l’article devenu classique de Padgett et Ansell sur les familles florentines à la Renaissance [34] mobilise jusqu’à cinq types de liens différents (voir figure 6).

Représenter les graphes multi-niveaux

Concernant les réseaux multi-niveaux, il apparaît dans la littérature que l’enjeu principal n’est pas de représenter au mieux ces graphes, en utilisant des algorithmes de visualisation spécifiques, mais plutôt de transformer de grands graphes (des graphes complexes) en graphes multi-niveaux afin d’améliorer l’esthétique de la représentation et de réduire les temps de calcul. La question centrale qui est abordée succinctement ici n’est donc pas « comment visualiser des graphes multi-niveaux ? » mais « comment transformer un grand graphe en un graphe multi-niveaux ? »

Pour faire simple, les algorithmes de visualisation multi-niveaux vont partitionner le graphe et agréger certains composants du graphe (des nœuds et des arcs) selon un processus itératif afin de le simplifier [19]. Ainsi, le graphe initial sera progressivement transformé en un graphe de plus en plus « simple » nommé : *coarse graph* (graphe « grossier »). Le postulat principal sur lequel ces recherches reposent est le suivant : les graphes simples produits à l’aide de ces algorithmes sont censés être représentatifs du graphe initial [19]. La perte d’informations induite par la simplification ne serait pas nuisible et conserverait la topologie et les principales propriétés de l’ensemble. L’esthétique générale d’un graphe dépendrait dès lors de sa macrostructure (de son squelette) et dépendrait peu de sa microstructure (des détails). Or ces graphes plus simples (en l’occurrence moins grands) peuvent être analysés plus vite. Il devient dès lors possible d’appliquer certains algorithmes

FIGURE 7 – Un procédé simple de création d'un graphe multi-niveaux par contraction des arcs du graphe a) au graphe c)



Source : Stefan Hachul (2005 [19])

de visualisation, comme par exemple ceux fondés sur les forces, à de grands graphes. Le terme multi-niveaux utilisé pour qualifier ces algorithmes est très pertinent. En effet, chaque graphe simple représente un graphe d'un niveau supérieur au graphe initial (un « macrographe »). Si dans la plupart des cas, seul le résultat final de la visualisation est représenté en 2D, il est tout à fait envisageable de représenter l'ensemble des graphes engendrés par l'algorithme multi-niveaux à l'aide d'une représentation 3D.

Le premier algorithme multi-niveaux a été développé par Hadany et Harel pour des travaux en physique des particules [20]. Cet algorithme a été simplifié et amélioré par Harel et Koren [22]. Il consiste à identifier un certain nombre de centres (en identifiant les solutions du problème d'optimisation dit p-centré) et à définir un rayon à partir duquel tous les nœuds sont agrégés en un seul. D'autres algorithmes proposent des modes d'agrégation plus simples, comme ceux reposant sur la contraction des arcs. Enfin, des algorithmes proposent d'identifier certaines configurations particulières (des arbres, des graphes complets...) afin de les agréger en un nœud et de les faire apparaître à l'aide d'un code couleur [1].

Indéniablement, ces algorithmes constituent une avancée majeure, car au-delà des simples gains en matière de temps de calcul, il est tout à fait pertinent de simplifier un grand graphe, dont certaines informations peuvent apparaître nuisibles à sa visualisation. De plus, à l'aide d'outils informatiques, il est possible de ne pas perdre d'informations, l'utilisateur pouvant alors librement parcourir et explorer les différents niveaux du graphe.

Conclusion

Si la réflexion relative à la visualisation des graphes simples et éventuellement valués est ancienne, la visualisation de graphes plus spécifiques (bipartis, multigraphes ou multi-niveaux) semble avoir été moins question-

née.

Les développements récents réalisés font notamment état de méthodes liées à la fusion graphique des nœuds et des liens (*edge bundling*) pour éliminer visuellement le problème du chevauchement des liens, tout en conservant (presque) toute l'information disponible. Si le rendu graphique est visuellement agréable, il est intéressant de noter que le procédé fut antérieurement proposé par W. Tobler (1987 [37]) pour la représentation cartographique des matrices de flux, sans que ce dernier soit d'ailleurs mentionné par ses auteurs (Phan *et al.*, 2005 [36]; Holten, 2006 [25]; Lambert *et al.*, 2009 [30]; Lambert *et al.* 2010 [29]). K. Verbeek *et al.* (2011 [41]) ont récemment remis en perspective le procédé, sous le nom d'arbre en spirale.

Les exemples de représentation graphique des réseaux, considérés comme prétendument innovants, renforcent l'hypothèse selon laquelle les développements (informatiques) ne s'inscrivent pas tant dans un contexte d'émergence d'un type particulier de connaissances. Ils visent surtout à donner du sens à la prolifération d'images disponibles, en recyclant des méthodes anciennes qui conduisent à changer l'image que nous en avons, ce changement étant lié aux potentialités algorithmiques qui renouvellent des connaissances anciennes.

G. Palsky [35] a d'ailleurs noté l'ingéniosité des solutions graphiques élaborées au XIX^e siècle, rappelant leur actualité au XXI^e siècle en ces termes :

Les outils graphiques des tableurs, ou la cartographie assistée par ordinateur ne nous fournissent pas des images très différentes de celles qui étaient obtenues il y a un siècle et demi. Ce caractère faiblement évolutif est un indice de l'autonomie du visuel par rapport aux données d'information.

Ainsi, malgré les efforts déployés par les développeurs, les images obtenues ne contribuent guère à résoudre le problème de la visualisation des graphes, surtout lorsqu'ils ne sont pas planaires, à un niveau autre que graphique et esthétique. U

Références

- [1] Daniel ARCHAMBAULT, Tamara MUNZNER et David AUBER : Topo-Layout : Graph Layout by Topological Features. *Information Visualization*, pages 12–13, 2005.
- [2] Françoise BAHOKEN : Représentation graphique des matrices : graphe et/ou carte des flux? *Groupe fmr*, 2011. <http://halshs.archives-ouvertes.fr/FMR/fr/>.
- [3] Laurent BEAUGUITTE : L'analyse des graphes bipartis. *Groupe fmr*, 2013. <http://halshs.archives-ouvertes.fr/FMR/fr/>.
- [4] Jacques BERTIN : *Sémiologie graphique. Les diagrammes - Les réseaux - Les cartes*. Éditions Gauthier-Villiar, Paris, 1967.

- [5] Jacques BERTIN : *La graphique et le traitement graphique de l'information*. Flammarion, Paris, 1977.
- [6] Katy BÖRNER : *Atlas of Sciences. Visualizing what we know*. MIT Press, 2010.
- [7] Ulrik BRANDES : *Layout of Graph Visualizations*. Thèse de doctorat, Universität Konstanz, 1999.
- [8] Ulrik BRANDES, Patrick KENIS, Jörg RAAB, Volker SCHNEIDER et Dorothea WAGNER : Explorations into the visualization of policy networks. *Journal of Theoretical Politics*, 11(1):75–106, 1999.
- [9] Chaomei CHAN : *Information visualization : Beyond the horizon*. Springer, Berlin, 2006.
- [10] César DUCRUET : Multigraphes, multiplexes et réseaux couplés. *Groupe fmr*, 2012. <http://halshs.archives-ouvertes.fr/FMR/fr/>.
- [11] Peter EADES : A heuristic for graph drawing. *Congressus Numerantium*, 42(11):149–160, 1984.
- [12] C. ERTEN, P.J. HARDING, S.G. KOBOUROV, K. WAMPLER et G.V. YEE : GraphAEL : Graph animations with evolving layouts. *Proceedings of Graph Drawing*, 2912:98–110, 2004.
- [13] Jean-Daniel FEKETE : Visualizing networks using adjacency matrices : Progresses and challenges. *In Proceedings of the 11th IEEE International Conference on Computer-Aided and Computer Graphics, CAD/Graphics*, 2011.
- [14] Linton FREEMAN : Visualizing Social Networks. *Journal of Social Structures*, 1(1), 2000.
- [15] Yaniv FRISHMAN et Ayellet TAL : Online dynamic graph drawing. *Proceedings of the IEEE Transactions on Visualization and Computer Graphics*, 14(4):727–740, 2008.
- [16] Thomas M.J. FRUCHTERMAN et Edward M. REINGOLD : Graph drawing by force-directed placement. *Software Practice Experience*, 21(11):1129–1164, 1991.
- [17] M.J. GHONIEN, J. FEKETE et P. CASTOGLIOLA : A comparison of the readability of graphs using node-link and matrix-based representation. *In Proceedings of the IEEE Symposium on Information Visualization Infovis*, pages 17–24, 2004.
- [18] D. GUO : Flow mapping and multivariate visualization of large spatial interaction data. *In Proceedings of the IEEE Symposium on Information Visualization and Computer Graphics*, volume 15, pages 1041–1048, 2009.
- [19] Stefan HACHUL et Michael JÜNGER : Drawing Large Graphs with a Potential-Field-Based Multilevel Algorithm. *In Graph drawing*, pages 285–295. Springer Berlin / Heidelberg, 2005.

- [20] Ronny HADANY et David HAREL : A multi-scale algorithm for drawing graphs nicely. *Discrete Applied Mathematics*, 113(1):3–21, 2001.
- [21] D. HAREL et Y. KOREN : Fast Drawing of Large Graphs. *Journal of Graph Algorithms and Applications*, 6(3):179–183, 2002.
- [22] David HAREL et Yehuda KOREN : A fast multi-scale method for drawing large graphs. *In Proceedings of the working conference on Advanced visual interfaces*, pages 282–285. ACM Press, 2000.
- [23] Marina HENNIG, Ulrik BRANDES, Jürgen PFEFFER et Ines MERGEL : *Studying Social Networks. A Guide to Empirical Research*. Campus Verlag, Frankfurt, 2012.
- [24] Nathalie HENRY et Jean-Daniel FEKETE : Représentations visuelles alternatives pour les réseaux sociaux. *Réseaux*, (152):59–92, 2008.
- [25] Danny HOLTEN : Hierarchical edge bundles : Visualization of adjacency relations in hierarchical data. *In Proceedings of the IEEE Transactions on Visualization and Computer Graphics*, volume 12, pages 741–748, 2006.
- [26] T. KAMADA et S. KAWAI : An algorithm for drawing general undirected graphs. *Information Processing Letters*, 31(1):7–15, 1989.
- [27] Alden S. KLOVDAHL : A Note on Images of Networks. *Social Networks*, 3(3):197–214, 1981.
- [28] Yehuda KOREN, L. CARMEL et David HAREL : Ace : A fast multiscale eigenvectors computation for drawing huge graphs. *In Information Visualization IEEE*, pages 137–144, 2002.
- [29] Antoine LAMBERT, David AUBERT et Guy MELANÇON : Living flows : enhanced exploration of edge-bundled graphs based on gpu-intensive edge rendering. *In Proceeding of the 14th International Conference on Information Visualization*, pages 523–530, 2010.
- [30] Antoine LAMBERT, R. BOURQUI et David AUBER : Winding roads : Routing edges into bundles. *In Proceedings of the Eurographics / IEEE-VGT Symposium on Visualization*, volume 29, pages 853–863, 2009.
- [31] Mark E.J. NEWMAN : *Networks : An introduction*. Oxford University Press, 2010.
- [32] Andreas NOACK : Energy models for graph clustering. *Journal of Graph Algorithms and Applications*, 11(2):453–480, 2007.
- [33] Andreas NOACK et Randolph ROTTA : Multi-level algorithms for modularity clustering. *In Proceedings of the 8th International Symposium on Experimental Algorithms*, pages 4257–268. Springer-Verlag, 2009.
- [34] John F. PADGETT et Christopher K. ANSELL : Robust Action and the Rise of the Medici, 1400–1434. *American Journal of Sociology*, 98(6): 1259–1319, 1993.

- [35] Gilles Palsky : *Les cartes de la connaissance*, chapitre Le calcul par l'oeil, pages 588–605. Karthala-Urbama, 2004.
- [36] Doantam PHAN, Ling XIAO, Ron YEH, Pat HANRAHAN et Terry WINOGRAD : Flow map layout. *In Proceedings of IEEE Symposium on Information Visualization*, pages 219–224, 2005.
- [37] Waldo TOBLER : Experiments in migration mapping by computer. *The American Cartographer*, 14(2):155–163, 1987.
- [38] Edward R. Tufte : *The Visual Display of Quantitative Information*. Graphics Press, Cheshire, 1983.
- [39] John W. Tukey : *Exploratory data analysis*. Addison-Wisley Publishing, 1977.
- [40] W.T. Tutte : How to draw a graph. *Proceedings of the London Mathematical Society*, 13(May 1962):743–767, 1963.
- [41] K. Verbeek, K. Buchin et B. Speckmann : Flow map layout via spiral trees. *In Proceedings of the IEEE Transactions on Visualization and Computer Graphics*, volume 17, pages 2536–2544, 2011.
- [42] Stanley Wasserman et Katherine Faust : *Social Network Analysis. Methods and Applications*. Structural analysis in the social sciences. Cambridge University Press, 1994.
- [43] L. Wilkinson et M. Friendly : The history of the cluster heat map. *The American Statistician*, 53(2):179–184, 2009.