Visualisation d'information : techniques et solutions visuelles pour l'exploration de données relationnelles, temporelles et spatiales

Arnaud Sallaberry





HABILITATION À DIRIGER DES RECHERCHES

UNIVERSITÉ DE MONTPELLIER
ÉCOLE DOCTORALE I2S - INFORMATIQUE

Visualisation d'information : techniques et solutions visuelles pour l'exploration de données relationnelles, temporelles et spatiales

---

Arnaud Sallaberry

LIRMM - Université de Montpellier - CNRS
Groupe AMIS - Université Paul-Valéry Montpellier 3

20 Novembre 2020

Pascale Kuntz	Professeure, LS2N - Université de Nantes, France	Rapportrice
Michael J. McGuffin	Professeur, École de Technologie Supérieure, Canada	Rapporteur
Alexandru Telea	Professeur, Université d'Utrecht, Pays-Bas	Rapporteur
Marianne Huchard	Professeure, LIRMM - Université de Montpellier, France	Examinatrice
Christophe Hurter	Professeur, École Nationale d'Aviation Civile, France	Examinateur
Guy Melançon	Professeur, LaBRI - Université de Bordeaux, France	Examinateur
Pascal Poncelet	Professeur, LIRMM - Université de Montpellier, France	Examinateur
Gilles Venturini	Professeur, LIFAT - Université de Tours, France	Examinateur

Table des matières

Chapitre 1

Introduction

En 2011, je soutenais ma thèse de doctorat intitulée *Visualisation d'information : de la théorie sémiotique à des exemples pratiques basés sur la représentation de graphes et d'hypergraphes*. Depuis, que ce soit dans le cadre de mon postdoctorat à l'université de Californie à Davis ou de mon poste de Maître de Conférences à l'université Paul-Valéry et au LIRMM, je me présente comme un chercheur en visualisation d'information, ou éventuellement en "DataViz" quand il s'agit de toucher un public professionnel. Généralement, mes interlocuteurs montrent un certain intérêt pour la discipline, pensent à un jeu de données sur lequel ils travaillent et me demandent quel outil ils pourraient utiliser pour le visualiser ou pour montrer les résultats de leurs analyses. Montrer est ici le mot important. La plupart des personnes que j'ai rencontrées dans un cadre professionnel voient la visualisation comme un outil de communication permettant de présenter des données : ils recherchent le tableau de bord qui leur permettra de mettre en valeur leur analyse, et non la visualisation interactive complexe qui leur permettra de mener d'autres analyses. Cette dichotomie entre **outil de présentation** et **outil visuel d'analyse** est apparue très tôt. Or, comme le met en exergue van Wijk dans son article *Views on Visualization* [192], un grand nombre de chercheurs en visualisation s'intéressent, à tort ou à raison, principalement au deuxième type [1]. Par conséquent, alors qu'un grand nombre d'articles scientifiques porte sur la conception d'outils visuels d'analyse, une vaste bibliographie issue de divers domaines (art et *design* de l'information [2], journalisme graphique [3], statistique [4]) porte sur la présentation graphique, souvent nommée *infographie*. Ce phénomène est particulièrement bien illustré par le célèbre adage *a picture is worth ten thousand words* [5], qui s'oppose ici à la formule *using vision to think* [33]. Cette dernière insiste en effet sur la capacité qu'à la visualisation à nous permettre de mener des analyses et non à seulement nous montrer, de façon synthétique, des informations qui pourraient être expliquées textuellement [6].

Une autre caractéristique que j'ai pu observer au cours de ma carrière, et qui est sans doute liée à la remarque précédente, est que mes interlocuteurs ont souvent une vision très orientée ***design*** [7] de mon métier. Un outil visuel n'est pas forcément pensé en terme d'efficacité, de précision, de fidélité des représentations au contenu réel des données, *etc.* mais en terme d'esthétique et de communication.

1. Il existe cependant des exceptions, d'où l'adverbe "principalement", comme l'article de Segel et Heer [162] qui caractérise les visualisations dites "narratives" ou de *storytelling*.

2. On peut ici citer les célèbres livres d'Edward Tufte qui ont popularisé la représentation graphique de l'information [176, 175, 177]. Le livre de Manuel Lima [108], qui d'ailleurs contient des représentations issues du monde de la recherche en visualisation d'information, illustre aussi cette approche. Un autre exemple montrant la dimension artistique est l'exposition *Talk to Me* au musée d'art moderne de New-York [5].

3. Outre les infographies disponibles dans divers journaux d'information, les livres de David McCandless [120, 121] constituent une excellente entrée pour montrer la beauté et la diversité des productions.

4. La visualisation dite statistique, contrairement au *design* de l'information et au journalisme graphique, se caractérise par un soucis de fidélité aux données plus important. Il n'en demeure pas moins, dans la majorité des cas, un outil plus orienté "représentation" que "analyse", voir par exemple [193].

5. Voir [33] qui suggère l'origine de cette expression.

6. Tory et Möller [174] ont mis à jour un tableau initialement proposé par Card *et al.* [33] décrivant comment la visualisation peut être utilisée pour renforcer la cognition.

7. Dans ce manuscrit, le mot anglais *design* est utilisé lorsque l'esthétique est privilégiée par rapport à l'efficacité, il est traduit par conception quand c'est l'inverse. Il est cependant utile de remarquer que la frontière est floue dans la mesure où toute production essaie d'allier les deux dimensions, et que l'esthétisme d'une visualisation a une influence sur son efficacité, ne serait-ce qu'en la rendant plus attractive et donc plus utilisée.

Toujours en dialoguant avec un interlocuteur intéressé par la visualisation, j'essaie habituellement de le convaincre du potentiel qu'ont les applications du domaine à aider les chercheurs ou les professionnels à mener des analyses leur permettant de répondre, au moins partiellement, à leur problématique (par exemple en leur permettant de valider des hypothèses). Le plus rapide est de leur montrer des exemples d'outils et des exemples d'analyses qui ont pu être réalisés avec. La visualisation est ici abordé selon son rôle de **technologie** au service de la résolution de problèmes. Se pose alors une seconde difficulté : la simplicité. En effet, un outil est d'autant plus efficace et persuasif qu'il est simple à comprendre. L'interlocuteur peut alors avoir l'impression que cette simplicité d'appréhension correspond à une simplicité de conception, et ne nécessite donc pas l'intervention d'un chercheur spécialisé du domaine. Or, la production d'outils de visualisation nécessite de suivre une démarche scientifique rigoureuse, sur laquelle nous reviendrons plus tard dans ce manuscrit, et qui s'apparente à celle de tout domaine scientifique : définition du problème, conception de la solution, validation des résultats. Comme j'ai pu l'observer en tant que relecteur d'articles, et comme l'ont fait remarquer Sedlmair *et al.* [161], une erreur typique de novice consiste à ne pas connaître suffisamment le domaine et ainsi à échouer à produire un outil efficace. Sans une lecture approfondie de la littérature, il est en effet difficile d'analyser correctement le problème en terme de besoins visuels, de connaître l'étendu de l'espace de conception afin de choisir au mieux parmi les différentes alternatives et de connaître les différentes méthodes de validation. La connaissance des résultats de la **science** de la visualisation, *i.e.* les différents types de données, les différents types de tâches visuelles, les familles de visualisations, les familles de techniques d'interaction, les modèles de conception, les méthodes d'évaluation, *etc.*, constitue donc une étape obligatoire à la mise en place d'outils visuels d'analyse.

Design, technologie, science, je retombe ici sur les trois dimensions considérées dans l'article de van Wijk déjà cité [192]. Il n'en reste pas moins que cela ne suffit pas à définir ce que fait un chercheur du domaine. Dans ma thèse de doctorat, je définissais la visualisation d'information comme la "présentation visuelle d'un ensemble d'informations issues de données abstraites et traitées par des moyens informatiques". Au vu de ce qui a été dit plus haut, je rajouterais maintenant "en vu de résoudre des problèmes issus d'un domaine d'application". Compte tenu de tout cela, le spécialiste en visualisation est donc celui qui trouve les règles générales régissant les représentations visuelles, applique ces règles pour créer des solutions visuelles à des problématiques identifiées, trouve les techniques et les algorithmes permettant de mettre en œuvre ces solutions et valide l'ensemble (règles, solutions, techniques, algorithmes). Trois questions sont alors fondamentales dans le processus de production : "Quoi ?", "Pourquoi ?" et "Comment ?" [8]

1.1 Quoi ? Pourquoi ? Comment ?

Dans cette section, je vais présenter les grandes familles d'éléments de réponse aux trois questions auxquelles un concepteur doit répondre afin de proposer une visualisation adaptée. Ces familles se basent sur le livre de Munzner [133] qui propose une vue d'ensemble de l'état actuel des connaissances sur l'espace de conception de la visualisation.

1.1.1 Quoi : abstraction des données

La première question à se poser est de savoir quelles sont les données et les structures associées qui seront visualisées / manipulées / explorées grâce à l'outil. Cette étape, aussi appelée "abstraction des données", consiste à identifier et surtout structurer les données brutes issues

8. Il existe d'autres angles pour introduire le domaine de la visualisation d'information. J'ai, par exemple, proposé un court historique montrant les différentes étapes qui ont permis de passer de l'écriture aux interfaces complexes actuelles [156]. Une rapide introduction au domaine a été proposée par Mazza [119]. Le livre de Bertin [19], en créant le domaine de la sémiologie graphique, constitue un travail précurseur de référence et d'inspiration. Le livre de Card *et al.* [33] regroupe une série d'articles précurseurs ainsi qu'une discussion très intéressante sur le domaine. Le livre désormais classique de Ware [185] étudie le domaine à travers l'angle de la perception visuelle. Enfin, les livres de Ward *et al.* [184] et de Telea [171] suivent une approche plus technique et algorithmique que ce que je vais présenter et constituent donc des outils complémentaires et incontournables pour se familiariser à la réalisation d'outils visuels.

d'un domaine d'application de façon à pouvoir leur appliquer des techniques de visualisation. Cette étape n'est cependant pas anodine, car les choix qui y seront fait vont bien entendu contraindre l'espace de conception, *i.e.* l'ensemble de types d'encodages visuel et des interactions disponibles.

Prenons par exemple d'un ensemble de pages *Web* avec leur mots-clés. Il est possible de le structurer sous la forme d'une table dont les lignes correspondent aux mots-clés et les colonnes aux pages *Web*. Les cellules contiennent alors une valeur correspondant au nombre de fois où un mot-clé apparaît dans une page. Il est cependant aussi possible de dériver de ce même ensemble un graphe de cooccurrence de mots-clés : chaque sommet du graphe représente alors un mot-clé et il existe un lien de poids p entre deux mots-clés ci ceux-ci apparaissent p fois ensembles dans les différentes pages [158]. Selon la structure choisie, table ou graphe, les techniques de représentations disponibles ne seront bien entendu pas les mêmes.

La figure 1.1 donne un aperçu des éléments disponibles pour structurer un jeu de données brutes. Donner tous les détails de ces éléments dépasse largement le cadre de ce manuscrit. Notons cependant quelques points. Il est convenu que les principaux types de jeux de données peuvent se ranger en quatre grandes catégories : **tables** (ensemble d'items avec des attributs), **réseaux**[9] (ensemble de nœuds et de liens), **champs** (variable continue dans le plan ou l'espace) ou **géométries** (ensemble de points, lignes, surfaces, et parfois même, volumes géolocalisés). Un second point important est le type des attributs. Contrairement à ce que l'on trouve dans d'autres domaines comme la statistique, les types d'attributs ne sont pas classés selon la dichotomie quantitatif / qualitatif (ordinal + catégoriel). Ici, on privilégie la dichotomie ordonné / catégoriel (quantitatif + ordinal), car elle correspond aux règles d'emploi de l'apparence des objets graphiques que nous verrons dans la section 1.1.2.

Outre le lien étroit qui existe entre l'abstraction des données et les techniques de visualisation disponibles, l'abstraction des données est aussi fortement liée à l'objectif de la visualisation, *i.e.* la réponse à la question "Pourquoi ?". En effet, selon les informations que l'on veut extraire, les structures à privilégier ne sont pas nécessairement les mêmes. Si je reprends l'exemple précédent, la structure graphe permet par exemple de répondre à la question "Quelle est la proximité entre tel mot-clé et tel mot-clé ?" alors que la structure table permet par exemple de répondre à la question "Quels sont les mot-clés contenus dans telle page ?".

1.1.2 Pourquoi : abstraction des tâches

L'objectif ici est de lister les tâches qui devront être réalisées à l'aide de la visualisation. Comme l'ont souligné Meyer *et al.* [126], la question "Pourquoi ?" relève de l'**identification** et non de la **conception**, contrairement aux questions "Quoi ?" et "Comment ?". En effet, pour répondre à ces deux dernières, des choix doivent être faits. Au contraire, le but ici est de "traduire" les questions que se posent les experts du domaine d'application en tâches réalisables sur les structures des données proposées en réponse à la question précédente.

La figure 1.2 montre un aperçu des types de tâches qui peuvent être identifiées. Comme on peut le constater, les tâches sont exprimées de façon abstraite, *i.e.* elles concernent les structures et non la sémantique des données, d'où la difficulté de cette étape. Cet aspect est primordial : il permet à la fois de généraliser les résultats de façon à ce qu'ils puissent s'appliquer à d'autres domaines mais également à aider à identifier les ressources bibliographiques qui pourraient venir en aide au concepteur.

Dans la figure 1.2, on peut observer que les tâches se définissent en terme d'**action** et de **cible**. Par exemple, on peut vouloir *découvrir* un *chemin* dans un réseau. De nombreuses autres taxonomie de tâches ont été proposées, le lecteur intéressé pourra se référer à l'article [28]. Il existe aussi des taxonomies plus détaillées selon les types de données sur lesquelles les tâches s'appliquent. Citons par exemple Lee *et al.* [105] pour les réseaux, Roth [151] pour les géométries ou Lammarsch *et al.* [104] pour les données temporelles.

9. Dans ce manuscrit, les mots *graphes* et *réseaux* sont interchangeables, de même que les mots *nœuds*/*sommets* et les mots *liens*/*arêtes*.

FIGURE 1.1 – Taxonomie pour l'abstraction des données [133]

1.1.3 Comment : conception des encodages visuels et des interactions

Les deux questions précédentes permettent de définir ce que l'on veut faire et sur quoi on veut le faire. En d'autres termes, elles permettent de poser le problème issu du domaine d'application en problème visuel. Il reste maintenant à trouver une solution à ce problème, donc de répondre à la question "Comment ?". En suivant Munzner [133], nous allons définir un **idiome visuel** ("idiome" dans la suite du manuscrit) comme un ensemble d'approches permettant de créer une représentation graphique ainsi que les interactions qui permettent de la manipuler. Il existe de nombreuses approches permettant d'encoder visuellement un ensemble de données. De plus, les différentes techniques d'interaction possibles rendent encore plus important la taille de l'espace de conception. La figure 1.3 en donne un aperçu très simplifié.

La partie encodage comprend les approches possibles pour la représentation des données,

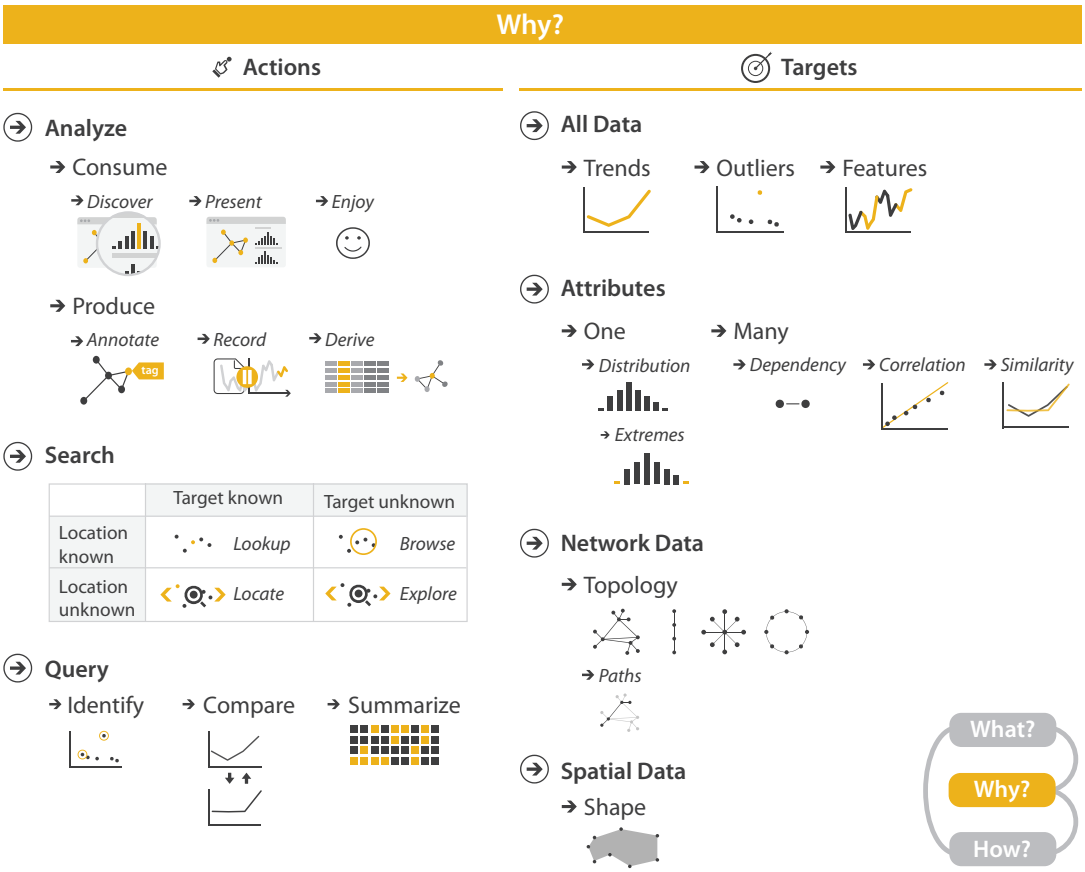


FIGURE 1.2 – Taxonomie pour l'abstraction des tâches [133]

alors que les autres parties sont directement liées à l'interaction. En suivant le vocabulaire issu de la sémiologie graphique de Bertin [19], encoder visuellement un ensemble de données consiste d'abord à définir un ensemble d'objets graphiques représentant des items des données. Ces objets ont un **type d'implantation**, *i.e.* ils peuvent être des points, des lignes, des surfaces ou des zones. Si je reprends mon exemple de graphe de cooccurrence de mots-clés dans des pages *Web*, les nœuds peuvent être représentés à l'aide de points et les liens à l'aide de lignes. Ces objets sont ensuite **positionnés** dans le plan ou l'espace. Je peux par exemple utiliser un algorithme de type masse-ressort [26] pour afficher le graphe de cooccurrence. Enfin, une apparence leur est attribuée en fonction des valeurs des attributs des items. Par exemple, la taille des nœuds du graphe de cooccurrence peut dépendre du nombre de pages dans lequel le mot-clé se trouve. Les dimensions de l'apparence (taille, orientation, luminosité, saturation, teinte, courbure, forme, mouvement...) sont appelées **variables visuelles**. Elles se classent généralement en deux catégories : celles qui représentent efficacement les attributs ordonnés et celles qui représentent efficacement les attributs catégoriels (d'où la dichotomie mentionnée plus haut). Par exemple, la forme des nœuds du graphe de cooccurrence sera inappropriée pour représenter le nombre de pages, car dire qu'un triangle représente un chiffre plus élevé qu'un carré n'a pas de sens. Au contraire, la taille est parfaitement adaptée.

Si l'on s'en tient aux encodages visuels, les idiomes prennent la forme de cartes statiques. Or, selon la taille ou la complexité des données, ou selon les tâches d'analyse à effectuer dessus, il est souvent nécessaire d'ajouter des moyens d'interagir avec les représentations. Les techniques les plus élémentaires et que l'on retrouve dans quasiment toutes les interfaces, permettent de **manipuler** une représentation : ce sont la modification, la sélection, la navigation (défilement, zoom...). Viennent ensuite les techniques permettant de **combiner** plusieurs représentations : juxtaposition, partition, superposition. Enfin, les techniques de **réduction** permettent de filtrer ou d'agréger les données, voir de combiner les deux (focus+contexte).

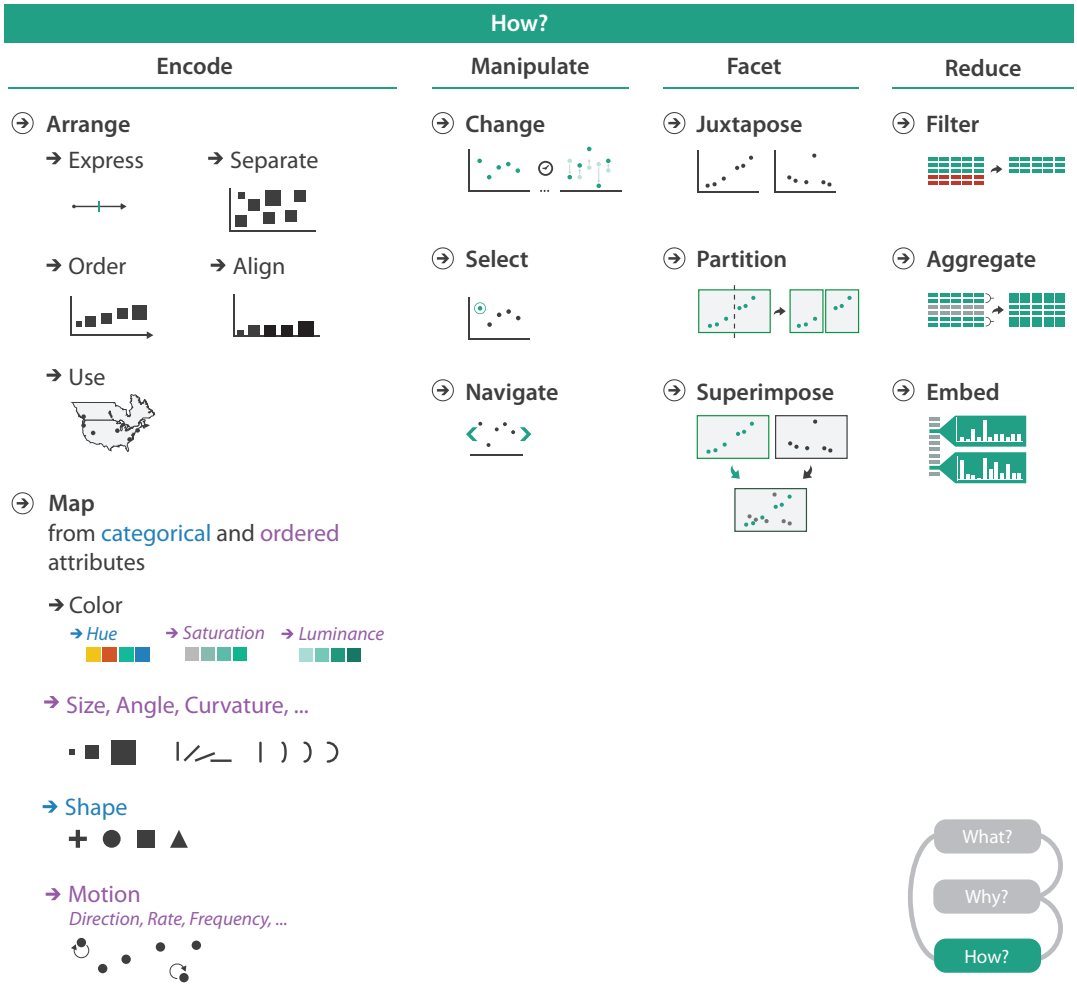


FIGURE 1.3 – Espace de conception des idiomes visuels [133]

1.1.4 Synthèse

Créer une visualisation consiste donc à répondre à ces trois questions : Quelles sont les données à traiter ? Quelles sont les tâches à accomplir dessus ? Comment visualiser et interagir avec des données pour accomplir ces tâches ? Il est important de rappeler ici que ces questions ne sont pas indépendantes. Comme le montre la figure 1.4, le choix des abstractions des données doit être guidé d'une part par l'identification des tâches et d'autre part par les idiomes que ces structures permettent de mettre en place. Comme mentionné en introduction, une bonne connaissance de la littérature est donc nécessaire à la mise en place d'outils visuels adaptés.

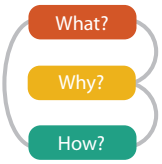


FIGURE 1.4 – Interdépendance des questions Quoi ? Pourquoi ? Comment ? [133]

Dans une démarche scientifique, chacun de ces aspects doit être validé à l'aide, par exemple, d'expérimentations contrôlées ou d'études de cas [142]. Je vais maintenant présenter un modèle qui traite de ce sujet.

1.2 Modèle imbriqué

Le modèle imbriqué pour la conception et la validation d'outils visuels, proposé par Munzner [131], a eu un fort impact sur la communauté. Il est courant maintenant (j'en ai d'ailleurs fait les frais), qu'un évaluateur demande une révision majeure pour un article afin d'y inclure le positionnement du travail effectué, aussi bien en terme de conception que d'évaluation, selon les axes du modèle. Comme le montre la figure 1.5, ce modèle est composé de quatre niveaux imbriqués que je vais maintenant détailler.

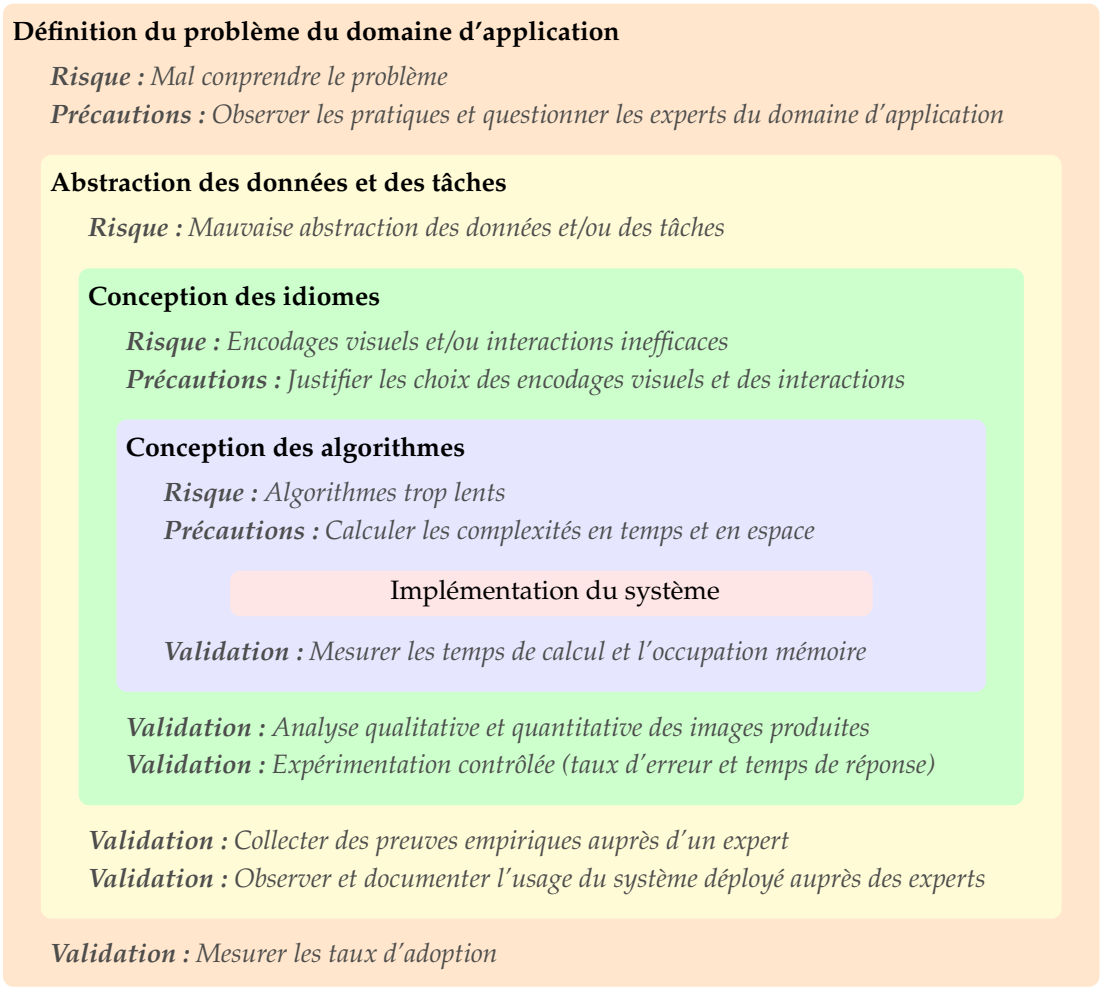


FIGURE 1.5 – Modèle imbriqué pour la conception et la validation d'outils visuels [131]

1.2.1 Définition du problème du domaine d'application

Le premier niveau consiste à comprendre les données et les objectifs du domaine d'application afin d'en déduire une liste de questions que se posent généralement les experts de ce domaine. Ces questions doivent être de bas niveau et servent ensuite d'entrée pour le niveau suivant du modèle. La difficulté réside ici dans le besoin d'acquérir des connaissances suffisantes du domaine d'application pour pouvoir comprendre les questions, tout en se restreignant au strict minimum afin de rendre l'opération le moins chronophage possible. Le concepteur ne doit pas devenir un expert du domaine d'application, il doit juste se contenter d'en connaître certains mécanismes afin de comprendre les problématique sous-jacentes.

Le principal risque de ce niveau est de ne pas comprendre la problématique du domaine. Afin de s'en prémunir, le concepteur doit **questionner les experts** et, si possible, **observer leurs**

pratiques. La validation après déploiement se fait grâce à **l'observation du taux d'utilisation de l'outil** au sein de l'audience cible.

1.2.2 Abstraction des données et des tâches

À partir des questions identifiées dans le niveau précédent, l'objectif de ce niveau est de choisir les abstractions des données et d'identifier les abstractions des tâches (voir les sections 1.1.1 et 1.1.2).

Le principal risque ici est de choisir les mauvaises structures de données ou de mal identifier les tâches à accomplir par un expert. La validation ne peut donc être réalisée que par un expert. Le plus souvent, il lui est demandé **d'utiliser l'outil et de voir s'il arrive à retrouver des connaissances qu'il avait déjà, à confirmer des hypothèses, à en formuler de nouvelles**, *etc.* Une autre méthode de validation, plus rigoureuse, consiste à **observer et documenter la façon dont les experts utilisent le système** après déploiement. Cela passe par une longue étude sur la façon dont le système s'intègre dans le processus de travail des experts. Dans la mesure où il est très difficile à mettre en place et très chronophage, ce type d'étude est très rare.

1.2.3 Conception des idiomes

À partir des abstractions des données et des tâches, l'objectif de ce niveau est de proposer des techniques d'encodage et d'interaction. Le tout permet de concevoir un ou plusieurs idiomes (voir la section 1.1.3). Ils doivent permettre d'accomplir les tâches identifiées sur les données.

Le principal risque est de produire des idiomes ne permettant pas de réaliser les tâches identifiées. Pour s'en prémunir, le concepteur doit pouvoir **justifier ses choix** en se basant sur l'abondante littérature concernant les principes de la perception visuelle (voir par exemple [185]) et les exemples d'idiomes ayant prouvé leur efficacité.

La validation de ce niveau ne nécessite pas forcément l'intervention d'un expert : les données et les tâches étant exprimées de façon abstraite, la connaissance du domaine d'application n'est pas nécessaire. La façon la plus efficace pour valider un idiome, ou pour choisir le plus efficace parmi plusieurs d'entre eux, est de réaliser une **expérimentation contrôlée**, durant laquelle un ensemble de personnes doit accomplir les tâches du niveau précédent sur différents jeux de données en utilisant une implémentation des idiomes. Les taux d'erreurs et les temps de réponse sont mesurés et une analyse statistique permet de valider/invalider et/ou comparer les approches. Ce type de validation est très efficace mais souvent difficile à mettre en place, surtout lorsque la liste des tâches du niveau précédent est étendue et le nombre d'idiomes important (dans le cadre d'un outil visuel multi-vues par exemple). Dans ce cas, une méthode plus souple de validation basée sur une **discussion qualitative des images ou des vidéos** issues des idiomes est privilégiée. L'objectif ici est de mettre en évidence que les résultats obtenus présentent bien les propriétés attendues. Des **études de cas** illustrent la capacité de l'outil à être utile pour accomplir les tâches identifiées sur le(s) jeu(x) de données. Pour finir, il peut s'avérer parfois utile de **valider de manière quantitative** les images produites. Il s'agit ici de définir des indices de qualité et de les mesurer sur ces images. Un exemple typique et bien étudié concerne les critères esthétiques mesurables pour la représentation de graphes [186].

Il est parfois utile de réaliser des **tests d'utilisabilité** avant d'appliquer les méthodes de validation évoquées ci-dessus. Ils ne permettent pas de montrer l'adéquation d'un idiome avec les tâches et les données, ou de montrer la supériorité d'une approche sur une autre, mais il est clair qu'un outil présentant un défaut d'utilisabilité ne permettra pas de remplir ces fonctions.

1.2.4 Conception des algorithmes

L'objectif de ce niveau est de concevoir les algorithmes qui permettent de mettre en œuvre les idiomes issus du niveau supérieur. Le principal risque est de concevoir des algorithmes trop lents (par exemple pour assurer l'interactivité du système) ou trop gourmands en espace. Un moyen de s'en prémunir est de **calculer les complexités théoriques**. La validation se fait ensuite par **observation des temps de calcul et de l'occupation mémoire**.

1.2.5 Synthèse

Le modèle imbriqué est souvent vu comme un modèle décrivant les différentes étapes à suivre pour produire une visualisation. Cependant, comme nous allons le voir en introduction du chapitre 3, concevoir une visualisation adaptée à un domaine ne se limite pas à suivre ces étapes. Le principal avantage de ce modèle est de fournir un outil permettant de prendre conscience des 4 niveaux conceptuels de la visualisation, des liens qu'ils ont entre eux et des types de validation qu'ils induisent. Il permet donc de caractériser les différentes contributions d'un travail (qui peut couvrir, ou non, les 4 niveaux) et de sélectionner le(s) type(s) d'étude(s) qui permettront des le(s) valider.

Ce modèle est imbriqué dans la mesure où il faut avoir les résultats (*output*) d'un niveau $N-1$ pour réaliser un niveau N. Il faut, par exemple, définir les idiomes avant de concevoir les algorithmes qui permettent de les mettre en œuvre. De plus, l'invalidation d'un niveau N entraîne l'invalidation des niveaux suivants. Par exemple, si les abstractions des données et des tâches à accomplir issues du niveau 2 ne permettent pas de résoudre la problématique soulevée dans le niveau 1, alors il faut non seulement les redéfinir, mais aussi re-concevoir les idiomes et les algorithmes qui en découlent.

Comme l'a souligné Beaudouin-Lafon [15], un modèle est d'autant plus puissant qu'il permet à la fois de décrire, évaluer et produire des outils. Le modèle imbriqué remplit ces trois objectifs. Il a cependant été raffiné depuis en intégrant deux nouveaux aspects : un découpage des niveaux en blocs et des lignes directrices entre ces blocs pour guider les choix de conception. Je ne le décrirai pas ici car le modèle initial est suffisant pour mon discours, mais j'invite le lecteur intéressé à lire l'article correspondant [126].

Ce manuscrit présente des travaux significatifs que j'ai mené et qui sont abordés selon le prisme de ce modèle. Il existe différentes approches permettant d'organiser des travaux scientifiques liés à la visualisation d'information. Suite à ce qui vient d'être dit, on peut les regrouper par abstraction des données, abstraction des tâches ou idiomes utilisés. On peut aussi les regrouper selon les domaines d'application (santé, environnement, humanités...). Dans ce manuscrit, j'ai choisi d'utiliser une approche transversale à celles-ci, qui consiste à regrouper mes recherches selon le type de contribution qu'elles ont apporté. Nous allons donc maintenant voir quels sont ces types dans le domaine de la visualisation.

1.3 Types de contribution

Lorsque l'on soumet un article à InfoVis [10], VAST [11], EuroVis [12] ou bien encore PacificVis [13], il est demandé de spécifier un "type d'article". Ces cinq types ont été introduits par Munzner et North pour InfoVis 2003, alors qu'ils étaient présidents de la conférence. Ils se sont rapidement répandus, notamment grâce à l'article de Munzner publié cinq ans plus tard [132] qui les détaille, et qui discute de façon à la fois précise et ludique comment repérer et mettre en valeur les contributions d'un travail.

Les cinq types de contribution, que je décris ci-dessous, donnent un aperçu des travaux réalisés par les chercheurs en visualisation d'informations. Bien entendu, une grande partie de ces travaux ne tombe pas exactement dans une seule de ces catégories, le chercheur choisit alors celle qui est la plus caractéristique de son travail.

1.3.1 Technique de visualisation

Dans la littérature anglophone, on trouve cette catégorie sous le terme *technique* ou les termes *technique & algorithm*. Concrètement elle regroupe les contributions portant sur de nouveaux

10. http://ieeevis.org/year/2020/info/call-participation/infovis-paper-types
11. http://ieeevis.org/year/2020/info/call-participation/vast-paper-types
12. https://conferences.eg.org/egev20/for-submitters/eurovis-for-submitters/eurovis-full-papers/
13. http://vis.tju.edu.cn/pvis2020/full_paper.html

idiomes visuels, ou sur des algorithmes permettant de réaliser ces idiomes. Il convient de noter ici que dans le premier cas, un ou des algorithme(s) sont souvent aussi proposés pour réaliser l'idiome. Cependant, le but étant de convaincre de son utilité, les évaluations tendent principalement à démontrer que ce nouvel idiome permet de réaliser des nouvelles tâches sur un certain type de données, ou permet de réaliser des tâches de façon plus efficace qu'en utilisant les techniques concurrentes (voir la section 1.2.3). Lorsque ces contributions sont exclusivement algorithmiques, elles portent sur un ou plusieurs nouveaux algorithmes. Dans ce cas, les évaluations se font en terme de temps de calcul et d'occupation mémoire (voir la section 1.2.4).

1.3.2 Conception de solutions visuelles

On trouve cette catégorie sous le nom de *design study* ou *application* en anglais. Elle regroupe les contributions portant sur la création de nouvelles solutions visuelles permettant de résoudre un problème réel issu d'un domaine d'application particulier. L'objectif ici est de décrire le domaine d'application (voir la section 1.2.1) afin de définir une liste de besoins en terme de tâches à accomplir, puis de concevoir une solution visuelle permettant d'accomplir ces tâches. La validation consiste principalement à convaincre que la solution proposée permet bien d'accomplir les tâches définies (voir la section 1.2.3), et que ces tâches correspondent bien au besoin des utilisateurs (voir la section 1.2.2). La proposition de nouvelles techniques de visualisation ou d'algorithmes reste marginale dans ce type de contribution.

1.3.3 Système de visualisation

La catégorie *system* ou *toolkit* en anglais regroupe les contributions portant la création de logiciels, boîtes à outils ou librairies de visualisation. Typiquement, ce type de contribution n'introduit pas de nouvelles techniques ou de nouveaux algorithmes. Les contributions se situent principalement autour des choix, en terme de visualisation et d'architecture, qui permettent d'obtenir un outil cohérent et efficace vis-à-vis du problème initial. On peut voir cette catégorie comme une sous-catégorie de la précédente, dans laquelle on s'interroge plus sur les choix permettant de produire un logiciel ou une librairie que sur les choix d'idiomes permettant de résoudre un problème donné.

Le modèle imbriqué nous éclaire sur la principale distinction que l'on peut faire entre les contributions *technique de visualisation* d'une part et les contributions *conception de solutions visuelles* et *système de visualisation* d'autre part. En effet, les premières portent essentiellement sur les deux dernières couches du modèle (conception des idiomes et des algorithmes) alors que les secondes portent essentiellement sur les trois premières (définition du problème, abstraction des données et des tâches, conception des idiomes). Les travaux de la première catégorie sont ainsi guidés par la technique explorée (*technique-driven work*) alors que ceux de la seconde sont guidés par la problématique des utilisateurs du domaine d'application (*problem-driven work*).

1.3.4 Étude empirique

Dans la littérature anglophone, cette catégorie se trouve sous les noms de *empirical sudy*, *evaluation*, *summative user study* ou *evaluation & empirical research*. Elle regroupe les contributions portant sur l'utilisation de techniques ou de systèmes de visualisation, principalement évaluée à l'aide d'expérimentations contrôlées. Ces contributions se positionnent donc essentiellement dans la troisième couche du modèle, *i.e.* la conception des idiomes (voir la section 1.2.3). Cependant, comme nous le verrons dans la section 4.3, les études empiriques à proprement parler ne se limitent pas à l'évaluation des performances utilisateur à l'aide d'expérimentations contrôlées. Lorsqu'elles prennent d'autres formes, comme les différents types de validation évoquées ci-dessus, elles ne constituent généralement pas la contribution principale du travail qui se classe alors dans une des autres catégories ci-dessus.

1.3.5 Théorie et modèle

La catégorie *theory & model* regroupe les contributions générales dont le but est d'aider les chercheurs à réaliser leurs travaux. On peut la diviser en trois sous catégories. La première est nommé **taxonomie** (*taxonomy*), elle porte sur la création de catégorisations des approches visuelles, des tâches ou des jeux de données, permettant de les classer dans des ensembles cohérents afin d'aider la conception d'outils visuels. La deuxième sous-catégorie, **formalisme** (*formalism*) regroupe les articles présentant de nouveaux modèles (de conception ou d'évaluation par exemple), définitions ou terminologies pour décrire des techniques ou des phénomènes. La dernière sous-catégorie, **discussion** (*commentary*), contribue à la visualisation en prenant position pour une idée, une pratique ou autre, et en argumentant dessus. De façon transversale à ses trois sous-catégories initialement proposées, une contribution *théorie et modèles* peut aussi porter sur la théorie de la perception et de la cognition dans la mesure où cette théorie a des applications en visualisation.

Le modèle imbriqué est une contribution *théorie et modèle*, on ne peut donc pas toujours situer les contributions de ce type dans l'un de ses niveaux (contrairement aux autres types). Cependant, l'exemple des taxonomies de tâches évoqué au début de ce chapitre nous montre que certaines théories ou certains modèles peuvent contribuer à mieux appréhender certains niveaux du modèle imbriqué.

1.4 Organisation du manuscrit

Au regard de ce bref aperçu du domaine de la visualisation, je vais maintenant pouvoir positionner mes principales contributions ainsi que mes futures orientations. Ces contributions se placent dans les deux premières catégories.

Je traiterai d'abord de travaux introduisant des *techniques de visualisation* pour l'exploration de données temporelles et relationnelles dans le chapitre 2. Ensuite, je présenterai des *conceptions de solutions visuelles* pour l'environnement et la santé dans le chapitre 3. Dans le chapitre 4, je conclurai et je présenterai les axes que je prévois d'explorer. Les deux premiers se situent dans la continuité des mes précédents travaux, *technique de visualisation* et *conception de solutions visuelles*, le troisième porte sur *les études empiriques*.

Chapitre 2

Techniques de visualisation de données temporelles et relationnelles

Comme je l'ai déjà mentionné en introduction, une contribution technique porte sur la proposition de nouveaux idiomes et/ou de nouveaux algorithmes. Dans cette section, je vais présenter quelques uns de mes travaux qui s'inscrivent dans ce cadre, en commençant par ceux plus orientés "idiomes" et en dérivant vers ceux plus orientés "algorithmes". Le modèle imbriqué nous a appris qu'une contribution technique part d'une problématique exprimée sous la forme d'abstractions de données et de tâches (*output* du niveau 2, voir la section 1.2.2). Concernant les données, nous commencerons par un exemple d'idiome pour la visualisation de séries temporelles (section 2.1). Puis, toujours dans le cadre de données temporelles, nous nous intéresserons à la visualisation interactives de grands graphes dynamiques (section 2.2). La structuration en graphe nous amènera ensuite à nous intéresser à des problèmes algorithmiques portant sur la lisibilité des représentations, et en particulier, sur la suppression de chevauchements de sommets (section 2.3).

2.1 *MultiStream* : exploration de séries temporelles

Le travail présenté ici fait suite à un premier article [182] dans lequel nous proposions *SentiCompass* [1], une visualisation permettant d'observer l'évolution de sentiments exprimés dans des *tweets*. Nous y présentions une procédure d'abstraction des données permettant d'extraire les sentiments des *tweets* selon les deux axes du modèle de Russell [153], puis un idiome circulaire compatible avec ce modèle (voir la figure 2.1).

Au niveau abstraction des données, *SentiCompass* traitait donc de séries temporelles (nombre de *tweets* évoluant dans le temps pour chaque sentiment) qui pouvaient être classées dans les quatre classes issues des deux dimensions du modèle de Russell. Cette première expérience nous a amené à nous intéresser de façon plus générale aux séries temporelles (*i.e.* ensemble de variables quantitatives évoluant au cours du temps) hiérarchisées (chaque série appartient à une classe, les classes pouvant elles-mêmes appartenir à d'autres classes, *etc.*). À titre d'exemple, le jeu de données qui sera utilisée pour illustrer mon propos se compose d'une classification de genres musicaux ainsi que, pour chacun des 32 genres du dernier niveau, du nombre de groupes formés chaque année. Au total, 10642 groupes formés entre 1960 et 2016 ont été considérés. Les données ont été extraites de *MusicBrainz* [2]

Concernant l'abstraction des tâches, nous avons identifié une liste de tâches caractéristiques de séries temporelles et de hiérarchies que nous avons classé en 3 catégories de besoins (*requirements*) pour l'idiome :

1. http://youtu.be/ZaMF6VNO7tA [dernière consultation le 17/03/2020]
2. *MusicBrainz* est une encyclopédie ouverte dédiée à la musique https://musicbrainz.org/ [dernière consultation le 17/03/2020]

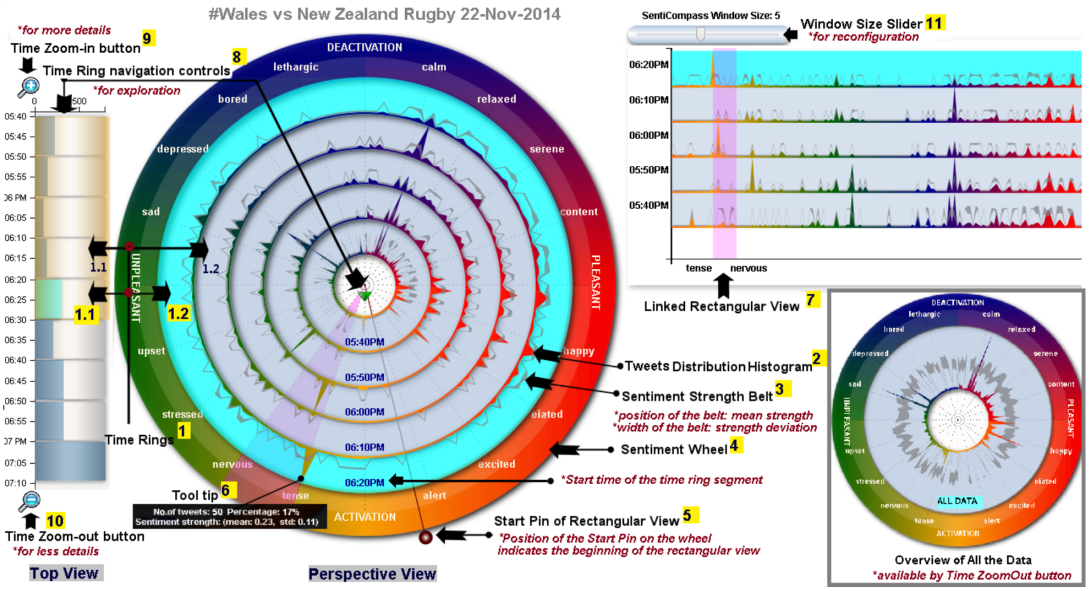


FIGURE 2.1 – *SentiCompass* [182]

[B1] Visualiser les motifs temporels généraux du jeu de données : principales tendances, pics, *etc.*

[B2] Sélectionner une période de temps pour visualiser en détail les motifs de cette période, tout en gardant le contexte.

[B3] Naviguer dans la hiérarchie pour visualiser les motifs temporels à différents niveaux d'agrégation.

[B4] Masquer des séries afin de visualiser les motifs temporels des autres séries plus en détail.

Après analyse de la littérature portant sur les données temporelles [3], il s'est avéré qu'il n'existait pas de méthode répondant à ces besoins. C'est pourquoi nous avons proposé *MultiStream* [4] [42], une nouvelle visualisation. Elle est basée sur des diagrammes en couches empilées (*streamgraphs*) dans lesquels le temps est représenté par l'axe horizontal et les séries par des couches colorées superposées dont l'épaisseur varie en fonction des valeurs [5] (voir la figure 2.2).

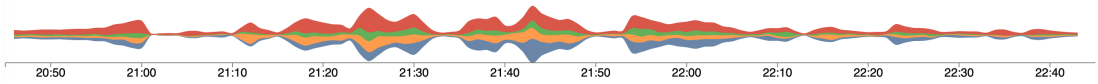


FIGURE 2.2 – Exemple de diagrammes en couches empilées [42]

Quelques travaux précédant le notre et aussi basés sur des diagrammes en couches empilées ont déjà été proposés pour visualiser des séries temporelles hiérarchisées [18, 188, 181, 14, 68, 47, 44]. Notre approche est cependant la seule à proposer un ensemble d'encodages visuels et d'interactions permettant d'explorer les séries à différents degrés de détail tout en préservant le contexte global de la visualisation.

3. La littérature portant sur les données temporelles est vaste. Le livre d'Aigner *et al.* [3] constitue une excellente entrée pour se familiariser avec les problèmes et pour avoir une vision des principales approches proposées.

4. Une démo et une vidéo sont disponibles en ligne : http://advanse.lirmm.fr/multistream/ [dernière consultation le 17/03/2020].

5. Les exemples de visualisations basées sur des diagrammes en couches empilées sont nombreux, citons par exemple [76, 111, 78, 79, 187, 32, 45, 194, 168, 183]. Pour une version plus détaillée des différentes approches liées à *MultiStream*, j'invite le lecteur à lire la section *Related Work* de notre article [42].

2.1.1 Conception des idiomes

La figure 2.3 montre une vue d'ensemble de *MultiStream*. Notre idiome est basé sur une approche multi-vues juxtaposées (*Facet* -> *Juxtapose* dans le résumé de l'espace de conception schématisé de la figure 1.3).

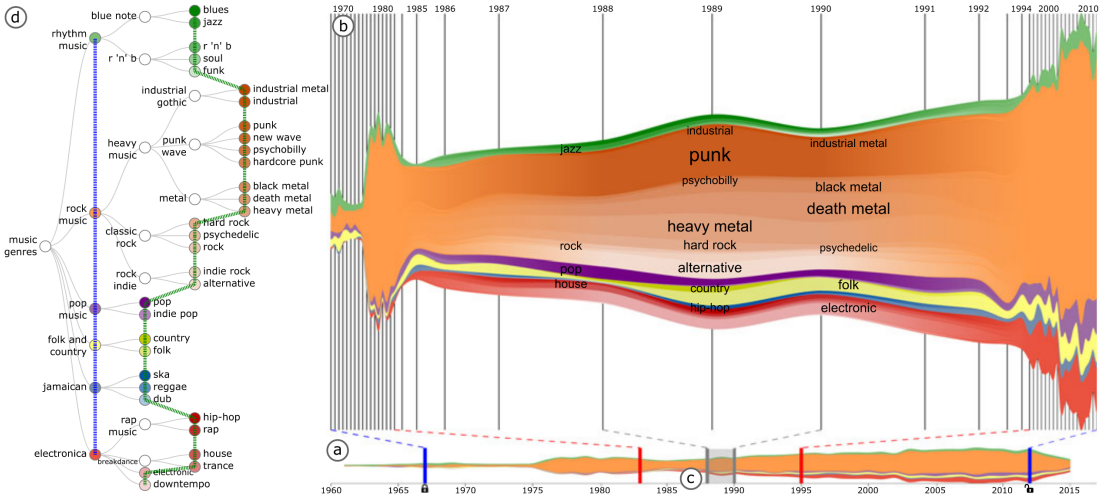


FIGURE 2.3 – *MultiStream* : (a) Une vue d'ensemble montre les séries temporelles à un haut niveau d'abstraction ; (b) Une vue multi-résolution montre les séries temporelles à différents niveaux d'abstraction ; (c) Un contrôleur permet de lier la vue d'ensemble avec la vue multi-résolution ; (d) Une vue de la hiérarchie montre la classification des séries et permet de filtrer et contrôler les niveaux d'abstraction des autres vues [42].

La vue d'ensemble (figure 2.3(a)) montre les séries à un haut niveau d'abstraction, *i.e.* les séries sont agrégées de façon à ce que les couches correspondent à un haut niveau de la hiérarchie. Cette vue permet donc de satisfaire le besoin [B1].

La vue multirésolution (figure 2.3(b)) montre les séries sur un intervalle de temps donné [B2]. Elle est basée sur une technique dite *focus + contexte* [40] (*Reduce* -> *Embed* dans le résumé de l'espace de conception schématisé de la figure 1.3). Le principe est d'intégrer, à l'intérieur d'une même vue, une ou plusieurs zone(s) montrant des données détaillées (*focus*) et une ou plusieurs zone(s) montrant des données à un plus haut degré d'abstraction (*contexte*). Une solution pour mettre en place une telle approche consiste à utiliser une distorsion du plan. Elle est connue sous le nom de *fisheye* [63, 91, 62]. La figure 2.4 illustre notre approche. Les barres verticales représentent des pas de temps uniformes (*i.e.* chacun d'entre eux représente la même durée). Comme on peut l'observer, les longueurs en x des pas de temps de la zone c (notre *focus*) sont plus importantes que les longueurs des pas de temps des zones a (*contextes*). Les zones b sont des zones de transition où les pas de temps passent progressivement des longueurs de a aux longueurs de c. On obtient ainsi une vue dans laquelle l'espace dédié au *focus* étant plus important, on peut y afficher un niveau de détail plus élevé : alors que tous les groupes de *jazz* sont regroupés dans les zones a, ils sont représentés dans leurs trois sous-catégories (*soul jazz*, *contemporary jazz* et *classical jazz*) dans la zone c. Une interpolation de couleurs dans les zones b permet d'assurer la fluidité de la transition. Il est à noter ici que nous n'appliquons pas de distorsion sur l'axe des y afin de ne pas altérer la lisibilité des valeurs qui sont représentées par la hauteur des couches.

Le contrôleur (figure 2.3(c)) permet de sélectionner les intervalles de temps affichés dans les différentes zones de la vue multirésolution. Comme l'illustre la figure 2.5, les barres verticales bleues peuvent être déplacées pour sélectionner l'intervalle total de temps affiché [B2]. Les barres rouges permettent quant à elles de sélectionner les intervalles affichés dans les zones

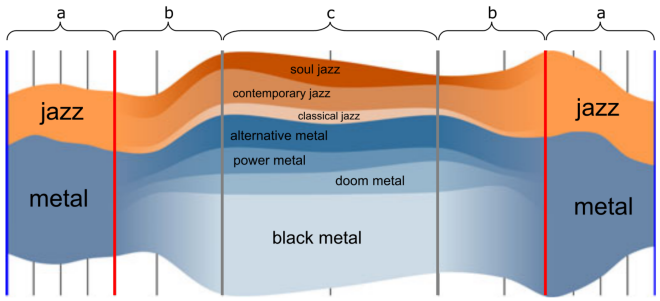


FIGURE 2.4 – Vue multirésolution : (a) zones de *contexte* montrant les séries avec un niveau d'abstraction élevé, (b) zones de transition, (c) zone de *focus* montrant les séries avec un niveau de détail élevé.

de transition et les barres grises l'intervalle affiché dans la zone du *focus*. De plus, l'utilisateur peut déplacer la zone grise dans le cadre fixé par les barres bleues. Dans ce cas, la zone du *focus* et les zones de transitions sont déplacées, l'intervalle total n'est en revanche pas modifié.

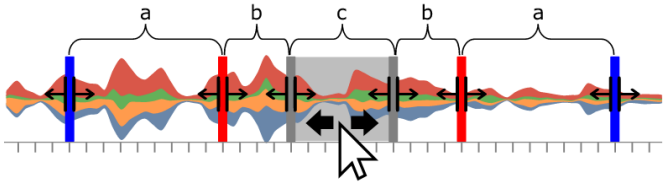


FIGURE 2.5 – Le contrôleur, placé sur la vue d'ensemble permet de sélectionner l'intervalle de temps affiché dans la vue multirésolution, ainsi que les zones de *contexte*, de transition et de *focus* de cette dernière.

La vue de la hiérarchie (figure 2.3(d)) montre la classification hiérarchique des séries temporelles. Elle permet aussi de sélectionner les niveaux d'agrégation affichés dans les autres vues [B3]. Une ligne en pointillés bleue montre les classes affichées dans la vue d'ensemble et dans les zones de *contexte* de la vue multirésolution. Une ligne en pointillés verte montre les classes affichées dans la zone de *focus* de la vue multirésolution. Lorsque l'utilisateur survole un sommet se trouvant sur une de ces lignes, une flèche rouge (resp. verte) apparaît et permet de descendre (resp. monter) la ligne d'un niveau. Bien entendu, la flèche rouge (resp. verte) ne s'affiche pas si la ligne passe par le dernier (resp. premier) niveau ou si l'autre ligne passe par le niveau inférieur (resp. supérieur). Lorsque l'utilisateur clique sur ces flèches, les deux autres vues sont automatiquement mises à jour comme le montre la figure 2.6. Cliquer sur un sommet de la hiérarchie permet aussi de masquer/afficher une classe de séries dans les différentes vues [B4] (*Reduce -> Filter* dans le résumé de l'espace de conception schématisé de la figure 1.3).

La figure 2.7 illustre les stratégies possibles pour juxtaposer des vues selon deux axes : la quantité de données partagée et le type d'encodage utilisé. Au regard de cette classification, on peut remarquer que les vues d'ensemble et multirésolution tombe dans la catégorie *Overview/Detail* dans la mesure où elles sont basées sur la même technique d'encodage (diagramme en couches empilées) mais que la vue multirésolution ne montre qu'un sous-ensemble des données. Concernant la combinaison de la vue hiérarchique avec les deux autres vues, on tombe dans la catégorie *Multiform* car deux encodages sont utilisés pour montrer l'ensemble des données.

2.1.2 Conception des algorithmes

La principale contribution de *MultiStream* se situe au niveau de l'idiome proposé dans la section précédente (troisième niveau du modèle imbriqué). Il a néanmoins été nécessaire de proposer un algorithme pour calculer les tailles des intervalles des différentes zones de la vue

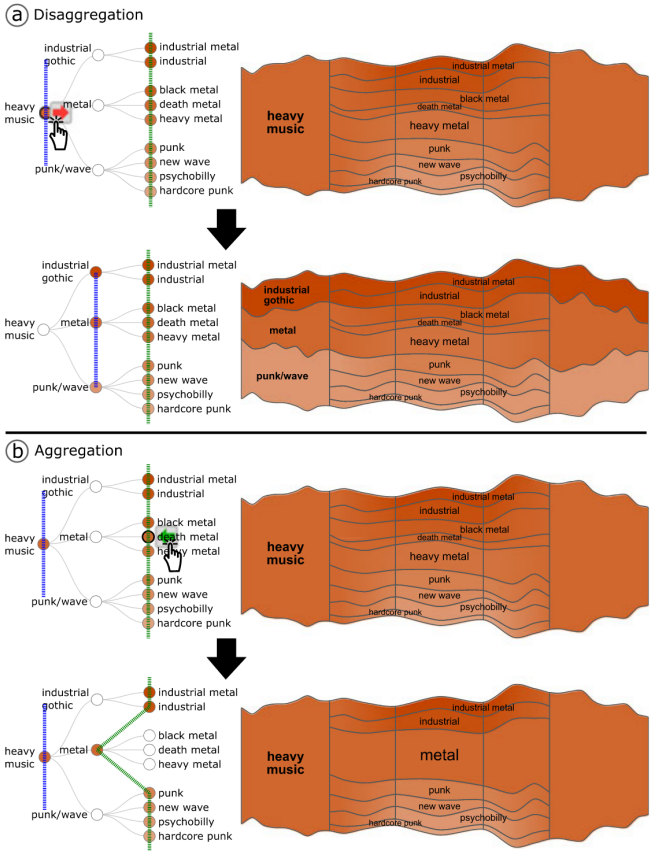


FIGURE 2.6 – La vue hiérarchique permet de : (a) sélectionner le niveau d'abstraction de la vue d'ensemble et des zones de *contexte* de la vue multirésolution, (b) sélectionner le niveau de détail de la zone du *focus* de la vue multirésolution (b).

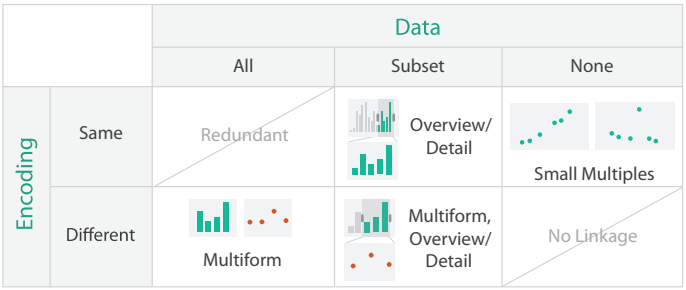


FIGURE 2.7 – Espace de conception pour la juxtaposition de vues coordonnées [133]

multirésolution. Le détail est disponible dans notre article [42] et je vais maintenant vous en donner un aperçu.

La figure 2.8 schématise le découpage de l'espace dédié aux vues d'ensemble et multirésolution. Comme on peut l'observer, dans la vue d'ensemble, les pas de temps de temps sont répartis uniformément sur l'axe des x. Le nombre de pas N devant être affichés dans la vue multirésolution, ainsi que les nombres de pas dans les zones de *contexte* (c_1 et c_2), de transition (t_1 et t_2) et de *focus* (d) sont définis par l'utilisateur grâce au contrôleur. L'objectif de l'algorithme est donc de découper la longueur S de la vue multirésolution de façon à afficher les N pas de temps selon la distorsion évoquée ci-dessus. De plus, nous avons introduit des coefficients, $\alpha > \beta > \gamma$, qui permettent de gérer le degré de cette distorsion.

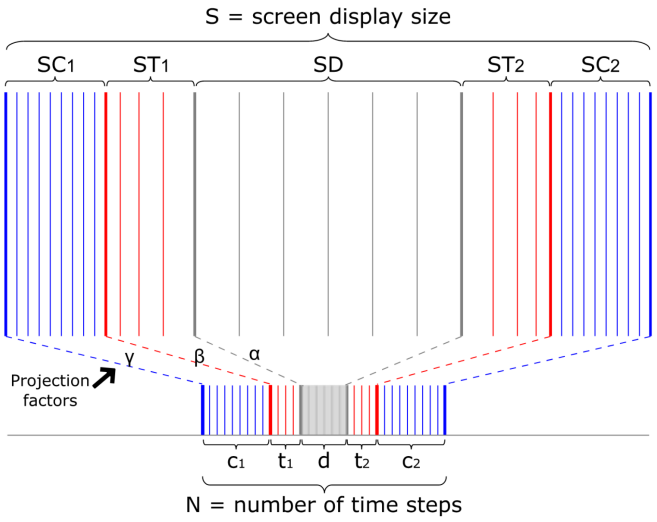


FIGURE 2.8 – Découpage de l'espace dédié aux vues d'ensemble et multirésolution

La figure 2.9 montre les notations des longueurs utilisées pour les pas de temps dans les différentes zones de la vue multirésolution. Comme on peut l'observer, les longueurs de la zone de *focus* et des zones de *contexte* sont des constantes : ID, IC_1 et IC_2 [6]. Elles dépendent des nombres de pas de temps sélectionnés et des coefficients de distorsion :

$$ID = S.\frac{\alpha}{\alpha.d + \beta.t_1 + \beta.t_2 + \gamma.c_1 + \gamma.c_2}$$

$$IC_i = S.\frac{\gamma}{\alpha.d + \beta.t_1 + \beta.t_2 + \gamma.c_1 + \gamma.c_2}$$

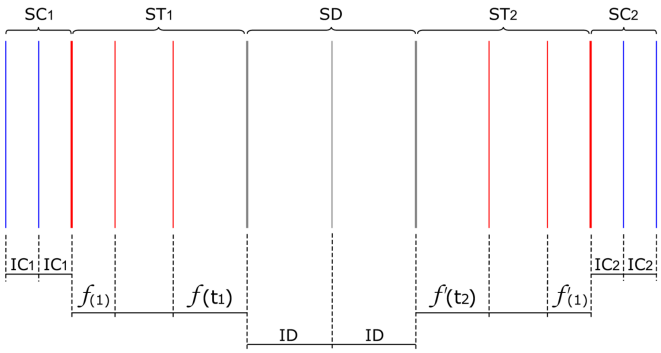


FIGURE 2.9 – Notations des longueurs utilisées pour les pas de temps dans les différentes zones de la vue multirésolution.

Le calcul des longueurs des pas de temps dans les zones de contexte est plus délicate. En effet, il s'agit de trouver une fonction $f(x) = a.b^x$ dans laquelle x est un pas de temps, a est la longueur du plus petit pas de temps et b est la *base*.

Puisqu'il est nécessaire que la longueur du premier pas de temps soit au moins aussi grande que la longueur des pas de temps du *contexte*, nous prenons $a = IC_i$.

Il reste donc maintenant à définir b tel que $f(t_i) \leq ID$. De plus, b doit être supérieur ou égal à 1 pour éviter que la longueur du premier pas de temps soit inférieure à la longueur des pas

6. Précisons ici que les longueurs des pas de temps dans les zones de *contexte* sont égales ($IC_1 = IC_2$). Cependant, les longueurs des zones elles-mêmes dans la vue varient ($SC_1 \neq SC_2$) car l'utilisateur peut définir un nombre de pas de temps différent dans les deux contexte grâce au contrôleur ($c_1 \neq c_2$ et $SC_i = c_i.IC_i$).

de temps du *contexte*. Lorsque l'utilisateur déplace les barres verticales du contrôleur, nous le calculons grâce à l'algorithme RPOLY [90] appliqué à l'équation :

$$ST_i = a \cdot b^1 + a \cdot b^2 + \ldots + a \cdot b^{t_i}$$

où ST_i est la longueur de la zone de transition. Nous appliquons ensuite les règles de mise à jour suivantes :

Si $b \leq 1$, nous ne mettons pas les vues à jour.

Sinon si $f(t_i) \geq ID$, nous ne mettons pas les vues à jour.

Sinon, nous mettons à jour le contrôleur et la vue multirésolution en utilisant les valeurs de la fonction f.

2.1.3 Validation

Dans ce travail, nous nous situons principalement dans le troisième niveau du modèle imbriqué (voir la figure 1.5 -> *Conception des idiomes*). Nous avons donc réalisé deux études de cas illustrant comment notre outil peut être utilisé pour réaliser les tâches indiquées ci-dessus. La première est basée sur un jeu de données contenant les sentiments exprimés sur *Twitter* lors de l'élection présidentielle américaine de 2016. Je ne la détaillerai pas ici mais j'invite le lecteur intéresser à se référer à notre article [42]. La seconde, que je vais maintenant détailler, a été réalisée avec le jeu de données musicales présenté en introduction de cette section.

La vue d'ensemble de la figure 2.10 montre un pic de la couche orange (*rock music*) entre le milieu des années 1970 et le milieu des années 1980 [B1]. Afin d'explorer en détail la composition de cette couche durant ce pic, nous utilisons le contrôleur pour positionner le *focus* de la vue multirésolution sur la période tout en gardant un *contexte* [B2]. On peut ainsi observer que le pic est dû à une forte augmentation de la valeur d'une seule des sous-couches de *rock music* : *heavy music*.

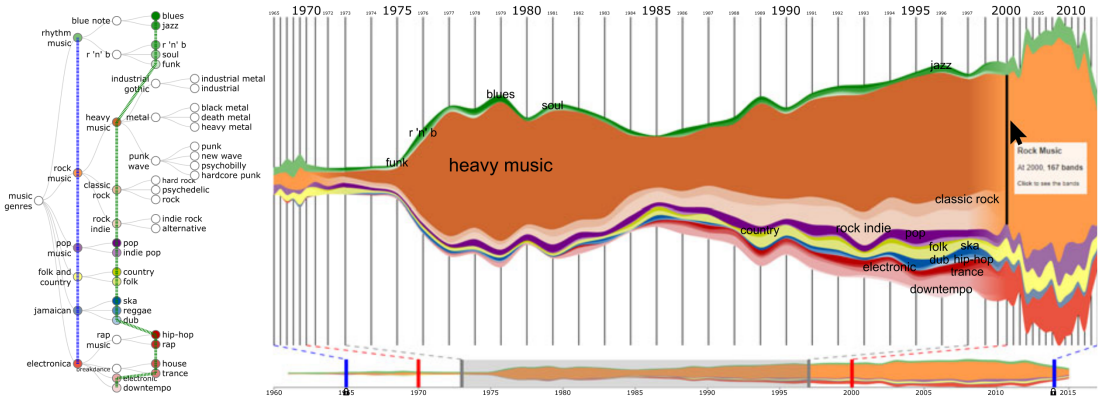


FIGURE 2.10 – Le pic de groupes de *rock music* dans les années 1975-1985 est dû à une forte croissance du genre *heavy music*.

Afin d'étudier plus en détail le phénomène, nous modifions les niveaux de détail affichés dans les zones *contexte* et dans la zone de *focus* en utilisant la vue de la hiérarchie [B3]. Comme le montre la figure 2.11 la ligne bleue (*contexte*) a été descendue d'un niveau, elle ne passe plus par la classe *rock music* mais ses sous-classes, *heavy music*, *classic rock* et *rock indie*. De même, la ligne verte (*focus*) passe maintenant par le dernier niveau de la hiérarchie. On observe ainsi que le pic des années 1975-1985 est majoritairement dû à une seule série du dernier niveau : *punk/wave*. On observe aussi qu'après une baisse au milieu des années 1980, les couches oranges (*rock music*) connaissent un nouvel essor entre 1985 et 1995. Même si la couche *punk/wave* est toujours présente sur cette période, elle reste stable. La croissance est due à l'émergence de groupes dans des classes qui étaient auparavant marginales : *heavy metal*, *death metal*, *black metal* et dans une moindre mesure, *indie rock* et *alternative*.

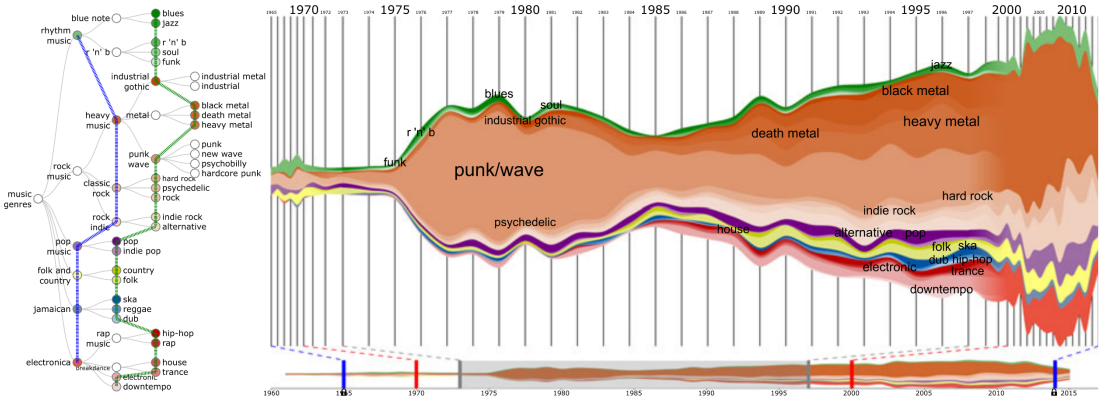


FIGURE 2.11 – Alors que le pic des années 1975-1985 est dû au sous-genre *punk/wave*, la forte croissance des années 1985-1995 est due aux sous-genres *heavy metal*, *death metal*, *black metal*, *indie rock* et *alternative*

Les figures 2.10 et 2.11 nous montrent que les séries représentées par la couche orange dominent le jeu de données. Par conséquent, il est difficile de visualiser les comportements des autres classes. Afin de remédier à cela, nous masquons cette couche en cliquant sur le nœud correspondant dans la vue de la hiérarchie [B4]. Nous obtenons ainsi la visualisation de la figure 2.12, dans laquelle on peut par exemple observer à quelle période la couche *rap music* est apparue (début des années 80). Il est aussi intéressant de voir que la couche verte (*rhythm music*) a dominé entre 1965 et 1970, alors que depuis 1975, c'est la couche rouge qui a pris de plus en plus d'importance (*electronica*). En particulier, la zone du *focus* montre un pic remarquable pour la sous-classe *electronic* en 2007.

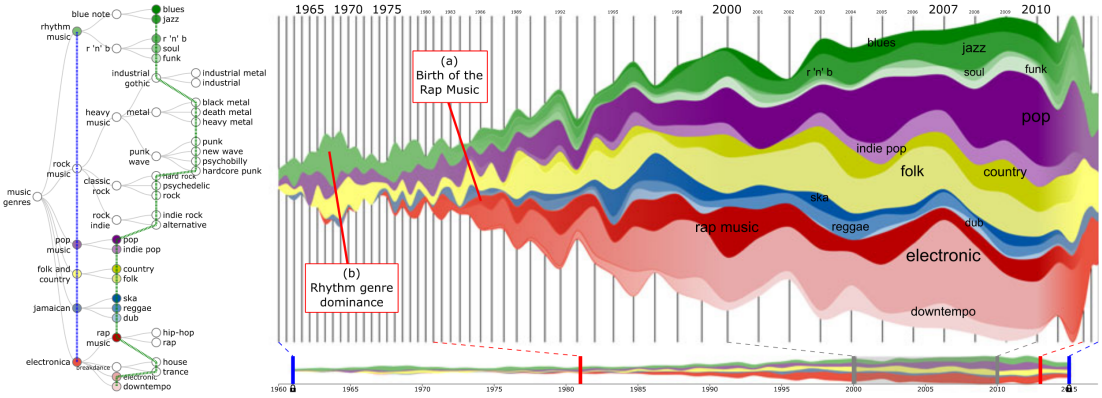


FIGURE 2.12 – Masquer le genre *rock music* permet de visualiser plus en détail les autres genres (*rhythm music*, *electronica*...) et leurs sous-genres (*rap music*, *electronic*...).

2.1.4 Synthèse

L'étude que je viens de présenter illustre la façon dont *MultiStream* répond aux besoins définis plus haut. Elle constitue donc un argument en faveur de l'idiome choisi au regard des tâches et des abstractions qui en ont motivé la conception (troisième niveau du modèle imbriqué). Il est ici important de noter que cette étude n'a pas vocation à valider les abstractions des tâches et des données. C'est pourquoi nous y avons décrit les structures visualisées et les opérations qui ont permis de les trouver, mais que nous n'en avons pas tiré de conclusions portant sur les données (les noms des genres ont juste été mentionnés pour faciliter la lisibilité). Afin de voir un exemple d'application de *MultiStream* à un domaine particulier, j'invite le lecteur à lire notre article [123] dans lequel nous nous intéressons à la caractérisation des discours à propos du VIH dans les forums de santé.

2.2 Visualisation interactive de grands graphes dynamiques

Toujours dans le cadre de la visualisation de données temporelles, je me suis intéressé à l'exploration de grands graphes dynamiques. Dans la littérature, on trouve deux types de données portant cette appellation : les graphes dont les sommets et les arêtes sont associés à des attributs variant au cours du temps et les graphes dont la topologie varie au cours du temps [152]. Nous nous sommes ici intéressé à la seconde catégorie.

Beck *et al.* [16] ont proposé une étude très complète des différentes méthodes permettant de visualiser des graphes dynamiques et j'invite le lecteur intéressé à s'y référer. Mentionnons simplement que l'on trouve dans la littérature deux grandes familles d'approches :

1. Animation montrant l'évolution au cours du temps (*Manipulate* -> *Change* dans le résumé de l'espace de conception schématisé de la figure 1.3),
2. Juxtaposition des représentations à différents pas de temps (*Facet* -> *Juxtapose* dans le résumé de l'espace de conception schématisé de la figure 1.3)

Le principal défaut de la seconde approche, dite *petits-multiples*, est de diviser l'écran, réduisant ainsi l'espace dédiée à la visualisation d'un pas de temps. En revanche, contrairement à la première[7], elle permet de comparer des pas de temps éloignés.

Nous avons proposé un idiome hybride ainsi que les algorithmes permettant de le mettre en place. L'objectif était de réaliser les tâches classique associées aux graphes (identifier des caractéristiques topologiques) et au caractère dynamique des données (comparer ces structures dans le temps). De plus, nous nous sommes fixés un objectif de scalabilité, *i.e.* la possibilité de traiter interactivement des graphes contenant des dizaines de milliers de sommets et des centaines de milliers d'arêtes. Nous avons ainsi proposé une approche consistant à juxtaposer deux vues : l'une statique montrant l'évolution des sommets dans le temps et l'autre interactive permettant de voir le graphe à un moment donné [155].

Même si, dans la suite, je présente l'abstraction des données, puis les idiomes, puis les algorithmes, je tiens à préciser que cela ne correspond pas à la succession dans le temps des tâches que nous avons effectuées. Comme mentionné en introduction, les réponses aux trois questions du concepteur sont liées entre elles, les niveaux du modèle imbriqué ne doivent donc pas être vus comme une succession d'étapes, mais plutôt comme une classification des choix en vu de leur évaluation. Ici particulièrement, l'abstraction des données, *i.e.* les structures dérivées des données brutes que nous avons mises en place afin d'alimenter la visualisation, a été fortement influencé par l'idiome choisi, qui lui-même a été sélectionné pour sa scalabilité.

2.2.1 Abstraction des données

Un graphe dynamique peut être vu comme une séquence $S = \{G_1, G_2, ..., G_k\}$ dans laquelle chaque graphe G_t contient les sommets et les arêtes présentes au temps t. Afin de réaliser l'idiome qui sera décrit dans la section suivante, nous avons commencé par partitionner les sommets des graphes de S en groupes évoluant au cours du temps. Cette opération a été réalisée en deux temps : (1) partition des sommets des graphes de chaque pas de temps (*time-step clustering*), (2) couplage des groupes résultant des partitions des pas de temps de façon à obtenir des groupes évoluant au cours du temps (*time-varying clustering*). Je vais maintenant donner un aperçu des méthodes employées pour chacune de ces étapes. Pour plus de détail, une première version de l'algorithme a été proposée dans l'article initial [155] et a ensuite été raffinée dans un article ultérieur [130] (amélioration de la stabilité des groupes et du temps de calcul).

Partition des graphes Pour chaque graphe $G_t \in S$, nous avons commencé par calculer une partition des sommets de façon à optimiser la fonction de modularité [136]. En effet, comme Noack l'a démontré [138], optimiser cette fonction est équivalent à optimiser une fonction

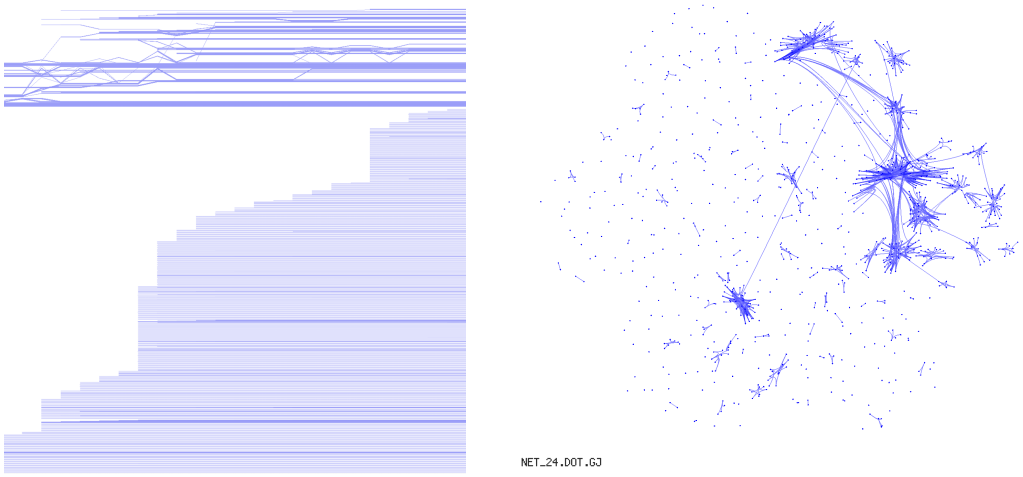
7. Archambault *et al.* [8] ont proposé une expérimentation pour comparer les deux approches en terme de capacité à accomplir certaines tâches.

d'énergie pour le dessin de graphe [26] et, comme nous allons le voir dans les sections suivantes, nous voulons justement utiliser les résultats de ces partitions pour dessiner en temps linéaire notre graphe à un temps donné. Le problème est difficile [27] et nous avons utilisé l'heuristique de Blondel *et al.* [22].

Couplage des groupes La seconde étape consiste à coupler les groupes de sommets les plus proches dans les pas de temps consécutifs. Le principe est le suivant. Soit un groupe g_i issu d'une partition d'un graphe G_t. On regarde quel est le groupe g_j le plus proche dans la partition de G_{t-1} et on le couple avec lui. On considère ainsi que le groupe g_i au temps t est l'évolution du groupe g_j du temps $t-1$. On obtient ainsi un ensemble de groupes évoluant au cours du temps. Si aucun groupe de G_{t-1} n'est proche de g_i, alors g_i est le premier élément d'un nouveau groupe. La proximité est calculée grâce à l'indice de Jaccard [89].

2.2.2 Conception des idiomes

Notre idiome est composé de deux vues juxtaposées : (1) une frise chronologique permet de visualiser l'évolution des groupes dans le temps, (2) un diagramme nœuds-liens permet de visualiser la topologie du graphe à un moment donné. La figure 2.13 en montre un exemple.



(a) (b)

FIGURE 2.13 – Exemple de visualisation de graphes dynamiques : (a) frise chronologique montrant l'évolution des groupes, (b) diagramme nœuds-liens montrant la topologie du graphe à un pas de temps donné.

Frise chronologique Dans cette vue, l'axe x représente le temps. Chaque sommet est représenté par une polyligne dont la position des points de brisure en y représente le groupe auquel il appartient. La figure 2.14(a) illustre notre approche. Elle montre l'évolution de 4 groupes (numérotés le long de l'axe y), 8 sommets (polylignes noires, verte et bleue) et 5 pas de temps (numérotés le long de l'axe x et représentés par des lignes grises verticales). Par exemple, la polyligne bleue représente un sommet qui appartient au groupe 4 aux temps 1 et 2, puis au groupe 3 aux temps 3 et 4. Ce sommet n'est plus présent au temps 5.

Diagramme nœuds-liens Cette approche consiste à représenter les sommets d'un graphe sous forme de points et les arêtes sous forme de (poly)lignes. Il existe un grand nombre de méthodes permettant de positionner les sommets et dessiner les arêtes [170]. Dans notre cas, nous

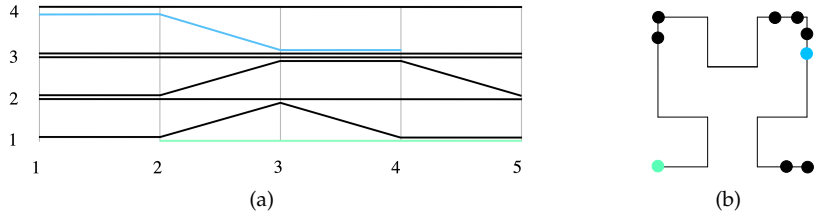


FIGURE 2.14 – Principe de l'idiome mis en place : (a) la frise chronologique montre la répartition des sommets dans les groupes au cours du temps ; (b) les sommets du graphe au temps 3 sont positionnés le long d'une courbe de Hilbert.

nous sommes inspiré de l'article [128] dans lequel les auteurs proposent une solution consistant à positionner les sommets le long d'une courbe fractale [8] (*space filling curve*). L'exemple de la figure 2.14(b) montre le positionnement des sommets du pas de temps 3 de la figure 2.14(a) le long d'une telle courbe (courbe de Hilbert). L'avantage de cette technique est que, pour un ordre prédéfini des sommets, le temps de placement le long de la courbe est linéaire (nous verrons dans la section suivante comment calculer cet ordre), ce qui nous permet d'assurer la scalabilité évoquée en introduction. Nous pouvons donc afficher un graphe pour n'importe quel pas de temps de façon interactive. Un autre avantage est lié à une propriété de ce type de courbes, connue sous le nom de *Worst-Case Locality* [77]. Elle garantit que la distance euclidienne entre les sommets dans le dessin est bornée par la distance des mêmes sommets dans l'ordre prédéfini. Par conséquent, si cet ordre dépend des résultats du partitionnement calculé précédemment, le dessin a tendance à rapprocher les sommets des mêmes groupes. Pour accentuer la visualisation des groupes, nous appliquons également une technique de *bundling* d'arêtes [85] en utilisant comme points de contrôle les centroïdes des groupes. La figure 2.13(b) montre un exemple du résultat final en utilisant une courbe de Gosper.

Interaction Le principal atout de notre approche est de permettre l'affichage instantané du graphe pour n'importe quel pas de temps grâce à la méthode décrite dans le paragraphe précédent. Pour cela, l'utilisateur doit cliquer sur le pas de temps correspondant dans la frise chronologique, ce qui provoque une mise à jour du diagramme nœuds-liens. Une interpolation des positions du pas de temps précédent vers le nouveau pas de temps sélectionné permet de suivre les mouvements des sommets. Notre prototype inclut aussi la possibilité d'animer le graphe pour observer son évolution pas à pas.

2.2.3 Conception des algorithmes

La création de notre idiome nécessite d'une part l'ordonnancement des groupes et d'autre part l'ordonnancement des sommets à l'intérieur des groupes pour chaque pas de temps. Le premier permet de définir la position y des groupes dans la frise chronologique, le second celle des polylignes représentant les sommets dans chaque groupe et pour chaque pas de temps. Ils sont aussi utilisés pour placer les sommets le long de la courbe fractale dans la vue nœuds-liens.

Ordonnancement des groupes L'objectif ici est de trouver un ordre permettant de rapprocher les groupes partageant des sommets au cours du temps afin de réduire le nombre de croisements des polylignes. Par exemple, il n'y a pas de croisement dans la figure 2.14(a), mais il y en aurait si les groupes 2 et 3 étaient inversés le long de l'axe y. Pour trouver cet ordre, nous commençons par créer un graphe dont les sommets sont les groupes et dans lequel il existe une arête de poids p entre deux groupes si ces deux groupes échangent p sommets au

8. J'avais déjà travaillé sur ce type d'approches pour visualiser des hiérarchies pendant ma thèse [11]. Ma collaboration avec Chris Muelder et Kwan-Liu Ma, auteurs de l'article [128] et co-auteurs de l'article présenté ici, a donc permis de trouver une autre application à cette approche.

cours du temps. Notre approche consiste alors à trouver un ordre minimisant la fonction d'arrangement linéaire [140] de ce graphe, ce qui est un problème NP-complet [67]. Nous utilisons l'heuristique de Koren et Harel [97] pour en trouver un solution approchée.

Ordonnancement des sommets Ici aussi, l'objectif de l'ordonnancement est de diminuer le nombre de croisements des polylignes. Pour cela, pour chaque sommet v d'un groupe d'un temps t, nous calculons la médiane des positions y des groupes auxquels appartient v aux autres temps. Les sommets dans le groupe sont ensuite ordonnés en fonction de cette valeur.

2.2.4 Validation

Comme pour *MultiStream*, ce travail s'inscrit principalement dans le troisième niveau du modèle imbriqué [9] (voir la figure 1.5 -> *Conception des idiomes*). Nous avons donc réalisé une étude de cas pour illustrer la façon dont notre approche peut être utilisée pour réaliser les tâches classiques liées aux graphes dynamiques (voir l'introduction de cette section).

La figure 2.15 montre un graphe représentant le réseau *Internet* entre 2001 et 2009. Il contient 41928 sommets et 218080 arêtes. Les couleurs représentent les différentes régions du monde. Même à cette échelle, le rendu est suffisamment rapide pour assurer la fluidité des transitions lors du changement de pas de temps ou de l'animation. La frise chronologique est dense. Cependant, il est quand même possible d'en extraire des motifs intéressants (voir la figure 2.15(a)). Tout d'abord, on peut observer plusieurs sections horizontales denses et assez stables dans le temps. Ensuite, on peut observer un déplacement graduel des sommets du bas vers le haut de la frise.

Comme le montrent les figures 2.15(b-f), ces motifs s'observent aussi dans le diagramme nœuds-liens. Le groupe de sommets majoritairement verts (côte ouest des États-Unis) à gauche reste assez stable, tout comme les groupes bleu (Russie) et rouge (Asie) au centre (même si le bleu se divise en fin de période). Les autres sommets verts (États-Unis) évoluent de façon assez chaotiques jusqu'aux deux dernières figures dans lesquelles ils convergent en haut à droite du diagramme. Les sommets violets (Europe) sont aussi intéressants à observer. On voit un groupe important et assez stable dans la première partie de la période étudiée, puis un autre groupe, lui aussi stable, dans la seconde partie de la période. La partie intermédiaire est quant à elle très chaotique comme l'illustre la figure 2.15(d).

2.2.5 Synthèse

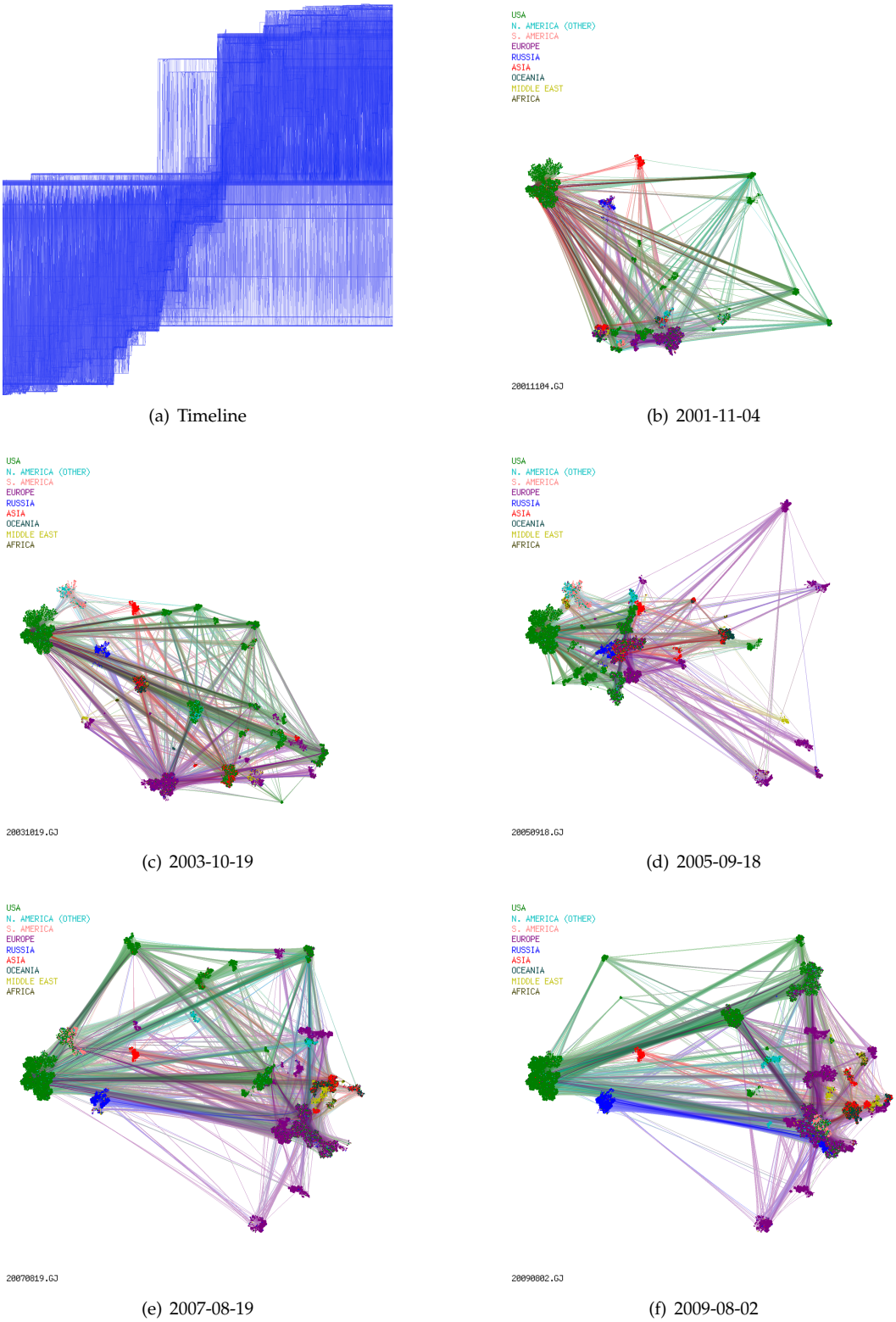
Comme pour *MultiStream*, l'étude que je viens de présenter illustre comment la visualisation permet de découvrir des motifs sans en tirer des conclusions sur les données elles-mêmes. Ceci est en accord avec la validation du troisième niveau du modèle imbriqué.

En plus du raffinement de l'algorithme de partitionnement que j'ai déjà évoqué [130], nous avons aussi proposé une amélioration du rendu de la frise chronologique ainsi qu'une approche permettant d'explorer graduellement les graphes dynamiques [129]. Ces expertises nous ont amené à contribué au chapitre intitulé *Analysis and Visualization of Dynamic Networks* dans l'*Encyclopedia of Social Network Analysis and Mining* [196].

Les données réelles pouvant être modélisées à l'aide de graphes sont nombreuses et diverses. Elles ont souvent la particularité d'associer à leurs éléments (sommets, arêtes) des attributs de tout type (un graphe dynamique peut d'ailleurs en constituer un exemple puisque l'on peut associer à chaque sommet/arête une séquence de booléens représentant sa présence au cours du temps). On les appelle alors des graphes multivariés [93, 139]. Dans ce cadre, nous avons proposé *ContactTrees* [157], un idiome permettant de visualiser le réseau social d'un individu de façon à voir la répartition de ses contacts selon différents attributs.

Un autre type de modélisation est le graphe multicouches [94], *i.e.* un ensemble de couches contenant des sommets et des arêtes (les arêtes pouvant être inter- et intra-couches). Par exemple

9. L'amélioration de l'algorithme de partitionnement proposée dans notre article ultérieur [130] se situe principalement dans le quatrième niveau et contient donc une validation et terme de temps de calcul.



(a) Timeline (b) 2001-11-04

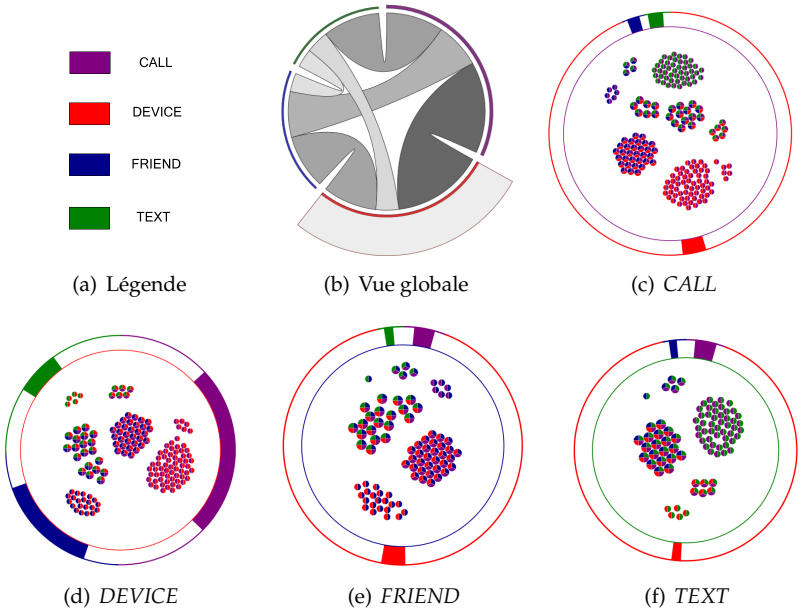
(c) 2003-10-19 (d) 2005-09-18

(e) 2007-08-19 (f) 2009-08-02

FIGURE 2.15 – Évolution du réseau *Internet* entre 2001 et 2009.

dans le cas des graphes dynamiques, les couches peuvent représenter les pas de temps. Un autre exemple est celui des réseaux de communications où les sommets représentent des individus et les couches les moyens de communication utilisés. Enfin, dans les réseaux de transports, les sommets peuvent correspondre aux villes et les couches aux types de transports.

Les approches basées sur ce modèle très flexible en visualisation connaissent actuellement une forte progression [122]. Dans ce cadre, nous avons proposé une nouvelle visualisation permettant de mettre en évidence les nombres d'arêtes partagées à travers les différentes couches [148] (voir la figure 2.16).



(a) Légende (b) Vue globale (c) *CALL*

(d) *DEVICE* (e) *FRIEND* (f) *TEXT*

FIGURE 2.16 – Graphe multicouches dans lequel les sommets représentent des individus, les arêtes des interactions et les couches des types d'interaction : (a) légende des couleurs utilisées dans les autres vues ; (b) vue globale montrant le nombre de liens partagés entre les couches ; (c-d) vues montrant comment comment les liens présents dans une couche se répartissent dans les autres couches.

2.3 Algorithmes de suppression de chevauchement

Toujours dans le cadre des graphes multi-couches, nous avons proposé une méthode de *bundling* permettant de différencier les arêtes selon leur couche d'appartenance [24]. Notre méthode commence par appliquer un algorithme de *budling* [103] des graphes simples. Cela produit un dessin dans lesquelles les arêtes des différentes couches se superposent dans les *bundles*. Ensuite, nous avons proposé un algorithme pour supprimer ces chevauchements en juxtaposant les arêtes en fonction de la couche auxquelles elles appartiennent. La figure 2.17 montre le résultat, dans lequel les couleurs représentent les couches.

Supprimer les chevauchements d'un dessin initial en vu de faciliter la lisibilité d'un graphe peut aussi concerner les sommets. Je vais maintenant résumer une étude comparative d'algorithmes [36, 37] que nous avons réalisée. Cette étude fait suite à un autre travail dans lequel nous avions proposé un algorithme de suppression de chevauchements de sommets dans des dessins de graphes en 1D [58, 59].

Il existe de nombreux algorithmes permettant, à partir d'un dessin initial, de déplacer les sommets de façon à supprimer les chevauchements tout en gardant la structure globale du graphe. Lorsque l'on met en place un idiome nécessitant le recours à de tels algorithmes, le choix n'est pas simple. En effet, les validations mises en place dans les différents articles ne comparent jamais l'algorithme proposé avec tous ses concurrents et les indices de qualité permettant de les comparer ne sont pas toujours les mêmes. Au total, nous avons recensé 22 indices définis dans les différents travaux. Nous avons réparti ces indices en cinq classes en fonction de la propriété qu'ils mesurent, puis nous en avons choisi un représentatif par classe. Nous avons

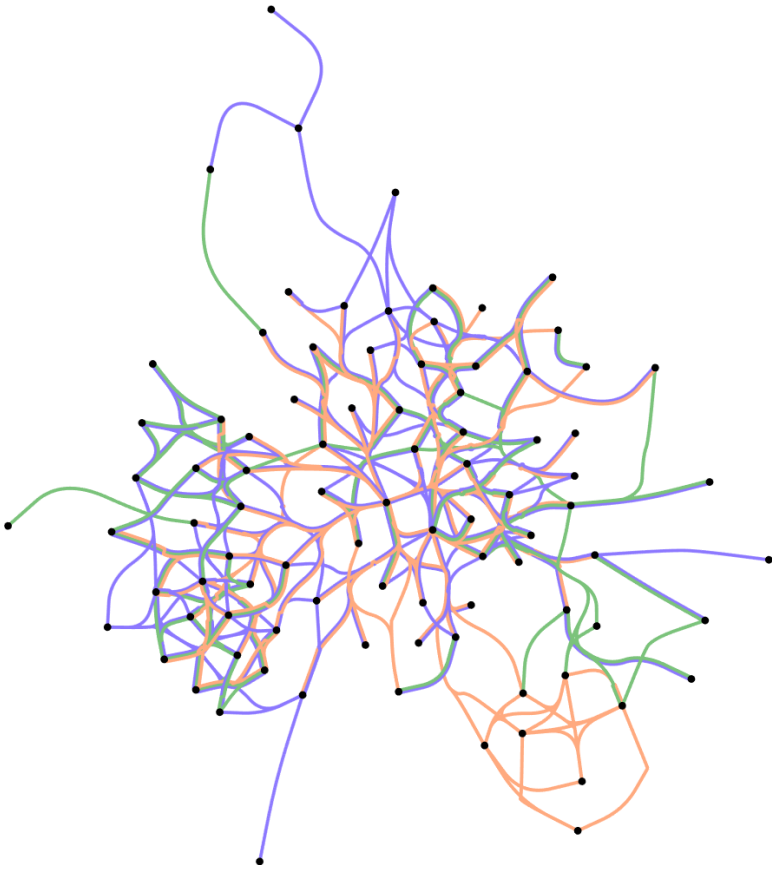


FIGURE 2.17 – *Bundling* d'arêtes pour les graphes multicouches.

ensuite comparé les résultats des algorithmes au regard de ces cinq indices et en fonction de leur temps de calcul.

2.3.1 Indices de qualité

Je vais maintenant présenter les cinq indices de qualité sélectionnés. La description de tous les indices, ainsi que les raisons de nos choix sont expliqués dans notre article [37]. Dans la suite, V est l'ensemble des sommets d'un graphe, E l'ensemble de ses arêtes. La position initiale d'un sommet est notée $v = (x_v, y_v)$, sa position après application d'un algorithme de suppression de chevauchements $v' = (x'_v, y'_v)$. Le nombre de sommets est n, le nombre d'arêtes m.

Préservation de l'ordre orthogonal Un première classe d'indices porte sur la préservation de l'ordre orthogonal [127], *i.e.* la préservation de l'ordre des sommets sur l'axe x et sur l'axe y. Pour notre étude, nous partons d'un indice proposé par Strobelt *et al.* [167] qui consiste à additionner le nombre d'inversions sur chaque axe :

$$nbinv = \sum_{\substack{(u,v)\in V^2 \\ x_u > x_v}} \begin{cases} 1 & \text{if } x'_u < x'_v \\ 0 & \text{otherwise} \end{cases} + \sum_{\substack{(u,v)\in V^2 \\ y_u > y_v}} \begin{cases} 1 & \text{if } y'_u < y'_v \\ 0 & \text{otherwise} \end{cases}$$

Nous normalisons ensuite ce nombre pour obtenir une valeur entre 0 et 1, 0 signifiant que l'ordre orthogonal est respecté :

$$ordre = \frac{nbinv}{n(n-1)}$$

Minimisation de l'expansion Lorsque l'on augmente le niveau de *zoom* tout en gardant les tailles initiales des sommets, les chevauchements sont supprimés. Cependant, cette solution n'est pas souvent adaptée car elle a tendance à créer des dessin très étendus. Les indices de cette seconde classe cherchent donc à évaluer l'expansion du dessin, l'objectif étant de favoriser les dessins les plus compacts. Celui que nous avons retenu a été proposé par Strobelt *et al.* [167] et consiste à calculer le ratio de l'aire de l'enveloppe convexe finale, ch', avec celle de l'enveloppe convexe initiale, ch :

$$expansion = \frac{area(ch')}{area(ch)}$$

Cet indice est supérieur ou égal à 1, 1 étant la valeur optimale.

Préservation de la forme globale Cette classe comprend les indices permettant d'évaluer la déformation globale du dessin. Nous proposons un indice adapté de Li *et al.* [107] qui consiste à calculer le rapport des ratios d'aspect (*aspect ratio*) des boites englobantes des deux dessins :

$$forme = \max\left(\frac{w'_{bb} \times h_{bb}}{h'_{bb} \times w_{bb}}, \frac{h'_{bb} \times w_{bb}}{w'_{bb} \times h_{bb}}\right)$$

où w_{bb} (resp. w'_{bb}) est la largeur de la boite englobante du dessin intial (resp. final), h_{bb} (resp. h'_{bb}) la hauteur de la boite englobante du dessin intial (resp. final). Cet indice est supérieur ou égal à 1, 1 étant la valeur optimale.

Minimisation du déplacement des sommets Les indices de cette classe évaluent les mouvements relatifs des sommets entre le dessin initial et le dessin final, l'objectif étant de les minimiser. Par exemple, un sommet placé dans le coin en bas à droite dans les deux dessins n'a pas subi de mouvement relatif, même si sa position a varié à cause de l'expansion. L'indice que nous avons choisi, inspiré de trois précédent travaux [64, 167, 118], est basé sur deux fonctions permettant d'aligner les dessins :

$$\begin{aligned} shift(v) &= (x_v + x'_{bb} - x_{bb}, y_v + y'_{bb} - y_{bb}) \\ scale(v) &= (x_v \times \frac{w'_{bb}}{w_{bb}}, y_v \times \frac{h'_{bb}}{h_{bb}}) \end{aligned}$$

Il est ensuite calculé de la façon suivante :

$$deplacements = \frac{1}{n} \times \sum_{v \in V} \|v' - scale(shift(v))\|^2$$

Cet indice est supérieur ou égal à 0, 0 étant la valeur optimale.

Préservation des longueurs des arêtes La dernière classe d'indices regroupe ceux basés sur la préservation relative des longueurs d'arêtes. En s'inspirant d'un indice proposé par Gansner et Hu [64], nous utilisons le coefficient de variation des ratios des longueurs d'arêtes :

$$r_{uv} = \frac{||u' - v'||}{||u - v||}, \quad (u, v) \in E$$

$$\overline{r} = \frac{1}{|E|} \sum_{(u,v)\in E} r_{uv}$$

$$longueurs = \frac{\sqrt{\frac{1}{|E|} \sum_{(u,v)\in E} (r_{uv} - \overline{r})^2}}{\overline{r}}$$

Cet indice est supérieur ou égal à 0, 0 étant la valeur optimale.

2.3.2 Comparaison des algorithmes

Dans notre étude, nous avons comparé les résultats de 9 algorithmes : augmenter le *Zoom*, *PFS* [127], *PFS'* [80], *FTA* [87], *VPSC* [49], *PRISM* [64], *RWordle-L* [167], *GTREE* [134], and *Diamond* [124].

Ces algorithmes ont été testés sur 840 graphes synthétiques contenant 10 à 1000 sommets. Ils ont été générés à l'aide de 4 modèles disponibles dans la librairie *OGDF* [38] : graphe aléatoire [52], arbre aléatoire, graphe petit-mode [189], graphe sans échelle [13]. Nous avons aussi utilisé 14 graphes issus de la suite *Graphviz* [10] [65], précédemment utilisés par les auteurs de *PRISM* [64] et *GTREE* [134]. Le positionnement initial des graphes synthétique a été réalisé avec l'implémentation de FM^3 [75] disponible dans *OGDF*, les graphes de *Graphviz* avec *SFDP* [86]. Après l'application des 9 algorithmes sur les 854 graphes, nous avons obtenu 7686 dessins sans chevauchement sur lesquels nous avons pu calculer les valeurs des 5 indices sélectionnés.

Dans la suite, je ne vais pas décrire les résultats obtenus qui sont présentés en détail dans notre article [37]. Je vais simplement donner les principales conclusions de l'étude.

La figure 2.18 montre les dessins obtenus sur un graphe sans-échelle contenant 100 sommets, 400 arêtes et initialement 274 chevauchements. La figure 2.18 montre les dessins obtenus sur le graphe *mode* de la suite *Graphviz*. Il contient 213 sommets, 269 arêtes et initialement 1105 chevauchements. Les tailles relatives des dessins ont été respectés, sauf pour *Zoom* et *PFS* dans le second exemple (figures 2.19(b) et 2.19(c)) pour lesquels j'ai divisé les tailles par 2 car ils prenaient trop d'espace.

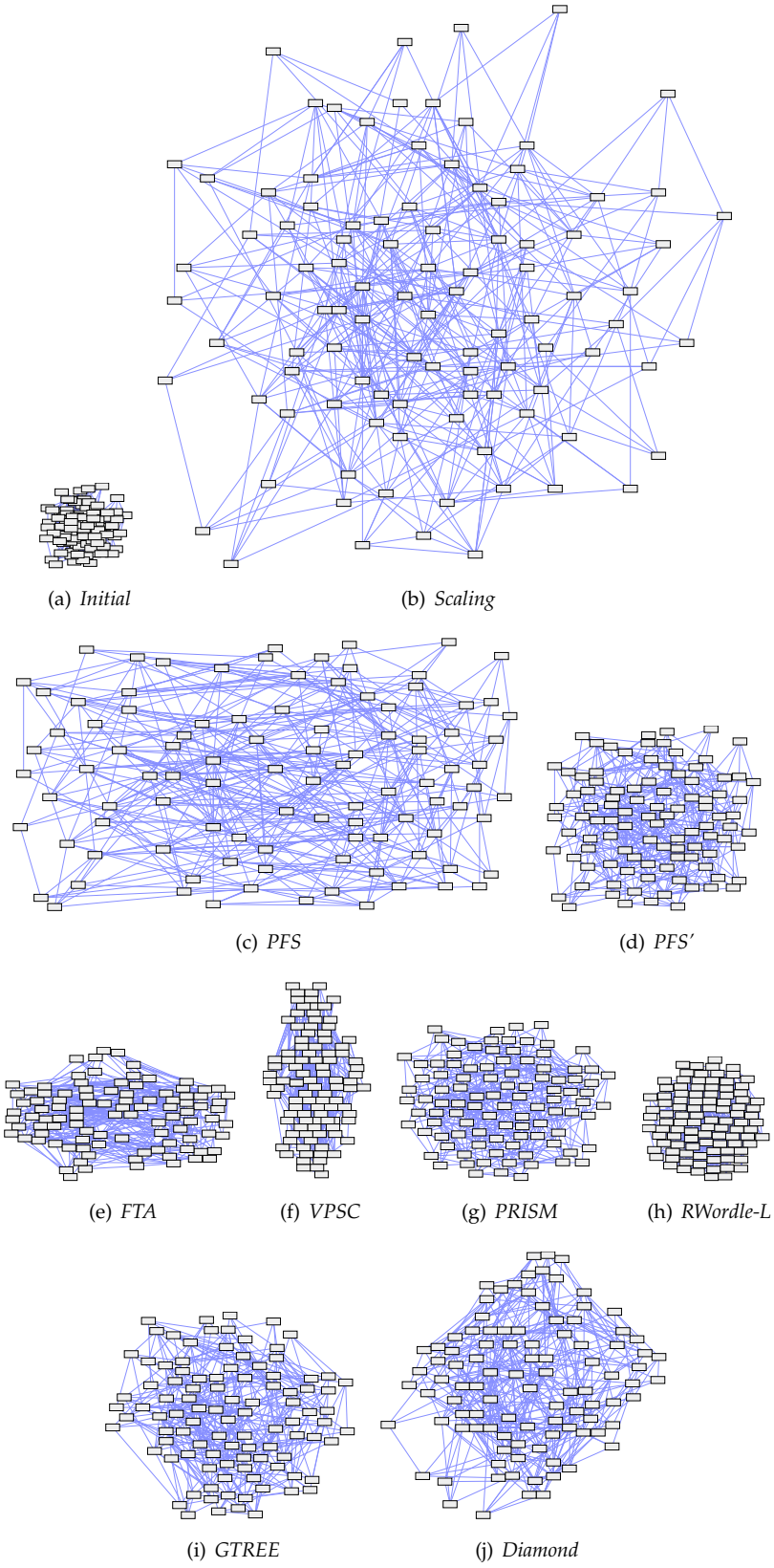
Comme on pouvait s'y attendre, *Zoom* obtient des résultats optimaux sur 4 des 5 indices et est très rapide à calculer. Cependant, il étend énormément le dessin (indice $expansion$) ce qui le rend inopérant dans beaucoup de cas. *PFS* étend aussi énormément les dessins.

FTA obtient des résultats intermédiaires sur tous les indices, ce qui le place en dessous de ses compétiteurs restants. En particulier, il déforme énormément le dessin en l'allongeant le long de l'axe x.

VPSC et *RWordle-L* obtiennent des résultats comparables sur tous les indices excepté la *forme*, pour laquelle *VPSC* obtient de mauvais résultats car il présente le même défaut que celui mentionné ci-dessus pour *FTA* (allongement du dessin sur une dimension). *VPSC* est en revanche plus rapide que *RWordle-L* pour les graphes de plus de 500 sommets. Ces deux algorithmes produisent les dessins les plus compacts (meilleurs scores pour l'indice *expansion*). Cette propriété empêche de visualiser les arêtes et les structures des graphes.

Si l'objectif des la visualisation est de mettre en évidence les chemins et les groupes de sommets, les options restantes sont donc à privilégier (*PFS'*, *PRISM*, *GTREE* and *Diamond*). Parmi elles, *Diamond* est la plus lente, mais elle dépend du solveur [11] que nous avons utilisé et qui

10. https://gitlab.com/graphviz/graphviz/blob/master/rtest/graphs/ [dernière consultation le 19/03/2020]
11. https://github.com/JWally/jsLPSolver [dernière consultation le 19/03/2020]



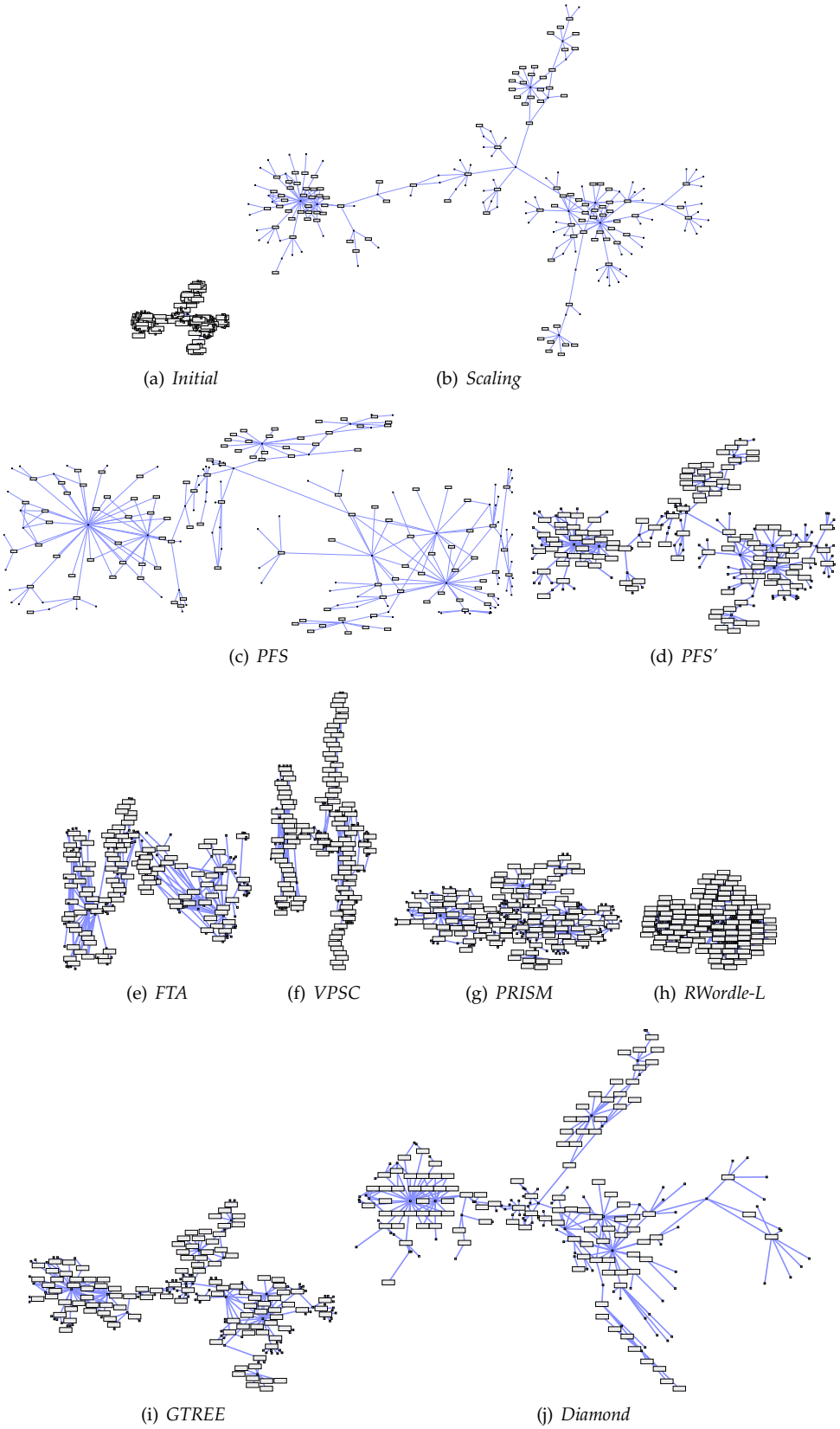
(a) *Initial* (b) *Scaling*

(c) *PFS* (d) *PFS'*

(e) *FTA* (f) *VPSC* (g) *PRISM* (h) *RWordle-L*

(i) *GTREE* (j) *Diamond*

FIGURE 2.18 – Graphe sans-échelle contenant 100 sommets, 400 arêtes et 274 chevauchements.



(a) *Initial* (b) *Scaling*

(c) *PFS* (d) *PFS'*

(e) *FTA* (f) *VPSC* (g) *PRISM* (h) *RWordle-L*

(i) *GTREE* (j) *Diamond*

FIGURE 2.19 – Graphe *mode* de la suite *Graphviz* contenant 213 sommets, 269 arêtes et 1105 chevauchements

n'est vraisemblablement pas aussi rapide que celui décrit dans l'article original [124]. Il obtient aussi de mauvais résultats pour les indices mesurant le déplacement des sommets et la préservation des *longueurs* d'arêtes. *GTREE* induit aussi des déplacements comparables mais est meilleur en terme de préservation des longueurs d'arêtes. *PFS'* et *PRISM* obtiennent des résultats comparables, meilleurs en moyenne que *GTREE* et *Diamond* en terme de déplacement de sommets. *PRISM* est légèrement meilleur, en particulier sur les graphes de *Graphviz*, mais il est aussi plus lent. Il devrait donc être privilégié pour les grands graphes quand l'interactivité n'est pas nécessaire alors que *PFS'* devrait être privilégié dans les autres cas.

Cette étude présente quelques limites en vue d'une généralisation. Tout d'abord, les conclusions que je viens de présenter ont été réalisées à partir des implémentations décrites dans notre article, elles peuvent éventuellement différer à cause de certaines interprétation ou choix laissés libres dans les articles. Ensuite, les résultats peuvent dépendre d'autres facteurs comme la structure du graphe, le placement initial, le nombre de chevauchements ou la forme des sommets. Une discussion adressant ces problèmes est menée dans notre article [37].

2.3.3 Synthèse

Le travail que nous avons mené ici se situe dans les deux derniers niveaux du modèle imbriqué (voir la figure 1.5). En effet, la discussion portant sur le choix des indices de qualité, donc sur ce que doit produire un algorithme de suppression de chevauchements, relève du niveau 3 (*Conception des idiomes*). En revanche, la conception d'un algorithme permettant d'optimiser ces indices issus du niveau 3 relève du niveau 4 (*Conception des algorithmes*). De même, la validation portant sur les indices relève du niveau 3 (c'est un exemple d'*analyse quantitative des images produites*), celle portant sur les temps de calcul du niveau 4.

En terme de type de contribution (voir la section 1.3), j'ai classé ce travail dans le chapitre dédié aux *techniques de visualisation*. J'ai fait ce choix dans la mesure où la discussion portant sur les indices que doit optimiser un algorithme est centrale dans l'article, et relève de la création d'idiomes. Cependant, ce travail pourrait aussi être classé dans le type *étude empirique* grâce à la comparaison des algorithmes réalisée.

Pour finir, l'ensemble du code que nous avons utilisé pour l'expérimentation est disponible en tant que librairie *JavaScript* [12]. De plus, une plateforme *Web*, AGORA [13] (*Automatic Graph Overlap Removal Algorithms*), est aussi disponible en ligne. Elle permet de téléverser des graphes, leur appliquer les algorithmes de suppression de chevauchements, télécharger les résultats et voir les valeurs des 22 indices mentionnés dans notre article.

12. https://github.com/agorajs/agorajs.github.io [dernière consultation le 19/03/2020]
13. https://agorajs.github.io/ [dernière consultation le 19/03/2020]

Chapitre 3

Conception de solutions visuelles pour l'environnement et la santé

Ce troisième chapitre présente mes contributions portant sur la *conception de solutions visuelles* (*design study*). Voici la définition qu'en donnent Sedmair *et al.* dans un article méthodologique sur le sujet [161] : "We define a design study as a project in which visualization researchers analyze a specific real-world problem faced by domain experts, design a visualization system that supports solving this problem, validate the design, and reflect about lessons learned in order to refine visualization design guidelines." La conception d'une telle solution relève des quatre niveaux du modèle imbriqué (voir la figure 1.5). Afin de traiter correctement ces différents niveaux, un projet de visualisation contient un certain nombre de tâches à réaliser qui peuvent être résumées par la figure 3.1 [161]. Les niveaux de validation concernent la section *CORE* et l'item *Reflect* de la section *ANALYSIS*. On remarque ici les flèches orientées vers la gauche qui mettent en évidence le caractère non linéaire du processus.

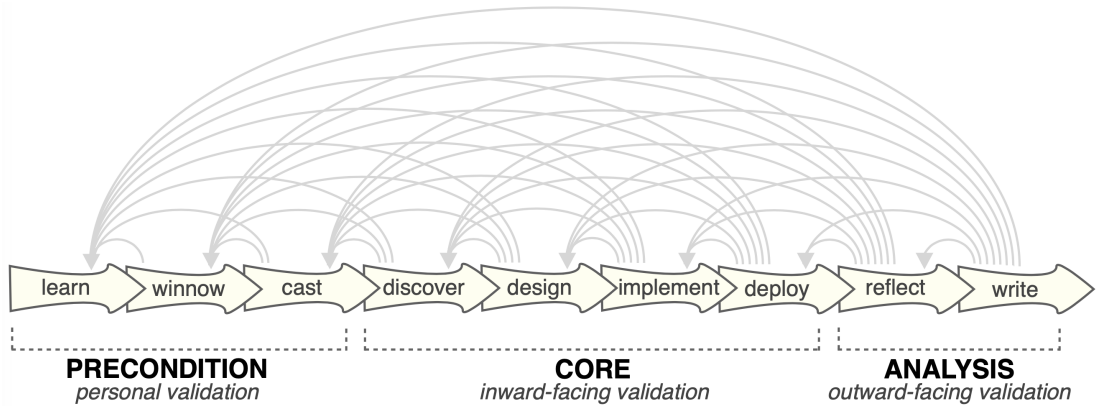


FIGURE 3.1 – Étapes de mise en place d'une solution visuelle [161] : étudier la littérature (*learn*), trouver des collaborateurs (*winnow*), identifier les rôles des collaborateurs (*cast*), identifier le problème (*discover*), concevoir une solution (*design*), implémenter un prototype (*implement*), déployer le prototype (*deploy*), valider et pointer les limites (*reflect*), rédiger l'article (*write*).

Dans ce chapitre, les solutions que je vais présenter concernent l'environnement et la santé, deux thématiques phares de l'équipe ADVANSE [1] du LIRMM dont je fais parti. En particulier, je vais en détailler deux : la première, *HydroQual*, est un outil permettant de suivre la qualité de l'eau des cours d'eau, la seconde, *EpidVis*, est un outil pour faciliter la veille en épidémiologie animale. Comme dans le chapitre précédent, je vais les aborder à travers le prisme du modèle imbriqué. Cependant, elles ont été conçues en accomplissant, dans une certaine mesure, les étapes mentionnées ci-dessus.

1. https://www.lirmm.fr/recherche/equipes/advanse [dernière consultation le 20/03/2020]

3.1 *HydroQual* suivre la qualité de l'eau des cours d'eau

Le développement économique basé sur l'industrialisation des sociétés, l'agriculture intensive et l'augmentation de la population entraîne une dégradation de la qualité de l'eau [125]. Des désastres emblématiques causés par des accidents industriels, comme la pollution du Rhin en 1986 [74], ont mis en évidence l'importance de ce problème pour la santé publique et l'environnement. Ceci a amené les gouvernements à prendre des mesures pour la préservation et la restauration des resources et des écosystèmes aquatiques [191]. Dans ce cadre, les opérationnels ont besoin d'outils pour interpréter les données complexes concernant la qualité de l'eau [2]. C'est pourquoi, en étroite collaboration avec des experts en hydrobiologie, nous avons proposé *HydroQual* [1], une solution visuelle permettant de répondre à ce besoin. Dans une démarche de visualisation analytique [173, 92] (*Visual Analytics*), notre outil combine une interface visuelle et interactive avec des méthodes issues de la fouille de données en vue d'aider les expert à explorer et analyser leurs données.

3.1.1 Définition du problème du domaine d'application

La qualité de l'eau est définie comme sa capacité à soutenir la vie aquatique et à pouvoir être utilisée pour différents usages [179]. Par exemple, l'eau potable ne contient pas de pathogènes et a des limites très strictes de toxicité [54]. La dégradation de la qualité peut avoir différentes causes, comme la contamination par des agents pathogènes (bactéries, *etc.*), des métaux lourds (plomb, *etc.*), des polluants micro-organiques (hydrocarbone, *etc.*), la dégradation de la matière organique et le changement de régime hydrologique [179]. L'origine de ces causes peut être naturelle (géologique, hydrologique ou climatique) et/ou humaine (industrie, agriculture, *etc.*). Ces processus naturels et humains sont entremêlés, ce qui rend difficile l'identification des causes sur lesquelles les décideurs peuvent agir. Pour cela, ils doivent pouvoir identifier les relations entre les paramètres biologiques (poissons, macroinvertébrés, diatomées, macrophytes), les paramètres physico-chimiques (macro et micropolluants comme le nitrate) et les activités humaines (occupation du sol, industrie, plan de recyclage de l'eau, *etc.*), tout en tenant compte des caractéristiques spatio-temporelles des phénomènes [101].

Dans ce cadre, les experts du domaine sont confrontés à différents problèmes : (1) la récolte et le stockage des données, (2) la compréhension des processus liés à la qualité de l'eau et (3) la prise de décision pour une gestion durable des resources en eau [179]. L'objectif de notre solution visuelle est de résoudre le point 1 et de faciliter le point 2 en vue de mener à bien le point 3.

Les données brutes que nous traitons sont issues de différentes stations de prélèvement placées sur des cours d'eau et extraits régulièrement (dimension spatiale et temporelle). Elles se présentent sous la forme de paramètres physico-chimiques et d'indices biologiques. Les premiers sont directement mesurés alors que les seconds, qui représentent la diversité de la faune aquatique, sont calculés en fonction des éléments biologiques présents dans les relevés. Dans notre approche, nous utilisons l'*Indice Biologique Global Normalisé*, IBGN [2], qui fournit une valeur entre 0 et 20 [7]. Les paramètres physico-chimiques et l'indice biologique sont groupés en 5 classes associées à un code couleur selon le système d'évaluation de la qualité des cours d'eau français (voir [6] ou [21]) : très bon (bleu), bon (vert), moyen (jaune), pauvre (orange) et mauvais (rouge).

Les experts ont ainsi besoin de comprendre les différents liens qui unissent ces données. Par exemple, une trop forte présence de matière organique, qui est le signe d'un excès de nourriture, peut expliquer la faible valeur d'un indice biologique. Mais est-ce tout le temps le cas ? Existe-t-il d'autres relations ? Peut-on voir des anomalies dans les données ? Existe-t-il des processus résiliants ? Peut-on identifier de nouvelles sources de problèmes ? Peut-on quantifier les effets d'une politique d'amélioration de la qualité ?

2. D'autres techniques ont déjà été proposées pour visualiser la qualité de l'eau. Certaines sont basées sur une approche géographique [31] qui peut être enrichie à l'aide de graphiques simples [12, 17]. D'autres sont basées sur des idiomes plus élaborés [60, 109, 88, 25]. Aucune d'entre elles ne permet d'effectuer les tâches identifiées ci-après.

3.1.2 Abstraction des tâches

En étroite collaboration avec les experts, nous avons commencé par identifier les questions qu'ils se posent au sujet de leurs données : (Q1) Pour un objectif donné, quel sous-ensemble du jeu de données doit être sélectionné ? (Q2) Pour une zone d'intérêt, quelles stations de prélèvement présentent un comportement similaire ? (Q3) Réciproquement, où se trouvent les stations présentant un comportement similaire ? (Q4) Quelle sont les valeurs des paramètres physico-chimiques et de l'indice biologique d'une station particulière ? (Q5) Pour un ensemble de stations, quelles sont les tendances en terme d'évolution d'indice biologique et de paramètres physico-chimiques ?

Nous avons ensuite transformé ces questions en tâches abstraites devant être accomplies avec la solution visuelle :
[T1] - Sélectioner un sous-ensemble des données (années, mois, indice biologique, paramètres physico-chimiques) pour Q1.
[T2] - Visualiser/naviguer à travers les stations à partir de leur emplacement pour Q2 et Q4.
[T3] - Visualiser/naviguer à travers les stations à partir de leur comportement pour Q3 et Q4.
[T4] - Sélectionner un groupe de stations en fonction de leur emplacement pour Q2 et Q5.
[T5] - Sélectionner un groupe de stations en fonction de leur comportement pour Q3 et Q5.
[T6] - Visualiser les paramètres physico-chimiques et les indices biologiques pour Q4.
[T7] - Extraire et visualiser les motifs temporels des indices biologiques et des paramètres physico-chimiques pour Q5.

3.1.3 Abstraction des données

Pour chaque station, nous disposons d'un ensemble de valeurs de l'indice biologique et des paramètres physico-chimiques mesurés lors de prélèvements à différents moments. La figure 3.2 illustre cela : les lettres A, B, C, D, E, F sont les paramètres ou l'indice, leur couleur montre leur classe au moment du prélèvement.

	January	February	March	April	May
Station 1	A B		D	E	A
Station 2	E	A C B	F		C D E
Station 3		E	A	C	F

FIGURE 3.2 – Données brutes : ensemble des valeurs des paramètres physico-chimique et de l'indice biologique mesurés pour chaque station lors de différents prélèvements.

A partir de cet exemple, nous construisons, pour chaque station, une séquence d'ensemble d'items (*sequence of itemsets*) comme l'illustre la figure 3.3. Cette structure nous permet ensuite de grouper les stations à partir de leur comportement ([T3,T5]) et d'en extraire des motifs temporels fréquents ([T7]).

	Sequences of itemsets
Station 1	< (A B) (D) (E) (A) >
Station 2	< (E) (A C B) (F) (C D E) >
Station 3	< (E) (A) (C) (F) >

FIGURE 3.3 – Séquences d'ensembles d'items permettant de structurer l'évolution des valeurs des paramètres physico-chimique et de l'indice biologique au cours du temps

Partition des séquences Pour effectuer les tâches [T3] et [T5], il faut grouper les stations ayant un comportement similaire. Nous utilisons une algorithme de partitionnement hiérarchique agglomératif. Ce type d'algorithme nécessite d'avoir une mesure de similarité entre les

séquences. Pour cela, nous adaptons la mesure *Dynamic Time Warping* [154], initialement prévue pour calculer la similarité entre des séquences contenant des valeurs continues, de façon à ce qu'elle prenne en compte la nature discrète de nos données (les équations sont disponibles dans notre article [1]).

Extraction des motifs temporels L'objectif ici est d'extraire des motifs temporels, plus particulièrement des motifs partiellement ordonnés. La figure 3.4 en montre un exemple. Chaque chemin du motif correspond à une séquence apparaissant dans au moins deux séquences d'ensembles d'items des stations. Les items apparaissant dans le chemin rouge ont été mis en surbrillance dans les tableau représentant les séquences initiales.

	Sequences
Station 1	< (A B) (D) (E) (A) >
Station 2	< (E) (A C B) (F) (C D E) >
Station 3	< (E) (A) (C) (F) >

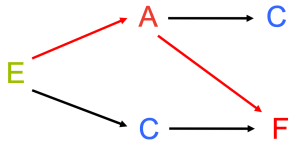


FIGURE 3.4 – Exemple de motif temporel.

L'algorithme que nous avons utilisé est décrit dans notre article [1]. C'est une adaptation de la méthode proposée par Fabrègue *et al.* [56], plus tard améliorée par les mêmes auteurs [55].

3.1.4 Conception des idiomes

La figure 3.5 montre une vue d'ensemble de *HydroQual*. Notre idiome est basé sur une approche multi-vues juxtaposées (*Facet -> Juxtapose* dans le résumé de l'espace de conception schématisé de la figure 1.3).

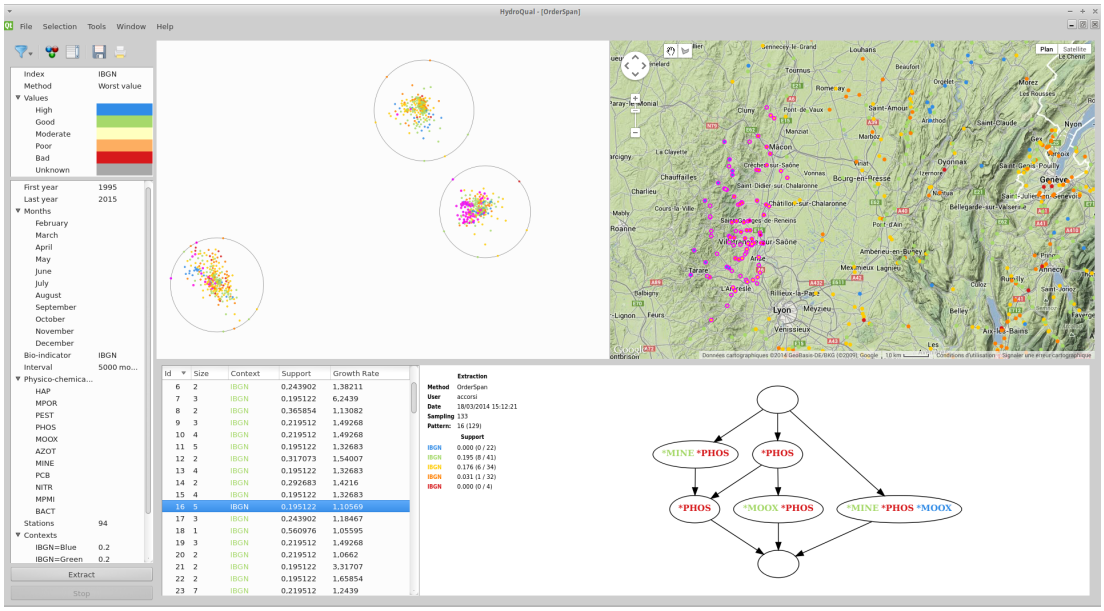


FIGURE 3.5 – *HydroQual*

La vue géographique permet de visualiser les stations de prélèvement sur des fonds de carte issus d'*Open Street Map* et de *Google Maps* [T2]. En particulier, le fond *physical background* de

Google Maps permet de voir les caractéristiques topologique de la zone, ce qui est de première importance pour les experts. Un fond de carte montrant l'occupation du sol (*Corinne Land Cover* [53]) nous a aussi été demandé par les experts et est disponible dans *HydroQual*. Les stations sont représentées par des points, la couleur de ces points montre la valeur du dernier *IBGN* calculé. Les interactions de *zoom* et de déplacements ont été ajoutées afin d'explorer la carte. Nous avons pour cela utilisé la librairie *Leaflet* [3].

La vue de la partition montre les stations regroupées selon leur comportement [T3]. Un cercle gris représente les groupes et des points à l'intérieur représentent les stations. Ici aussi, la coloration des points représente le dernier *IBGN* calculé. La distance entre les points à l'intérieur des groupes et entre les groupes représente la similarité. Nous y reviendrons dans la section sur la conception des algorithmes.

La vue des motifs temporels est composée d'une liste contenant les motifs fréquents extraits du jeu de données ainsi que d'une représentation sous forme de diagrammes nœuds-liens d'un motif particulier [T7]. L'utilisateur sélectionne dans la liste le motif qu'il veut afficher dans la représentation.

Les interactions permettent à l'utilisateur de naviguer à travers les différentes vues. Tout d'abord, il peut charger un jeu de données et le filtrer selon des intervalles de mois, d'années, de valeurs de paramètres physico-chimiques et d'indices biologiques [T1]. Dans la vue géographique (resp. la vue de la partition), l'utilisateur peut sélectionner une station en cliquant dessus ou un ensemble de stations grâce à un lasso. Les motifs temporels sont ensuite calculés en fonction de cette sélection et affichés dans la vue des motifs temporels [T4] (resp. [T5]). Comme nous l'avons dit précédemment, les couleurs des stations dans les vues géographique et de la partition montrent la valeur du dernier indice biologique mesuré. L'utilisateur peut remplacer cette valeur par celle de n'importe quel paramètre physico-chimique [T6].

3.1.5 Conception des algorithmes

Afin de réaliser la vue de la partition, nous avons proposé un algorithme de placement des stations et des groupes. Cet algorithme comporte trois étapes :

1. Nous commençons par positionner les stations dans le plan grâce à un algorithme de positionnement multidimensionnel [4], *Smacof* [106]. L'objectif de ce type de méthode est de positionner les objets de façon à minimiser la différence entre la distance euclidienne dans le dessin et la similarité entre les objects. Dans notre cas, la mesure de similarité est la même que celle utilisée pour effectuer le partitionnement. *Smacof* nécessite un positionnement initial, nous utilisons pour cela l'heuristique *freefold embedding* [143]. Une fois les stations positionnées nous dessinons les groupes à l'aide de cercles dont les centres sont placés au barycentre des stations qu'ils contiennent et les rayons correspondent au nombre de ces stations (voir la figure 3.6(a)).
2. Ensuite, nous détectons les chevauchements entre les groupes et nous appliquons un *Zoom*, tout en maintenant les tailles des groupes fixes, jusqu'à supprimer ces chevauchements (voir la figure 3.6(b)).
3. Pour finir, afin de faciliter la lisibilité des stations à l'intérieur des groupes tout en préservant leurs positions relatives, nous calculons l'enveloppe convexe des stations (voir la figure 3.6(c)), nous positionnons les stations de cette enveloppe sur les points du cercle les plus proches (voir la figure 3.6(d)), nous triangulons le résultats et nous appliquons un algorithme de Tutte [178] permettant de répartir les sommet dans le groupe tout en maintenant la planarité de la triangulation (voir la figure 3.6(e)).

3. https://leafletjs.com/ [dernière consultation le 20/03/2020]
4. Voir [30] pour une introduction au positionnement multidimensionnel (*Multidimensional Scaling*).

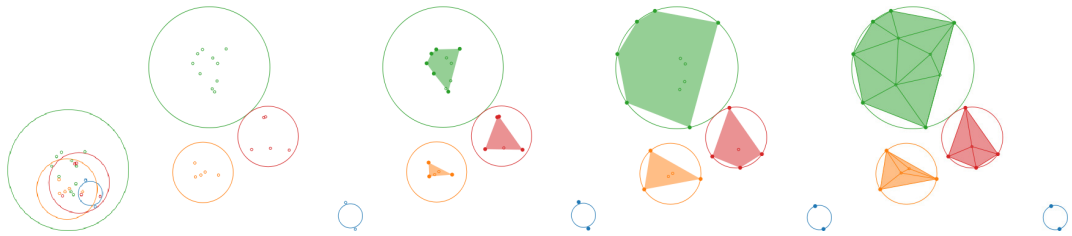


FIGURE 3.6 – Illustration des étapes réalisées pour positionner les stations et dessiner les groupes dans la vue de la partition.

3.1.6 Validation

Comme je l'ai mentionné en introduction de ce manuscrit, la conception de solutions visuelles porte essentiellement sur les trois premiers niveaux du modèle imbriqué. En particulier, comme les techniques de visualisation que nous avons proposées ici ne présentent pas d'innovation importante, la validation de notre approche doit essentiellement porter sur l'évaluation des abstractions des données et des tâches. Pour rappel, cette validation est réalisée en collectant les preuves empiriques auprès d'experts (voir la figure 1.5). Dans le cas d'*HydroQual*, un expert a utilisé l'outil et a réussi à confirmer des hypothèses connues et à en trouver de nouvelles. Il serait trop long ici de décrire le cheminement complet, donc je vais simplement décrire certaines de ses conclusions. J'invite cependant le lecteur intéressé à se référer à notre article [1] pour plus de détails.

Comme exemple de confirmation d'hypothèse, l'expert a observé que la valeur de l'indice biologique *IBGN* est fréquemment précédée dans le temps par des valeurs équivalentes du paramètre physico-chimique *MINE*. De plus, ce paramètre est souvent associé à une valeur équivalente pour le paramètre *MOOX*. Tout ceci indique que que l'absence de pollution organique induit un indice biologique élevé, ce qui est typique des zones dans lesquelles on observe peu d'activités humaines.

L'expert a aussi pu formuler de nouvelles hypothèses grâce à notre solution. Il a ainsi pu observer qu'un faible indice biologique était fréquemment associé à une faible valeur de *PHOS*, mais pas nécessairement à une faible valeur de *MOOX*. De faibles valeurs pour l'association *PHOS-AZOT* indiquent une source de pollution ponctuelle, due par exemple à la présence de bétail ou à des installations de traitement des eaux. A l'inverse, de faibles valeurs pour le paramètre *NITR* indiquent une pollution diffuse due à l'agriculture. L'expert a observé dans certaines zones que le couple *PHOS-AZOT* avait un impact négatif plus important que le *NITR* sur l'indice biologique. Ceci l'a amené à penser que les décisionnaires devaient privilégier, pour les cours d'eau concernés, les décisions portant sur la réduction de pollutions ponctuelles, comme l'amélioration des installations de traitement des eaux ou la limitation de l'utilisation de détergents.

En plus de cette étude, nous avons demandé à deux experts de remplir un questionnaire sur l'outil après quelques jours d'utilisation. Ils ont trouvé *HydroQual* intuitif et répondant à leurs besoins. Ils ont apprécié les encodages visuels et les modes d'interaction utilisés. Ils ont estimé le temps d'apprentissage à 15-30 minutes. Dans un premier temps, la plus grande difficulté à consisté à comprendre le sens des motifs temporels. Cependant, après avoir compris, ils ont tous deux mentionné que la structure était utile pour leurs travaux. Comme suggestion, ils ont proposé de regrouper les motifs selon leur similarité car beaucoup d'entre eux ne présentent que quelques différences. Ce problème reste ouvert.

3.1.7 Synthèse

L'étude décrite ci-dessus permet de valider nos abstractions des données et des tâches. Contrairement à celles qui ont été réalisées dans le chapitre précédent, et conformément aux types de

validation issus du modèle imbriqué, il a été nécessaire de demander à un expert de mener cette étude car elle ne peut être réalisée qu'en connaissant la dimension sémantique des données.

3.2 *EpidVis* : faciliter la veille en épidémiologie animale

Internet joue un rôle fondamental dans la veille épidémiologique. En effet, outre les sites spécialisés, tels que l'OIE [5], diffusant des informations validées mais tardives, une grande quantité de données sont publiées chaque jour et peuvent constituer des indices pour la détection précoce d'épidémies. Les spécialistes en épidémiologie animale lancent donc régulièrement des requêtes sur le *Web* afin d'obtenir des nouvelles récentes. Cette tâche est souvent réalisée manuellement et constitue donc un activité chronophage. C'est pourquoi, en étroite collaboration avec des experts en épidémiologie de l'équipe de recherche *Veille Sanitaire Internationale* [6], nous avons proposé *EpidVis* [7] [57], une solution visuelle permettant d'aider les experts (1) à construire, lancer et sauvegarder des requêtes, (2) les enrichir grâce à des données externes, et (3) visualiser et sauvegarder leurs résultats [8].

3.2.1 Définition du problème du domaine d'application

EpidVis est destiné aux professionnels de santé spécialisés en intelligence épidémique (IE). L'IE englobe toutes les activités liées à l'identification précoce de potentiels risque sanitaires (vérification, estimation, analyse, *etc.*) en vue de recommander des mesures de contrôle de santé. Nous nous sommes focalisés ici sur l'épidémiologie animale mais l'outil peut être étendu à l'épidémiologie humaine.

Le premier signe d'une épidémie est l'émergence de *symptômes*, spécifiques ou non, sur des espèces d'animaux (*hôtes*) susceptibles de porter une *maladie*. L'objectif des experts est de détecter ce type de signes le plus tôt possible. Pour cela, ils suivent le processus décrit dans la figure 3.7. Ils démarrent avec une liste de mots-clés, parfois conservée dans un fichier texte (*Expert knowledge*). Ils ont aussi des connaissances additionnelles au sujet des liens unissant ces mots-clés (par exemple, ils savent qu'une maladie particulière est associée à tel ou tel symptôme). En tenant compte de tout cela, ils créent (*Create*) des requêtes (*Query*) qu'ils lancent sur le *Web* (*Launch*). Ils obtiennent ainsi une liste de résultats (*Results*) dont l'analyse leur permet d'extraire (*Extract*) de nouvelles connaissances (*New insights*), comme la suspicion de l'émergence d'une nouvelle maladie. Parfois, ces nouvelles connaissances peuvent aider à raffiner la requête (*Update*) ou à enrichir la liste de mots-clés initiale (*Enrich*), par exemple en associant une nouvelle espèce à une maladie. Enfin, des collègues ou des méthodes de fouille de textes (*External knowledge*) peuvent aussi les aider à enrichir (*Enrich*) leurs connaissances initiales.

Le principal inconvénient de cette pratique est le manque d'une plateforme unique permettant de gérer toutes les étapes du processus d'extraction. Le manque d'outil pour stocker les mots-clés et leurs relations peut induire une perte d'information. Les experts n'ont pas de solution pour gérer les connaissances externes. Les requêtes qu'ils créent sont pauvres et ils peuvent oublier d'y inclure des mots-clés importants. Ils ne peuvent pas sauvegarder facilement les résultats et les retrouver plus tard. C'est pourquoi nous avons proposé *EpidVis*, une plateforme intégrant des fonctionnalités permettant d'accomplir l'ensemble des étapes du processus décrit ci-dessus.

5. https://www.oie.int/fr/ [dernière consultation le 23/03/2020]
6. https://www.plateforme-esa.fr [dernière consultation le 23/03/2020]
7. https://youtu.be/DixKWcIDDXs [dernière consultation le 23/03/2020]
8. Il existe de nombreux outils de visualisation permettant d'observer des données épidémiologiques [34]. Les approches peuvent être basées sur des cartes [41, 61, 29, 48] ou des graphiques [149, 135]. Cependant, ces outils traitent de la visualisation de données déjà récoltées et non de la récolte des données. Il existe aussi de nombreux outils visuels de construction de requêtes [163, 83, 165, 50, 20, 23, 180, 172, 141, 99, 195, 43] mais aucun n'est directement applicable aux besoins des épidémiologistes traités dans cette section.

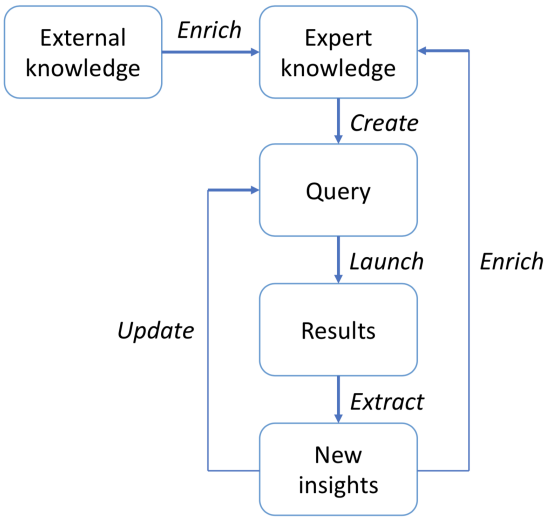


FIGURE 3.7 – Processus d'extraction de connaissances suivi par les experts en épidémiologie.

3.2.2 Abstraction des tâches

Plusieurs entretiens avec les experts nous ont permis de formuler quatre types de tâches :
[T1] Gestion des mots-clés. Les experts veulent pouvoir visualiser, ajouter, supprimer, sauvegarder leurs mots-clés. De plus, il veulent aussi pouvoir créer à partir de ces mots-clés des requêtes pour les lancer sur des moteurs de recherche comme *Google*, *Yahoo* ou *Bing*.
[T2] Gestion des relations entre les mots-clés. Les discussions avec les experts ont montré l'importance de prendre en compte les relations, qu'elles soient faibles ou fortes, entre les mots-clés. Par exemple, la grippe aviaire infecte principalement les oiseaux, ces deux mots-clés entretiennent donc une relation forte. Cependant, la grippe aviaire peut aussi parfois infecter les humains, les chevaux ou les cochons, ce qui induit des relations plus faibles. Les experts veulent donc pouvoir visualiser, ajouter, supprimer, sauvegarder les relations entre les mots-clés ainsi qu'en tenir compte pour la construction de requêtes.
[T3] Intégration de connaissances externes. Les experts ont besoin d'intégrer à leurs ensembles de mots-clés et de relations des connaissances additionnelles issues de sources externes. Ces sources peuvent provenir de collègues ou de méthodes de fouille de textes [115, 10]. Ils veulent donc pouvoir charger ces fichiers, les comparer avec leurs données et sélectionner les éléments qu'ils veulent récupérer.
[T4] Gestion des résultats des requêtes. Les experts veulent visualiser les résultats retournés par les moteurs de recherche, les ordonner/filtrer en fonction des mots-clés qu'ils contiennent et les sauvegarder. Ils désirent aussi les utiliser pour raffiner leurs requêtes dynamiquement.

3.2.3 Abstraction des données

Les mots-clés manipulés par les experts se répartissent en trois catégories : *maladies*, *hôtes* et *symptômes*. Ces mots-clés sont liés entre eux par des relations. Par exemple, une maladie peut provoquer certains symptômes et contaminer certains hôtes. Nous structurons cet ensemble sous la forme d'un graphe tripartite, dans lequel chaque partie correspond à une catégorie, chaque sommet à un mot-clé et chaque arête à une relation entre deux mots-clés issus de catégories différentes.

3.2.4 Conception des idiomes

La figure 3.8 montre une vue d'ensemble d'*EpidVis*. Notre idiome est basé sur une approche multi-vues juxtaposées (*Facet -> Juxtapose* dans le résumé de l'espace de conception schématisé

de la figure 1.3). Cependant, contrairement à *HydroQual*, les vues sont gérées par un système de fenêtres. On peut donc ouvrir plusieurs fenêtres contenant la même vue mais avec des données différentes. De plus, ces vues ne sont pas indépendantes. Par exemple, une vue dédiée aux mots-clés permet de lancer une vue dédiée à la construction des requêtes. Cette dernière ne pourrait donc pas être affichée sans la première. Je vais maintenant les détailler une à une.

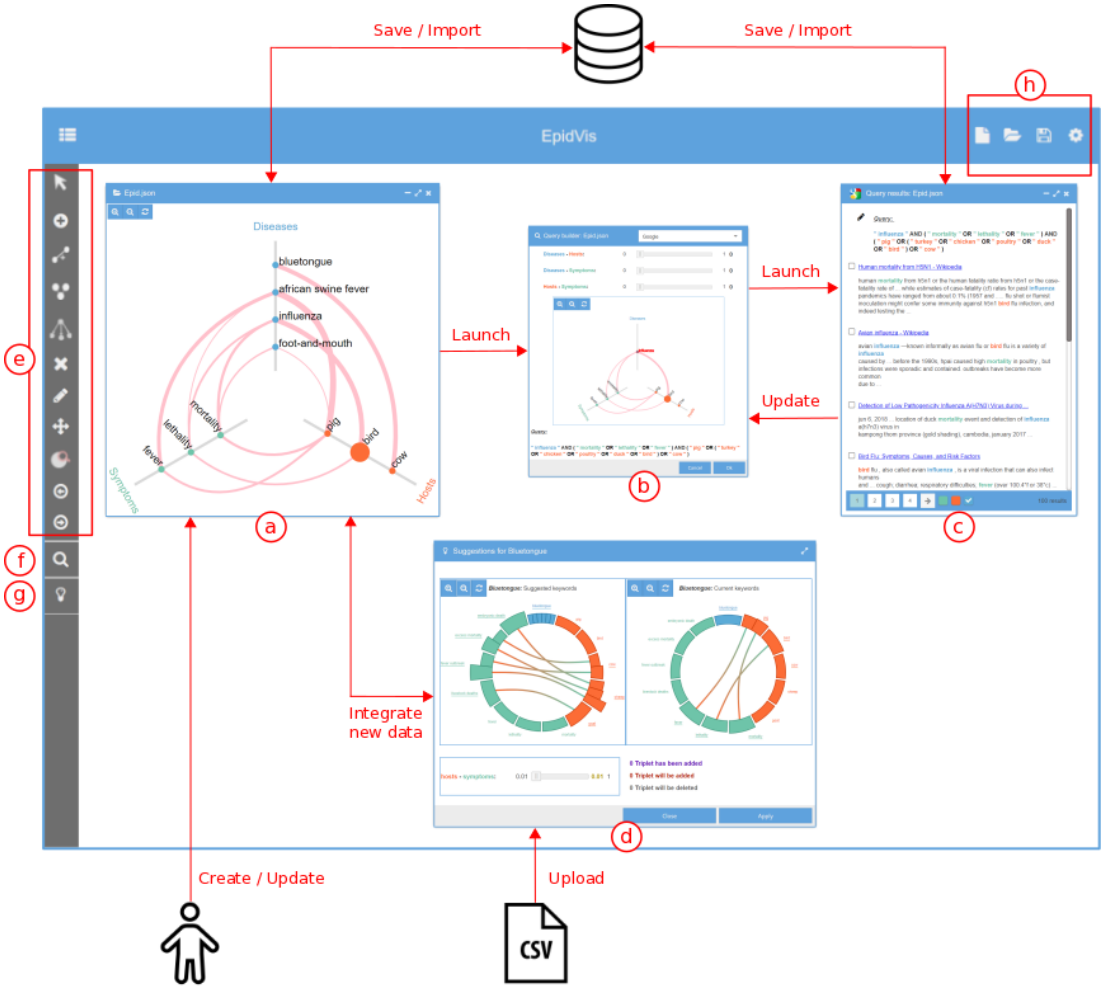


FIGURE 3.8 – *EpidVis*

Le gestionnaire de mots-clés permet d'ajouter, éditer, supprimer, sauvegarder et charger un ensemble de mots-clés [T1] et de relations entre ces mots-clés [T2]. La figure 3.8(a) montre un exemple de cette vue et les figures 3.8(e) et 3.8(h) montrent les boites à outils associées. Le graphe tripartite correspondant au jeu de données est représenté sous la forme d'un *hive plot* [100]. L'intérêt de cette approche est de clairement différencier les catégories des mots-clés (chaque axe représentant une catégorie), d'afficher les différents types de liaisons sur des portions différentes du plan et de ne pas établir de hiérarchie implicite entre les catégories grâce à l'approche radiale. Les mots-clé sont répartis uniformément le long des axes. Afin de limiter leur nombre, l'utilisateur peut en fusionner (il peut par exemple vouloir fusionner *cochon*, *porc* et *pig*, car ces trois mots renvoient aux mêmes entités). La taille des sommets représente alors le nombre de mots-clés fusionnés. La taille des arêtes représente le poids des liens. On notera ici que les couleurs associées à chaque catégorie (bleu pour *maladie*, orange pour *hôte* et vert pour *symptôme*) ont été choisies de façon à être distinguables par des personnes daltoniennes [9]. De plus, leur luminosité est équivalente pour ne pas induire de hiérarchie. Ces trois couleurs se

9. https://projects.susielu.com/viz-palette avec les couleurs #6ec4a9, #fc6c36 et #5babd7 [dernière consultation le 23/03/2020]

retrouvent dans toutes les vues. À partir de cette vue, l'utilisateur peut sélectionner un mot-clé et lancer le constructeur de requêtes (figure 3.8(f)).

Le constructeur de requêtes permet, à partir du mot-clé sélectionné dans le gestionnaire de mots-clés, de paramétrer la requête et de la lancer sur différents moteurs de recherche [T1,T2]. La figure 3.8(b) en montre un exemple. La requête initialement proposée contient le mot-clé sélectionné, ainsi que tous ses voisins connectés grâce à des opérateurs logiques comme le montre la figure 3.9. La vue se divise en trois parties : (1) les ascenseurs en haut permettent de filtrer les relations (et par conséquent les mots-clés) en fonction de leur poids (voir le détail dans notre article [57]), (2) le *hive plot* permet de voir les mots-clés apparaissant dans la requête en fonction des filtres appliquées et (3) la requête correspondante est affichée en bas de la vue. Le code couleur est le même que celui employé dans le gestionnaire de mots-clés. Une fois la requête définie, l'utilisateur peut choisir un moteur de recherche (liste déroulante dans la zone bleue) et la lancer sur ce moteur.

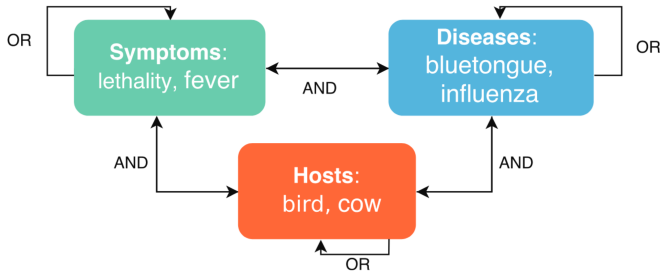


FIGURE 3.9 – Opérateurs logiques utilisés pour construire la requête.

La vue des résultats montre la liste des pages retournées par le moteur de recherche [T4]. Comme le montre la figure 3.8(c), la requête est affichée au-dessus de cette liste et peut être éditée à tout moment, auquel cas l'utilisateur est renvoyé vers le constructeur de requêtes avec les paramètres précédemment sélectionnés. Chaque item de la liste est composé d'un lien hypertexte renvoyant vers la page *Web* et du *snippet* de la page dans lequel les mots-clés sont mis en surbrillance. L'utilisateur peut réordonner les résultats, les sauvegarder ou en charger de nouveaux (voir la figure 3.8(h)).

La vue des suggestions permet d'importer un fichier de données externe et ajouter les éléments qu'il contient, après validation manuelle, au gestionnaire de mots-clés [T3] (voir la figure 3.8(d)). Le bouton de la figure 3.8(g) permet d'y accéder à partir du gestionnaire de mots-clés après avoir sélectionné dans ce dernier un sommet. Elle se compose de deux représentations. La première, à gauche, montre les données du fichier externe et la seconde, à droite, les données du gestionnaire de mots-clés liées au sommet sélectionné. La juxtaposition de ces deux représentations permet à l'utilisateur de comparer les deux jeux de données [10]. Elles sont synchronisées et l'utilisateur peut facilement sélectionner des ensembles de mots-clés et de relations et observer l'impact de leur ajout sur le gestionnaire des mots-clés (voir le détail dans notre article [57]).

3.2.5 Validation

Comme pour *HydroQual*, la validation de notre approche porte essentiellement sur l'évaluation des abstractions des données et des tâches. Cependant, afin de vérifier qu'*EpidVis* ne nécessite pas un temps d'apprentissage trop élevé et que ses fonctionnalités sont utiles et intuitives, nous avons aussi mené une étude d'utilisabilité (voir la section 1.2.3). Pour rappel, ce

10. Il existe trois stratégies permettant de comparer deux jeux de données [70, 69] : juxtaposition, superposition et encodage explicite des différences. Notre visualisation combine les deux premières : (1) deux représentations juxtaposées montrent les deux jeux de données et (2) l'utilisateur peut sélectionner des éléments dans la représentation de gauche et visualiser l'impact de leur ajout dans la représentation de droite (superposition).

type d'étude relève de la conception des idiomes. Je ne la détaillerai pas ici car elle est marginale vi-à-vis de mon propos, mais j'invite le lecteur intéressé à se reporter à notre article [57].

Pour valider les abstractions des données et des tâches, un expert a utilisé notre outil afin de découvrir de nouvelles informations. Je vais me contenter ici de résumer sa démarche et d'en présenter les principaux résultats, mais le détail est disponible dans l'article.

Dans un premier temps, l'expert a créé et lancé deux requêtes simples ne contenant que le nom d'une maladie chacune : *african swine fever* et sa traduction en français *peste porcine africaine*. Cette première tâche correspond à ce qui est généralement accompli en utilisant un moteur de recherche classique et a pu donc lui servir de référence. Parmi les 10 premiers résultats obtenus sur chaque requête, 8 étaient pertinents, d'après son propre jugement, pour la première et 10 pour la seconde.

Dans un deuxième temps, l'expert a utilisé la fonction de fusion des mots-clés pour créer rapidement une requête comprenant le nom de la maladie dans différentes langues : français car il y a eu des cas de la maladie en Belgique (*"peste porcine africaine"*), polonais car il y en a eu aussi en Pologne (*"afrykańskiego pomoru świń"*) et allemand car la Pologne possède une frontière avec l'Allemagne (*"Afrikanische Schweinepest"*). Les 10 résultats étaient pertinents : 7 étaient communs avec la première requête en français, mais 3 nouveaux (2 en allemand et 1 en polonais) contenaient des informations supplémentaires à celles déjà trouvées précédemment. L'expert a ici apprécié la possibilité de sauvegarder son jeu de données afin de pouvoir relancer cette requête rapidement en quelques clics plus tard.

Dans un troisième temps, l'expert est reparti du mot-clé *african swine fever* en ajoutant à son jeu de données des *hôtes* et des *symptômes* en lien avec lui (*pig*, *wild boar*, *porcine* pour les *hôtes* et *mortality*, *haemorrhagic*, *fever* pour les *symptômes*). Cet ajout lui a permis de trouver 4 résultats pertinents supplémentaires. Ici encore, grâce à la possibilité de sauvegarder son jeu de données, l'expert pourra désormais relancer ce type de requêtes en quelques clics.

Enfin, dans un quatrième temps, l'expert a chargé des *symptômes* et les *hôtes* contenus dans un fichier externe obtenu à l'aide d'une méthode de fouille de textes proposée par Arsevska *et al.* [10]. Il a sélectionné ceux dont les poids étaient les plus importants et les a ajouté à son ensemble de mots-clés et de relations. En relançant une requête à partir de ce nouveau jeu de données, il a pu trouver trois articles pertinents supplémentaires.

3.2.6 Synthèse

EpidVis a été conçu dans le cadre d'un projet de plus grande envergure portant sur l'épidémiologie animale. Ce projet comprenait des épidémiologistes ainsi que des informaticiens spécialistes en TAL [11] et en visualisation. Le travail présenté ici constitue la première étape du système d'intelligence épidémique que nous avons mis en place. Grâce à lui, on peut extraire des pages *Web*. Ensuite, à partir de ces pages, les informaticiens en TAL ont mis au point une technique permettant d'extraire automatiquement les *maladies*, les *hôtes*, les *symptômes* ainsi que des données spatiales et temporelles [9]. Pour finir, nous avons proposé une solution visuelle, *EpidNews* permettant d'explorer l'ensemble de ces dimensions, et de comparer les données ainsi récoltées avec les sources officielles telles que celles fournies par l'OIE [71, 72] (voir la figure 3.10).

Outre mes travaux liés à l'épidémiologie animale, je me suis également intéressé à la santé humaine. Nous avons par exemple proposé une technique permettant de découvrir des cellules cancéreuses rares [169]. Nous avons aussi proposé une méthode permettant d'analyser les mouvements de patients utilisant un jeu sérieux de rééducation [146]. Enfin, nous avons mis au point un outil permettant d'aider au diagnostique de l'évolution de patients souffrant d'héminégligence [150].

11. Traitement Automatique du Langage

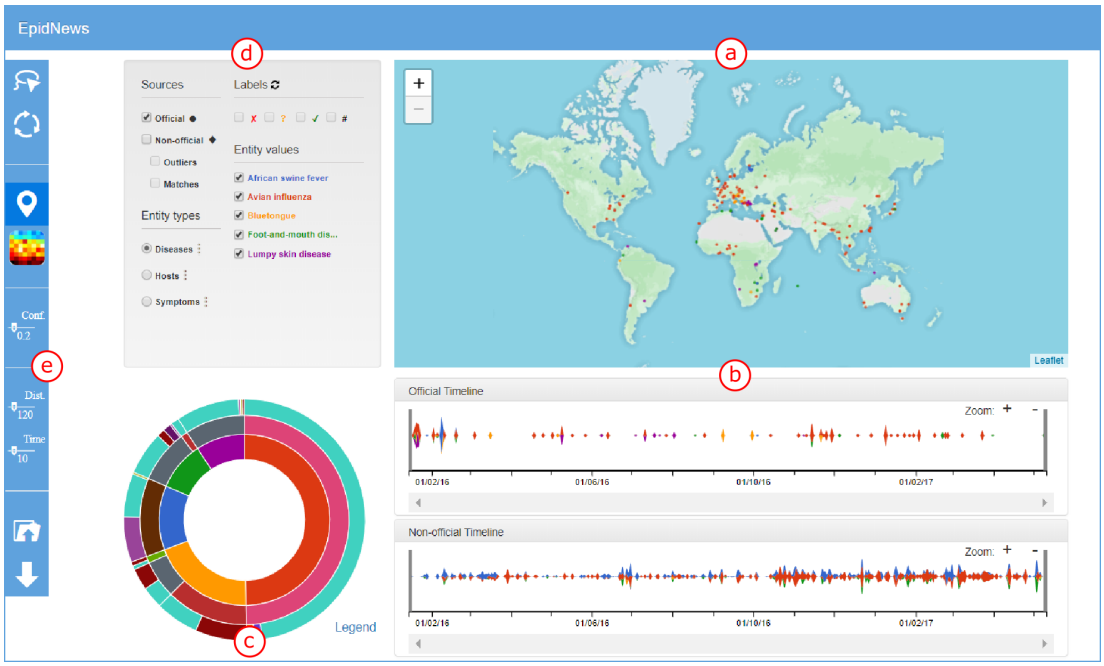


FIGURE 3.10 – *EpidNews*, un outil permettant d'explorer les articles portant sur les épidémies : (a) géolocalisation des articles, (b) évolution du nombre d'articles officiels et non officiels, (c) proportion des articles par maladies, hôtes et symptômes, (d) filtres, (e) boite à outils.

Chapitre 4

Perspectives

Dans cette section, je vais présenter les différents axes que je prévois de développer dans mes futurs travaux. Le premier se situe dans le prolongement des travaux précédents et concerne la conception de nouvelles techniques de visualisation et de solutions visuelles pour des données relationnelles, spatiales et temporelles. Le deuxième axe concerne le couplage de la visualisation avec de la classification supervisée réalisée à l'aide d'un modèle d'apprentissage profond. J'étudierai comment un tel modèle peut aider à évaluer des techniques de visualisation et comment des solutions visuelles peuvent également aider à mieux interpréter une classification réalisée à l'aide de ce modèle. Enfin, le troisième axe concerne un autre type de travail que ceux menés jusqu'à présent : l'étude empirique (voir la section 1.3). Plus particulièrement, il s'agit de mener des expérimentations contrôlées afin de comparer des techniques de visualisation et de s'intéresser aux résultats issus des sciences cognitives afin de voir quelles en sont les applications en visualisation.

4.1 Visualisation de données relationnelles, spatiales, temporelles

Comme nous avons pu le voir dans les chapitres précédents, le plupart de mes travaux ont porté sur la visualisation de données relationnelles (*e.g. visualisation de graphes dynamiques*, *comparaison d'algorithmes de suppression de chevauchements*, *EpidVis*), spatiales (*e.g. HydroQual*) et temporelles (*e.g. MultiStream*, *visualisation de graphes dynamiques*). Cette section contient d'autres pistes liées à ces types de données.

4.1.1 Techniques de visualisation

Comme le montrent un récent état de l'art [122] et un séminaire à Dagstuhl auquel j'ai été invité [95], de nombreux problèmes liés à la visualisation de réseaux multi-couches restent ouverts. À titre d'exemple, nous sommes actuellement en train de finaliser un outil interactif permettant de rechercher un motif dans un tel graphe et de visualiser les occurrences de ce motif. Comme *EpidNews*, l'outil permet de construire visuellement une requête. Nous finalisons actuellement les idiomes permettant d'explorer les résultats. Une version préliminaire, montrant l'intérêt de ce type d'approche, a déjà fait l'objet d'une démonstration dans une conférence en bases de données en 2018 [43].

Des travaux futurs nous amèneront à aborder le problème de la représentation holistique des graphes multicouches, *i.e.* un idiome montrant une vue d'ensemble des éléments du graphe et de ses couches. Nous étudierons aussi les méthodes permettant de passer de cette vue à des représentations détaillées.

4.1.2 Conception de solutions visuelles

La conception de solutions visuelles passe par la définition d'un problème lié à un domaine d'application. Dans les années à venir, je prévois de continuer à travailler dans le domaine de la santé, et plus particulièrement dans le domaine de la santé humaine. Dans ce cadre, nous

venons d'obtenir un projet H2020 MOOD [1] dans lequel nous prévoyons de développer des solutions permettant de suivre l'évolution de maladies dans le temps et l'espace, soit issue de données réelles, soit issue de modèles de propagation.

En accord avec la stratégie de développement des humanités numériques mise en place par l'université Paul-Valéry, je prévois également de m'intéresser à des données historiques. Dans le cadre de l'ANR Daphne [2], nous sommes en train de concevoir une solution visuelle permettant d'explorer des bases de données prosopographiques. Elles sont constituées de descriptions d'individus historiques incluant leurs parcours, leurs diplômes, leurs emplois, *etc.*

Que ce soit dans le cas des données épidémiologiques ou dans le cas des données historiques, les mêmes défis émergent. En effet, nous sommes confrontés aux problématiques liées à la visualisation de grands ensemble de données multidimensionnelles pouvant être modélisées de différentes façon. Dans ce cadre, l'abstraction de données sous forme de graphes multivariés [93, 139] semble prometteuse : graphe de dissémination de maladie dans le premier cas, graphe des parcours des individus dans le second. Les attributs spatio-temporels constitueront des caractéristiques communes à ces deux graphes et nous devrons donc : (1) étudier les abstractions des tâches selon les domaines d'applications, si possible en les faisant converger, et (2) proposer des idiomes selon ces abstractions [3].

4.2 Visualisation et apprentissage profond

L'émergence du domaine de la visualisation analytique [173, 92] (*Visual Analytics*) a entraîné une multiplication des outils combinant des techniques d'analyse automatique des données avec des visualisations interactives. En particulier les méthodes d'apprentissage ont fait l'objet d'un grand nombre de travaux [51]. Certaines de mes contributions s'inscrivent d'ailleurs dans ce cadre en combinant des méthodes de représentation avec des techniques de classification non supervisée (*clustering*) [158, 155, 1, 146].

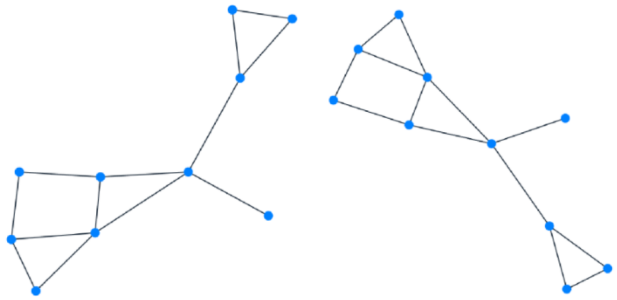
Les perspectives décrites ici concernent le couplage de la visualisation avec de la classification supervisée cette fois, réalisée à l'aide d'un modèle d'apprentissage profond (*deep learning*). Je vais d'abord aborder l'utilisation de l'apprentissage pour aider le développement de techniques de visualisation. Ensuite, je présenterai des pistes afin de mettre en place des solutions visuelles pour l'interprétation de modèles d'apprentissage. Quelques notions d'apprentissage profond sont nécessaires pour comprendre les enjeux décrits dans cette section et les présenter déborderait le cadre de ce manuscrit. J'invite le lecteur intéressé à se référer à l'article de Garcia *et al.* [66] pour une introduction à ces méthodes dans le cadre de la visualisation ou au livre [73] [4] pour une vue d'ensemble.

4.2.1 Améliorer les techniques de visualisation grâce à l'apprentissage

Récemment, un travail original portant sur l'utilisation d'une technique d'apprentissage profond pour la visualisation a été proposé par De Luca *et al.* [116]. Ce travail fait suite à un précédent article dans lequel les auteurs montraient que les deux mesures de détection de symétries dans les dessins de graphe (K [96] et P [145]) pouvaient parfois donner des résultats surprenants [190] (voir la figure 4.1). Or, la mise en évidence de symétries constitue un critère esthétique recherché et pris en compte dans l'évaluation des algorithmes de dessin de graphes [145]. Il est donc fondamental d'arriver à l'évaluer afin de comparer les différentes approches. Dans ce contexte, De Luca *et al.* ont proposé d'utiliser une technique d'apprentissage profond pour classer les dessins comme symétriques ou non. Celle-ci a été comparée avec les deux mesures précédemment utilisées. Cette première contribution ouvre la porte à de

1. https://www.moodspatialdata.com/ [dernière consultation le 27/03/2020]
2. http://daphne.huma-num.fr/ [dernière consultation le 27/03/2020]
3. Il existe un certain nombre de techniques permettant de visualiser des données relationnelles spatialisées [160] (https://geographic-networks.github.io/ [dernière consultation le 26/03/2020]). Une autre source d'inspiration peut également venir des outils de visualisation analytique dédiés aux mouvements [4].
4. https://www.deeplearningbook.org/ [dernière consultation le 27/03/2020]

nombreuses perspectives que je désire explorer. Les résultats de l'étude peuvent-ils être améliorés avec un autre modèle d'apprentissage ? Les dessins étant rarement complètement symétriques, peut-on proposer une approche multi-classes afin d'évaluer un "degré de symétrie" (ce que font les mesures précédentes) ? Sachant, par exemple, que P a une complexité temporelle en $O(n^7)$, peut-on aussi proposer une approche multi-classes permettant d'approximer cette mesure ou bien celle de K ? Est-il possible de généraliser cette approche pour déterminer l'esthétisme ou la qualité des dessins de graphes (ou éventuellement d'autre types de visualisations) au-delà de la symétrie ?



(a) $P = 0.397$ et $K = 0.615$ (b) $P = 1$ et $K = 0.077$

FIGURE 4.1 – Deux dessins quasiment identiques (modulo une rotation et une réflexion) mais obtenant des scores de symétrie K [96] et P [145] très différents : (a) Le dessin obtient un score K supérieur à son score P. (b) Le dessin obtient un très bon score P et un très mauvais score K [190].

4.2.2 Concevoir des solutions visuelles pour l'interprétation de modèles d'apprentissage

Contrairement à la section précédente qui se focalisait sur l'apprentissage profond pour la visualisation, cette section traite de la visualisation pour l'apprentissage profond. Ce domaine est en plein expansion comme le montre la variété des contributions. L'étude très complète réalisée par Hohman *et al.* [84] en 2018 donne un aperçu de ces contributions classées selon 6 axes (voir la figure 4.2). Par exemple, un outil visuel pour interpréter les représentations apprises par un modèle profond (*Pourquoi/Why*) permet à un développeur de modèle (*Qui/Who*) de visualiser les activations des neurones dans un réseau de convolution (*Quoi/What*) grâce à un positionnement t-SNE (*Comment/How*) après une phase d'entraînnement (*Quand/When*) afin de résoudre un problème de planification urbaine (*Où/Where*).

Comme le remarquent Hohman *et al.* en introduction de leur article [84], la visualisation peut : (1) aider les concepteurs de modèles profond à comprendre et améliorer leur approche et (2) aider les utilisateurs finaux à juger de la fiabilité des résultats en vue de les prendre en compte dans leur activité de prise de décision. Dans le premier cas, il s'agit de comprendre l'architecture (*architecture understanding*), analyser les résultats de la phase d'entraînement (*training analysis*) et comprendre les caractéristiques des données prises en compte par le modèle (*feature understanding*) [66] [5]. Les concepteurs doivent donc pouvoir *interpréter* et *expliquer* les résultats [6]. Au contraire, dans le second cas, l'objectif est principalement d'*interpréter les résultats*. Lorsque l'on regarde le tableau récapitulatif des techniques répertoriées par Hohman *et al.*, on peut observer que seules 10 publications parmi les 38 décrites concernent ce cas (*Who -> Non experts* dans la figure 4.2). C'est donc dans cet axe que je souhaite développer mes travaux.

A titre d'exemple, dans le cadre de la thèse d'Alexis Delaforge démarrée en septembre 2019, et dont je suis l'un des co-encadrants, nous nous intéressons à la classification de textes en

5. Liu *et al.* [112] et Garcia *et al.* [66] offrent un panorama des techniques disponibles pour réaliser ces opérations.
6. Voir Chatzimparmpas *et al.* [35] pour une discussion sur la définition de ces deux termes dans le contexte de l'apprentissage profond.

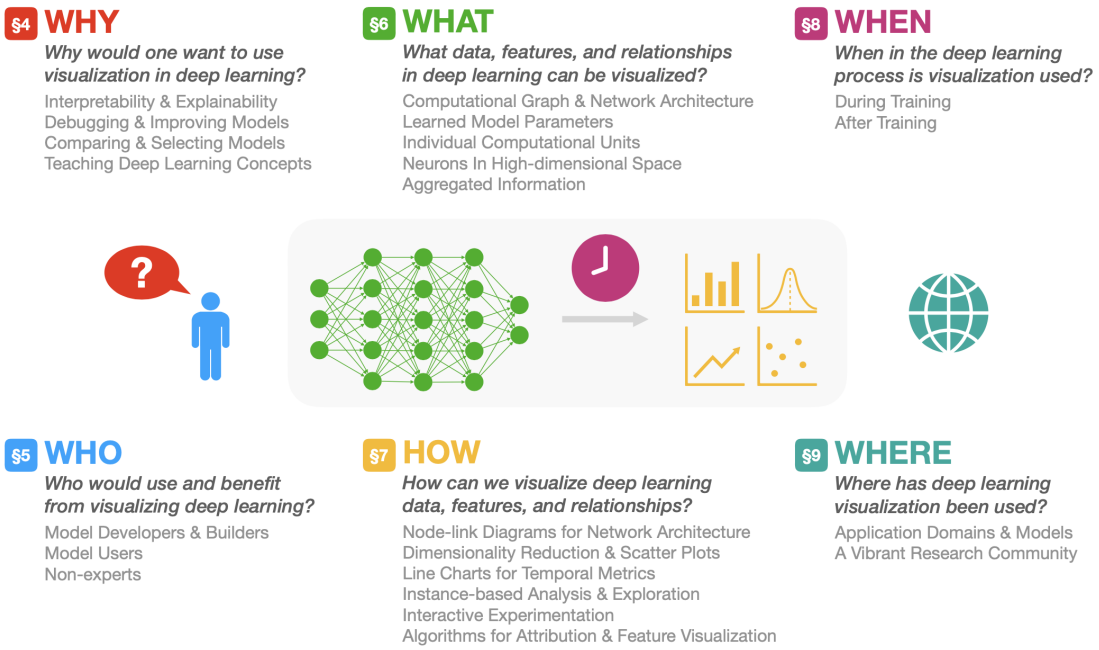


FIGURE 4.2 – Cinq axes de classification des outils de visualisation pour l'apprentissage profond [84].

utilisant des méthodes d'apprentissage profond [7]. On retrouve cette problématique parmi les principales identifiées dans l'état de l'art (*survey of surveys*) proposé par Chatzimparmpas *et al.* [35] (*topic 10*). Plus précisément, notre objectif est de permettre à des utilisateurs finaux (*Qui/Who*) d'interpréter les résultats d'une classification binaire (*Pourquoi/Why*) obtenus après une phase d'entraînements (*Quand/When*). Notre travail vise donc à définir ce que nous allons représenter (*Quoi/What*) et à concevoir la solution visuelle permettant de le représenter (*Comment/How*). L'idée que nous explorons actuellement consiste à visualiser les données d'entrée dans l'espace de représentation issu de la dernière couche avant le classifieur. Habituellement, des méthodes de réduction dimensionnelle, *e.g.* t-SNE [117, 147], sont utilisées mais elles ne permettent pas d'identifier la proximité des phrases avec la frontière séparant les deux classes dans l'espace multidimensionnel. C'est ce à quoi nous travaillons car nous pensons que cela aidera un utilisateur final à comprendre la classification et ses limites.

Ce travail s'inscrit dans différents défis mis en évidence par Hohman *et al.* [84], principalement améliorer l'interprétabilité des réseaux neuronaux et concevoir une solution scalable [113]. Il s'inscrit aussi dans les défis de conception de visualisations porteuses de sens, de l'amélioration de la confiance envers les résultats, de l'incertitude et de la prise en compte des connaissances des utilisateurs (néophytes dans notre cas) soulevés par Chatzimparmpas *et al.* [35]. Mes travaux futurs dans ce domaine viseront à approfondir les recherches portant sur ces axes.

4.3 Études empiriques

En introduction, nous avons vu comment chaque étape de conception du modèle imbriqué peut être évalué. En particulier, nous avons vu quels étaient les principales méthodes d'évaluation : observation des experts, expérimentation contrôlée, analyse qualitative et quantitative des images produites, études de cas, mesure des temps de calcul des algorithmes. Ces méthodes répondent à la question *Comment évaluer une visualisation ?* Cependant, dés 2003, Kosara *et al.* [98] ont mis en évidence la nécessité de se poser deux questions complémentaires :

7. Dans ce cadre, plusieurs techniques de visualisation pour l'apprentissage profond ont déjà été proposées (voir par exemple [166, 137]). D'une façon plus générale, l'état de l'art dans le domaine de la visualisation couplée à des méthodes d'analyse de texte a été présenté par Liu *et al.* [110].

Quoi évaluer ? et *Pourquoi évaluer ?* Ces deux questions reviennent à définir les objectifs d'une évaluation et c'est grâce à ces derniers que l'on peut sélectionner la ou les méthode(s) d'évaluation approprié(s). En 2012, Lam *et al.* [102] ont défini sept objectifs (qu'ils appellent scénarios) regroupés en deux grandes catégories : (1) *analyse de données*, qui comprend les évaluations portant sur la capacité d'un outil visuel à permettre d'analyser les données représentées, et (2) *visualisation*, qui comprend les évaluations portant sur les encodages visuels, les interactions et les algorithmes en eux-mêmes. Le tableau 4.1 montre les sept objectifs, avec en dernière colonne, les références des cinq articles que j'ai détaillés dans les sections 2 et 3 :

Analyse de données	1. Comprendre les habitudes de travail	[1, 57]
	2. Évaluer l'analyse visuelle de données	[42, 155, 1, 57]
	3. Évaluer la communication visuelle	-
	4. Évaluer l'analyse collaborative de données	-
Visualisation	**5. Évaluer la performance utilisateur**	**en perspective**
	6. Évaluer l'expérience utilisateur	[1, 57]
	7. Évaluer les algorithmes	[36]

TABLE 4.1 – Objectifs des évaluations en visualisation d'information.

Une large majorité des contributions en visualisation, quel que soit leur type, contient une ou plusieurs évaluations. Par exemple, une conception de solution visuelle commence généralement par une compréhension des habitudes de travail du domaine d'application (*objectif 1*). Elle finit aussi généralement par une évaluation de la capacité de la visualisation à permettre d'analyser visuellement des données (*objectif 2*) et une évaluation de l'expérience utilisateur (*objectif 6*). C'est ce qui a été réalisé pour *HydroQual* [1] (voir la section 3.1) et *EpidVis* [57] (voir la section 3.2), bien que je n'ai pas détaillé l'expérience utilisateur qui est disponible dans les articles. Dans le modèle imbriqué, cela correspond aux précautions du premier niveau et à la validation des deuxième et troisième niveaux (voir la figure 1.5). On retrouve ici ce qui a été dit en introduction : la conception d'une solution visuelle s'inscrit principalement dans les premiers niveaux du modèle. Les validations de *MultiStream* (voir la section 2.1 [42]) et de la visualisation de graphes dynamiques (voir la section 2.2 [155]) se positionnent aussi sur la capacité de la visualisation à permettre d'analyser visuellement les données (*objectif 2*). Cependant, nous nous sommes arrêtés à l'observation des caractéristiques (*visual features*), et non à l'explication de ces caractéristiques, puisque la contribution portait sur le troisième niveau du modèle imbriqué. Enfin, la validation employée dans la comparaison des algorithmes de suppression de chevauchement (voir la section 2.3 [36]) portait sur l'évaluation des algorithmes (*objectif 7*).

En introduction, nous avons vu qu'il existe un type de contribution portant exclusivement sur les *études empiriques*. En pratique, ce type de publication se focalise essentiellement sur l'*objectif 5*, qui correspond d'ailleurs à la définition que nous avons donné à ce type. En clair, le syntagme *étude empirique* regroupe des évaluations aux objectifs diverses et aux formes variées selon Lam *et al.* [102] alors qu'il est restreint aux *expérimentations contrôlées* en vue d'évaluer la performance utilisateur (*objectif 5*) dans le type de contribution associé.

4.3.1 Expérimentation contrôlée

Un expérimentation contrôlée [8] consiste principalement à comparer les performances en terme de temps et de taux d'erreurs des utilisateurs à accomplir des tâches sur plusieurs idiomes concurrents. Une analyse statistique est ensuite produite pour déterminer quel est l'idiome le plus adapté selon le contexte. Purchase [144] a décrit précisément la procédure à adopter pour mener a bien une telle étude. Elle s'inscrit principalement dans le quatrième niveau du modèle imbriqué (*conception des idiomes*).

8. Dans la littérature, cette notion se rencontre sous différentes terminologies : *controlled experiments*, *quantitative evaluation*, *factorial design experiments* [98].

Dans un premier temps, il faut déterminer un ensemble de techniques à comparer $TECH$, un ensemble de tâches à accomplir par un utilisateur avec ces techniques $TASKS$ et un ensemble de jeux de données sur lesquels réaliser ces tâches DS. Ensuite, il faut produire une série de sessions, chaque session étant composée d'essais, chaque essai consistant à réaliser une tâche particulière sur un jeu de données en employant une technique particulière. L'expérimentation consiste alors à faire effectuer par un ensemble de participants les sessions ainsi produites, puis d'analyser les résultats à l'aide d'outils issus de la statistique (moyenne, médiane, écart-type, intervalles de confiance, analyse de la variance).

Une des principales difficultés consiste à contrebalancer les essais, *i.e.* faire en sorte que chaque essai possible soit réalisé un même nombre de fois par des participants différents. Ceci est d'autant plus important que les jeux de données sont variés et peu nombreux [114]. Dans ce cadre, je travaille actuellement sur un projet avec des chercheurs de l'UMR GRED visant à comparer les différentes techniques utilisées par les spécialistes en aménagement du territoire pour visualiser les zones impactées par un désastre naturel (tsunami, tremblement de terre, *etc.*). Notre protocole se compose de 24 scénarios réalisés à partir de 6 techniques ($TECH$ *A-F*), 4 tâches ($TASK$ *A-D*) et 3 jeux de données (DS 1-3). La figure 4.3 montre un schéma des 4 premières sessions produites de façon à contrebalancer les résultats.

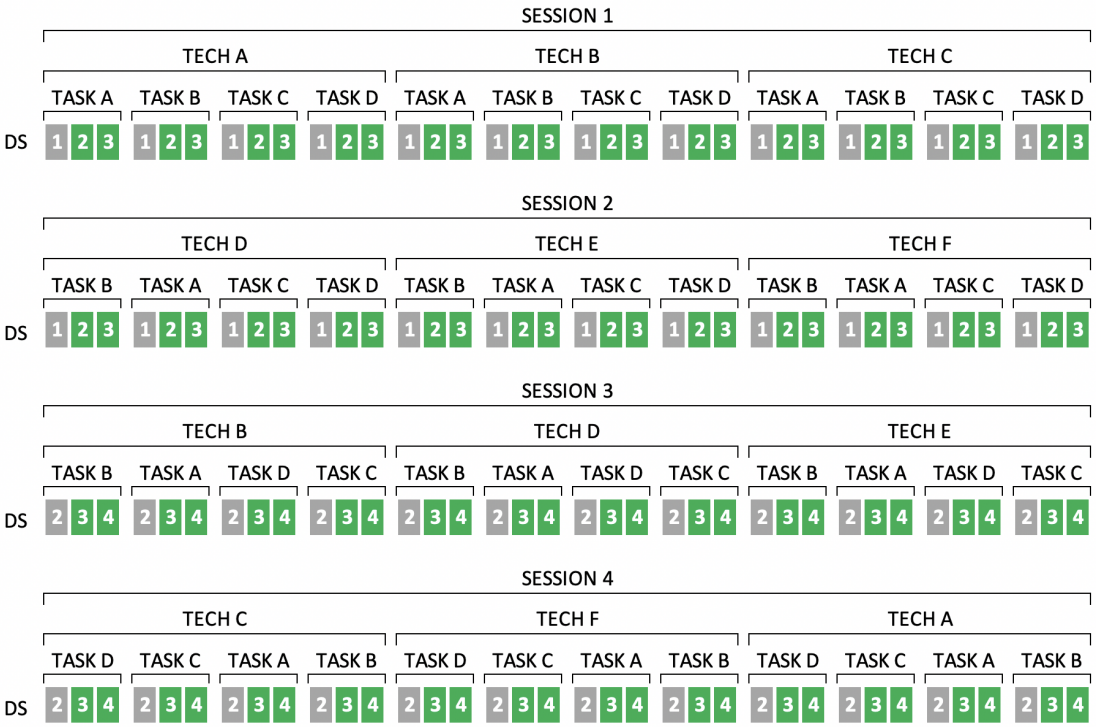


FIGURE 4.3 – Quatre premières sessions d'une expérimentation contrebalancée. Les résultats des essais grisés ne sont pas analysés (entraînement) contrairement aux verts. Les techniques et les tâches sont correctement contrebalancées : chaque technique apparait le même nombre de fois et chaque tâche apparait le même nombre de fois pour chaque technique, même si les ordres sont différents. Il faudra au moins 8 sessions pour contrebalancer les jeux de données.

Mon objectif, dans les prochaines années, est donc d'utiliser l'expérience acquise lors du travail en cours mentionné ici afin de développer ce type d'expérimentations pour d'autres types d'applications.

4.3.2 Visualisation et sciences cognitives

Les expérimentations contrôlées décrites dans la section précédente permettent de déterminer, parmi plusieurs représentations, lesquelles sont les mieux adaptées à l'accomplissement

de certaines tâches. Ce type de recherche possède des liens étroits avec certains domaines de la psychologie expérimentale et notamment avec l'étude de la perception et de la mémoire visuelle. Healey et Enns [81] donnent un aperçu de ce que ces domaines peuvent apporter à la visualisation. En se basant sur une vaste bibliographie issue des sciences cognitives, ils expliquent comment certaines tâches, telles que la détection de cibles, la détection de frontières, le regroupement d'éléments évoluant de façon identique ou la comptabilisation d'éléments, peuvent être détectés de façon pré-attentive, *i.e* dans un délai de temps inférieur à 200-250 millisecondes. Ensuite, ils présentent les différentes théories de la vision préattentive et leurs impacts/défis pour la visualisation. Dans de futurs travaux que nous allons initier avec des membres du laboratoire Epsylon, nous prévoyons d'étudier les implications de théories récentes issues des sciences cognitives. Par exemple, dans l'article [46], les auteurs proposent un modèle pour l'estimation d'une quantité d'objets graphiques en fonction, bien sur, du nombre de ces objets mais aussi en fonction d'autres paramètres, tels que leur taille ou leur espacement. Ceci peut avoir un impact sur la visualisation de nuages de points ou de graphes.

Une loi importante issue de la psychophysique [164] est la loi de Stevens qui stipule que la variation d'intensité perçue S est égale à la variation d'intensité réelle I puissance une constante k dépendante de la nature du stimulus ($S = I^n$). La figure 4.4 montre les valeurs pour différents stimuli liés à la visualisation. On peut y observer que les différences de longueur sont perçues de façon plus correctes que celles engendrées par d'autre stimuli visuels, ce qui permet d'ordonner les variables visuelles lors de la conception d'idiomes. Il serait ici utile de développer ces recherches afin d'affiner les résultats, par exemple dans le cas des aires. Heer et Bostock [82], en s'inspirant des travaux de Cleveland et McGill [39], ont par exemple montré que notre perception des aires de surfaces rectangulaires est moins précise que celle de surfaces circulaires. Ce phénomène implique-t-il que la constante k est différente selon les formes des aires ? Qu'en est-il des autres formes ? De tels résultats pourraient avoir de fortes implications en visualisation, par exemple en déterminant quel type de *treemap*[9] (*slice and dice*, *squarrified*, *Voronoi*, *etc.*) permet au mieux d'évaluer les valeurs qu'il représente.

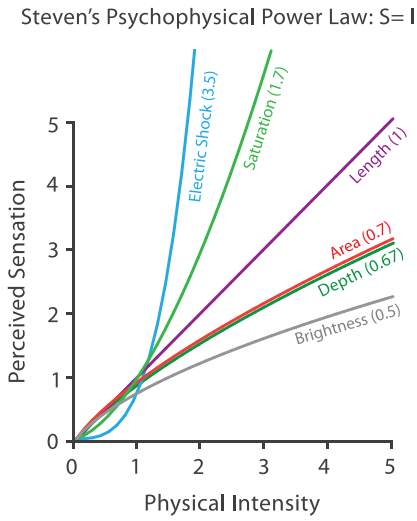


FIGURE 4.4 – Loi de Stevens en fonction de différents stimuli visuels [133], adapté de [164].

9. Un état de l'art de la visualisation d'arbres, incluant les différentes formes de *treemap*, est disponibles à cette adresse : https://treevis.net/ [159].

Bibliographie

[1] Pierre ACCORSI et al. « HydroQual : Visual analysis of river water quality ». In : *Proceeding of the IEEE Conference on Visual Analytics Science and Technology (VAST)*. 2014, p. 123-132.

[2] AFNOR. *Qualité écologique des milieux aquatiques - Détermination de l'indice biologique global normalisé (IBGN)*. Rapp. tech. NF T90-350. Association francaise de normalisation, 2004.

[3] Wolfgang AIGNER et al. *Visualization of Time-Oriented Data*. Springer, 2011.

[4] Gennady L. ANDRIENKO et al. *Visual Analytics of Movement*. Springer, 2013.

[5] Paola ANTONELLI. *Talk to Me : Design and Communication between Peaople and Objects*. The Museum of Modern Art (MoMA), 2011.

[6] Virginie ARCHAIMBAULT, Philippe USSEGLIO-POLATERA et Jean-PierreVanden BOSSCHE. « Functional Differences Among Benthic Macroinvertebrate Communities in Reference Streams of Same Order in a Given Biogeographic Area ». In : *Hydrobiologia* 551.1 (2005), p. 171-182.

[7] Virginie ARCHAIMBAULT et al. « Assessing pollution of toxic sediment in streams using bio-ecological traits of benthic macroinvertebrates ». In : *Freshwater Biology* 55.7 (2010), p. 1430-1446.

[8] Daniel W. ARCHAMBAULT, Helen C. PURCHASE et Bruno PINAUD. « Animation, Small Multiples, and the Effect of Mental Map Preservation in Dynamic Graphs ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 17.4 (2011), p. 539-552.

[9] E. ARSEVSKA et al. « Web Monitoring of Emerging Animal Infectious Diseases Integrated in the French Animal Health Epidemic Intelligence System ». In : *PLoS ONE* 13.8 (2018), e0199960.

[10] Elena ARSEVSKA et al. « Identification of Associations between Clinical Signs and Hosts to Monitor the Web for Detection of Animal Disease Outbreaks ». In : *International Journal of Agricultural and Environmental Information Systems* 7.3 (2016), p. 1-20.

[11] David AUBER et al. « GosperMap : Using a Gosper Curve for Laying Out Hierarchical Data ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 19.11 (2013), p. 1820-1832.

[12] Richard AXLER et al. *Minnesota lake water quality on-line database and visualization tools for exploratory trend analyses*. Rapp. tech. NRRI/TR-2009/28. University of Minnesota Duluth, 2009.

[13] Albert-László BARABAŚI et Réka ALBERT. « Emergence of scaling in random networks ». In : *Science* 286.5439 (1999), p. 509-512.

[14] Dominikus BAUR, Bongshin LEE et Sheelagh CARPENDALE. « TouchWave : Kinetic Multi-touch Manipulation for Hierarchical Stacked Graphs ». In : *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS)*. 2012, p. 255-264.

[15] Michel BEAUDOUIN-LAFON. « Designing interaction, not interfaces ». In : *Proceeding of the Conference on Advanced Visual Interfaces (AVI)*. 2004, p. 15-22.

[16] Fabian BECK et al. « A Taxonomy and Survey of Dynamic Graph Visualization ». In : *Computer Graphics Forum* 36.1 (2017), p. 133-159.

[17] Stephen T. BENEDICT et al. *Data Visualization, Time-Series Analysis, and Mass-Balance Modeling of Hydrologic and Water-Quality Data for the McTier Creek Watershed, South Carolina, 2007–2009*. Rapp. tech. Open-File Report 2011–1209. U.S. Geological Survey, 2012.

[18] Lior BERRY et Tamara MUNZNER. « Binx : Dynamic Exploration of Time Series Datasets Across Aggregation Levels ». In : *Proceedings of the IEEE Symposium on Information Visualization (InfoVis)*. 2004.

[19] Jacques BERTIN. *Sémiologie Graphique. Les diagrammes, les réseaux, les cartes*. EHESS, 3e édition, (1e éd. Gauthier-Villars 1967, 2e éd. 1973) 1999.

[20] Sourav S. BHOWMICK, Byron CHOI et Shuigeng ZHOU. « VOGUE : Towards a Visual Interaction-aware Graph Query Processing Framework ». In : *Proceedings of the Conference on Innovative Data Systems Research (CIDR)*. 2013, p. 1-10.

[21] Gilles BILLEN et al. « The Seine system : Introduction to a multidisciplinary approach of the functioning of a regional river system ». In : *Science of The Total Environment* 375.1–3 (2007), p. 1-12.

[22] Vincent D. BLONDEL et al. « Fast unfolding of communities in large networks ». In : *Journal of Statistical Mechanics : Theory and Experiment* (2008).

[23] Harald BOSCH et al. « ScatterBlogs2 : Real-Time Monitoring of Microblog Messages through User-Guided Filtering ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 19.12 (2013), p. 2022-2031.

[24] Romain BOURQUI et al. « Multilayer graph edge bundling ». In : *Proceeding of the IEEE Pacific Visualization Symposium (PacificVis)*. 2016, p. 184-188.

[25] Joseph N. BOYER, Peter STERLING et Ronald D. JONES. « Maximizing Information from a Water Quality Monitoring Network through Visualization Techniques ». In : *Estuarine, Coastal and Shelf Science* 50.1 (2000), p. 39-48.

[26] Ulrik BRANDES. « Drawing on physical analogies ». In : *Drawing Graphs : Methods and Models*. Sous la dir. de M. KAUFMANN et D. WAGNER. Springer. T. 2025. Lecture Notes in Computer Science. 2001, p. 71-86.

[27] Ulrik BRANDES et al. « On Modularity Clustering ». In : *IEEE Transactions on Knowledge and Data Engineering (TKDE)* 20.2 (2008), p. 172-188.

[28] Matthew BREHMER et Tamara MUNZNER. « A Multi-Level Typology of Abstract Visualization Tasks ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 19.12 (2013), p. 2376-2385.

[29] Wouter Van den BROECK et al. « The GLEaMviz Computational Tool, a Publicly Available Software to Explore Realistic Epidemic Spreading Scenarios at the Global Scale ». In : *BioMed Central Infectious Diseases* 11.1 (2011), p. 37.

[30] Ingwer BROG et Patrick J. F. GROENEN. *Modern multidimensional scaling : Theory and applications*. Springer-Verlag, 1997.

[31] Dale A. BRUNS et Thomas O. SWEET. « Geospatial Tools to Support Watershed Environmental Monitoring and Reclamation : Assessing Mining Impacts on the Upper Susquehanna-Lackawanna American Heritage River ». In : *Advanced Integration of Geospatial Technologies in Mining and Reclamation*. 2004.

[32] Lee BYRON et Martin WATTENBERG. « Stacked Graphs - Geometry & Aesthetics ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 14.6 (2008), p. 1245-1252.

[33] Stuart K. CARD, Jock D. MACKINLAY et Ben SHNEIDERMAN. *Readings in Information Visualization : Using Vision to Think*. Morgan Kaufmann, 1st edition, 1999.

[34] Lauren N CARROLL et al. « Visualization and Analytics Tools for Infectious Disease Epidemiology : a Systematic Review ». In : *Journal of Biomedical Informatics* 51 (2014), p. 287-298.

[35] Angelos CHATZIMPARMPAS et al. « A survey of surveys on the use of visualization for interpreting machine learning models ». In : *Information Visualization* (to appear).

[36] Fati CHEN et al. « Node Overlap Removal Algorithms : A Comparative Study ». In : *Proceeding of the International Symposium on Graph Drawing and Network Visualization (GD)*. Sous la dir. de Daniel ARCHAMBAULT et Csaba D. TÓTH. T. 11904. Lecture Notes in Computer Science. Springer, 2019, p. 179-192.

[37] Fati CHEN et al. « Node Overlap Removal Algorithms : an Extended Comparative Study ». In : *Journal of Graph Algorithms and Applications (JGAA)* (2020), to apprear.

[38] Markus CHIMANI et al. « The Open Graph Drawing Framework (OGDF) ». In : *Handbook on Graph Drawing and Visualization*. Sous la dir. de Roberto TAMASSIA. Chapman et Hall / CRC, 2013, p. 543-569.

[39] William S. CLEVELAND et Robert MCGILL. « Graphical Perception : Theory, Experimentation, and Application to the Development of Graphical Methods ». In : *Journal of the American Statistical Association* 79.387 (1984), p. 531-554.

[40] Andy COCKBURN, Amy K. KARLSON et Benjamin B. BEDERSON. « A review of overview+detail, zooming, and focus+context interfaces ». In : *ACM Computing Surveys* 41.1 (2008), 2 :1-2 :31.

[41] Nigel COLLIER et al. « BioCaster : Detecting Public Health Rumors with a Web-based Text Mining System ». In : *Bioinformatics* 24.24 (2008), p. 2940-2941.

[42] Erick CUENCA et al. « MultiStream : A Multiresolution Streamgraph Approach to Explore Hierarchical Time Series ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 24.12 (2018), p. 3160-3173.

[43] Erick CUENCA et al. « Visual querying of large multilayer graphs ». In : *Proceeding of the International Conference on Scientific and Statistical Database Management (SSDBM)*. 2018, 32 :1-32 :4.

[44] Weiwei CUI et al. « How Hierarchical Topics Evolve in Large Text Corpora ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 20.12 (2014), p. 2281-2290.

[45] Weiwei CUI et al. « TextFlow : Towards Better Understanding of Evolving Topics in Text ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 17.12 (2011), p. 2412-2421.

[46] Nicholas K. DEWIND et al. « Modeling the approximate number system to quantify the contribution of visual stimulus features ». In : *Cognition* 142 (2015), p. 247 -265.

[47] Wenwen DOU et al. « HierarchicalTopics : Visually Exploring Large Text Collections Using Topic Hierarchies ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 19.12 (2013), p. 2002-2011.

[48] Cody DUNNE et al. « VoroGraph : Visualization Tools for Epidemic Analysis ». In : *Proceedings of the ACM Conference on Human Factors in Computing System (CHI)*. 2015, p. 255-258.

[49] Tim DWYER, Kim MARRIOTT et Peter J STUCKEY. « Fast node overlap removal ». In : *Proceedings of the International Symposium on Graph Drawing (GD)*. Springer, 2005, p. 153-164.

[50] Niklas ELMQVIST, John STASKO et Philippas TSIGAS. « DataMeadow : A Visual Canvas for Analysis of Large-Scale Multivariate Data ». In : *Information Visualization* 7.1 (2008), p. 18-33.

[51] Alex ENDERT et al. « The State of the Art in Integrating Machine Learning into Visual Analytics ». In : *Computer Graphics Forum* 36.8 (2017), p. 458-486.

[52] Paul ERDÖS et Alfréd RÉNYI. « On random graphs ». In : *Publicationes Mathematicae Debrecen* 6 (1959), p. 290-291.

[53] EUROPEAN ENVIRONMENT AGENCY. *CLC2006 Technical Guidelines*. Rapp. tech. No 17/2007. Office for Official Publications of the European Communities, 2006.

[54] EUROPEAN UNION. « Council directive 98/83/EC of 3 November 1998 on the quality of water intended for human consumption ». In : *Official Journal* OJ L 330 (2009), p. 1-32.

[55] Mickaël FABRÈGUE et al. « Mining closed partially ordered patterns, a new optimized algorithm ». In : *Knowledge Based Systems* 79 (2015), p. 68-79.

[56] Mickaël FABRÈGUE et al. « OrderSpan : Mining Closed Partially Ordered Patterns ». In : *International Symposium on Intelligent Data Analysis (IDA)*. 2013, p. 186-197.

[57] Samiha FADLOUN et al. « EpidVis : A visual web querying tool for animal epidemiology surveillance ». In : *Information Visualization* 19.1 (2020), p. 48-64.

[58] Samiha FADLOUN et al. « Node Overlap Removal for 1D Graph Layout ». In : *Proceeding of the International Conference on Information Visualisation (iV)*. 2017, p. 224 -229.

[59] Samiha FADLOUN et al. *Node Overlap Removal for 1D Graph Layout : Proof of Theorem 1*. Rapp. tech. Lirmm-01521385. LIRMM - Université de Montpellier, 2017.

[60] Adam B. FORGANG, Bernd HAMANN et Carl F. CERCO. « Visualization of water quality data for the Chesapeake Bay ». In : *IEEE Symposium on Information Vizualization (InfoVis)*. 1996, p. 417-420.

[61] Clark C FREIFELD et al. « HealthMap : Global Infectious Disease Monitoring through Automated Classification and Visualization of Internet Media Reports ». In : *Journal of the American Medical Informatics Association* 15.2 (2008), p. 150-157.

[62] George W. FURNAS. « A fisheye follow-up : further reflections on focus + context ». In : *Proceedings of the ACM Conference on Human Factors in Computing System (CHI)*. 2006, p. 999-1008.

[63] George W. FURNAS. « Generalized Fisheye Views ». In : *Proceedings of the ACM Conference on Human Factors in Computing System (CHI)*. 1986, p. 16-23.
[64] Emden GANSNER et Yifan HU. « Efficient, proximity-preserving node overlap removal. » In : *Journal of Graph Algorithms and Applications (JGAA)* 14.1 (2010), p. 53-74.
[65] Emden R GANSNER et Stephen C NORTH. « An open graph visualization system and its applications to software engineering ». In : *Software : practice and experience* 30.11 (2000), p. 1203-1233.
[66] Rafael GARCIA et al. « A task-and-technique centered survey on visual analytics for deep learning model engineering ». In : *Computer & Graphics (C&G)* 77 (2018), p. 30-49.
[67] Michael R. GAREY et David S. JOHNSON. *Computers and Intractability : a Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.
[68] Mohammad GHONIEM et al. « NewsLab : Exploratory Broadcast News Video Analysis ». In : *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST)*. 2007, p. 123-130.
[69] Michael GLEICHER. « Considerations for Visualizing Comparison ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 24.1 (2018), p. 413-423.
[70] Michael GLEICHER et al. « Visual comparison for information visualization ». In : *Information Visualization* 10.4 (2011), p. 289-309.
[71] Rohan GOEL et al. « EpidNews : An Epidemiological News Explorer for Monitoring Animal Diseases ». In : *Proceeding of the International Symposium on Visual Information Communication and Interaction (VINCI)*. 2018, p. 1-8.
[72] Rohan GOEL et al. « EpidNews : Extracting, Exploring and Annotating News for Monitoring Animal Diseases ». In : *Journal of Computer Languages (COLA)* 56 (2020), p. 100936.
[73] Ian J. GOODFELLOW, Yoshua BENGIO et Aaron C. COURVILLE. *Deep Learning*. MIT Press, 2016.
[74] Herbert GUTTINGER et Werner STUMM. « Ecotoxicology An Analysis of the Rhine Pollution caused by the Sandoz Chemical Accident, 1986 ». In : *Interdisciplinary Science Reviews* 17.2 (1992), p. 127-136.
[75] Stefan HACHUL et Michael JÜNGER. « Drawing Large Graphs with a Potential-Field-Based Multilevel Algorithm ». In : *Proceedings of the International Symposium on Graph Drawing (GD)*. Springer, 2005, p. 285-295.
[76] Robert L. HARRIS. *Information Graphics : A Comprehensive Illustrated Reference*. Oxford University Press, 1999.
[77] Herman J. HAVERKORT et Freek van WALDERVEEN. « Locality and bounding-box quality of two-dimensional space-filling curves ». In : *Computational Geometry, Theory and Applications* 43.2 (2010), p. 131-147.
[78] Susan HAVRE, Elizabeth HETZLER et Lucy NOWELL. « ThemeRiver : Visualizing Theme Changes over Time ». In : *Proceedings of the IEEE Symposium on Information Visualization (InfoVis)*. 2000, p. 115-123.
[79] Susan HAVRE et al. « ThemeRiver : Visualizing Thematic Changes in Large Document Collections ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 8.1 (2002), p. 9-20.
[80] Kunihiko HAYASHI et al. « A Layout Adjustment Problem for Disjoint Rectangles Preserving Orthogonal Order ». In : *Proceedings of the International Symposium on Graph Drawing (GD)*. Springer, 1998, p. 183-197.
[81] Christopher G. HEALEY et James T. ENNS. « Attention and Visual Memory in Visualization and Computer Graphics ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 18.7 (2012), p. 1170-1188.
[82] Jeffrey HEER et Michael BOSTOCK. « Crowdsourcing graphical perception : using mechanical turk to assess visualization design ». In : *Proceeding of the ACM Conference on Human Factors in Computing Systems (CHI)*. 2010, p. 203-212.
[83] Orland HOEBER, Xue Dong YANG et Yiyu YAO. « VisiQ : Supporting Visual and Interactive Query Refinement ». In : *Web Intelligence and Agent Systems : An International Journal* 5.3 (2007), p. 311-329.
[84] Fred HOHMAN et al. « Visual Analytics in Deep Learning : An Interrogative Survey for the Next Frontiers ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 25.8 (2019), p. 2674-2693.

[85] Danny HOLTEN. « Hierarchical Edge Bundles : Visualization of Adjacency Relations in Hierarchical Data ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 12.5 (2006), p. 741-748.

[86] Yifan HU. « Efficient, high-quality force-directed graph drawing ». In : *Mathematica Journal* 10.1 (2005), p. 37-71.

[87] Xiaodi HUANG et al. « A new algorithm for removing node overlapping in graph visualization ». In : *Information Sciences* 177.14 (2007), p. 2821 -2844.

[88] Seok Soon Park HYE WON LEE Kon Joon Bhang. « Effective visualization for the spatiotemporal trend analysis of the water quality in the Nakdong River of Korea ». In : *Ecological Informatics* 5.3 (2010), 281–292.

[89] Paul JACCARD. « Distribution de la flore alpine dans le Bassin de Dranses et dans quelques regions voisines ». In : *Bulletin de la Société Vaudoise des Sciences Naturelles* 37 (1901), p. 241-272.

[90] Michael A. JENKINS. « Algorithm 493 : Zeros of a Real Polynomial [C2] ». In : *ACM Transactions on Mathematical Software* 1.2 (1975), p. 178-189.

[91] Alan KEAHEY. « The Generalized Detail-In-Context Problem ». In : *Proceedings of the IEEE Symposium on Information Visualization (InfoVis)*. 1998, p. 44-51.

[92] Daniel A. KEIM et al. « Visual Analytics : Definition, Process, and Challenges ». In : *Information Visualization - Human-Centered Issues and Perspectives*. T. 4950. Lecture Notes in Computer Science. Springer, 2008, p. 154-175.

[93] Andreas KERREN, Helen C. PURCHASE et Matthew O. WARD, éd. *Multivariate Network Visualization*. T. 8380. Lecture Notes in Computer Science. Springer, 2014.

[94] Mikko KIVELÄ et al. « Multilayer networks ». In : *Journal of Complex Networks* 2.3 (2014), p. 203-271.

[95] Mikko KIVELÄ et al. « Visual Analytics of Multilayer Networks Across Disciplines (Dagstuhl Seminar 19061) ». In : *Dagstuhl Reports* 9.2 (2019), p. 1-26.

[96] Roman KLAPAUKH. « An Empirical Evaluation of Force-Directed Graph Layout ». Thèse de doct. Victoria University of Wellington, 2014.

[97] Yehuda KOREN et David HAREL. « A Multi-scale Algorithm for the Linear Arrangement Problem ». In : *Proceedings of the International Workshop on Graph-Theoretic Concepts in Computer Science (WG)*. T. 2573. LNCS. Springer, 2002, p. 296-309.

[98] Robert KOSARA et al. « User Studies : Why, How, and When ? » In : *IEEE Computer Graphics and Applications (CG&A)* 23.4 (2003), p. 20-25.

[99] Robert KRUEGER, Tina TREMEL et Dennis THOM. « VESPa 2.0 : Data-Driven Behavior Models for Visual Analytics of Movement Sequences ». In : *Proceedings of the International Symposium on Big Data Visual Analytics (BDVA)*. 2017, p. 1-8.

[100] Martin KRZYWINSKI et al. « Hive Plots-Rational Approach to Visualizing Networks ». In : *Briefings in Bioinformatics* 13.5 (2011), p. 627-644.

[101] Nathalie LALANDE et al. « Implementing the DPSIR framework to link water quality of rivers to land use : methodological issues and preliminary field test ». In : *International Journal of River Basin Management* (2014), p. 1-17.

[102] Heidi LAM et al. « Empirical Studies in Information Visualization : Seven Scenarios ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 18.9 (2012), p. 1520-1536.

[103] Antoine LAMBERT, Romain BOURQUI et David AUBER. « Winding Roads : Routing edges into bundles ». In : *Computer Graphics Forum* 29.3 (2010), p. 853-862.

[104] Tim LAMMARSCH et al. « Developing an Extended Task Framework for Exploratory Data Analysis Along the Structure of Time ». In : *Proceeding of the EuroVis Workshop on Visual Analytics (EuroVA)*. 2012.

[105] Bongshin LEE et al. « Task taxonomy for graph visualization ». In : *Proceeding of the Workshop on BEyond time and errors : novel evaluation methods for information visualization (BELIV)*. 2006, p. 1-5.

[106] Jan de LEEUW. « Applications of convex analysis to multidimensional scaling ». In : *Recent Developments in Statistics*. 1977, p. 133-145.

[107] Wanchun LI, Peter EADES et Nikola NIKOLOV. « Using Spring Algorithms to Remove Node Overlapping ». In : *Proceedings of the Asia-Pacific Symposium on Information Visualisation (APVis)*. 2005, p. 131-140.

[108] Manuel LIMA. *Cartographie des réseaux : l'art de représenter la complexité*. Eyrolles (traduit de l'édition anglaise originale : *Visual Complexity : Mapping Patterns of Information*, Princeton Architectural Press, 2011), 2013.

[109] Gunnar LISCHEID. « Non-linear visualization and analysis of large water quality data sets : a model-free basis for efficient monitoring and risk assessment ». In : *Stochastic Environmental Research and Risk Assessment* 23.7 (2009), p. 977-990.

[110] Shixia LIU et al. « Bridging Text Visualization and Mining : A Task-Driven Survey ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 25.7 (2019), p. 2482-2504.

[111] Shixia LIU et al. « TIARA : Interactive, Topic-Based Visual Text Summarization and Analysis ». In : *ACM Transactions on Intelligent Systems and Technology* 3.2 (2012), 25 :1-25 :28.

[112] Shixia LIU et al. « Towards better analysis of machine learning models : A visual analytics perspective ». In : *Visual Informatics* 1.1 (2017), p. 48-56.

[113] Shusen LIU et al. « Visualizing High-Dimensional Data : Advances in the Past Decade ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 23.3 (2017), p. 1249-1268.

[114] María-Jesús LOBO, Emmanuel PIETRIGA et Caroline APPERT. « An Evaluation of Interactive Map Comparison Techniques ». In : *Proceeding of the ACM Conference on Human Factors in Computing Systems (CHI)*. 2015, p. 3573-3582.

[115] Juan Antonio LOSSIO-VENTURA et al. « Biomedical Term Extraction : Overview and a New Methodology ». In : *Information Retrieval Journal* 19.1 (2016), p. 59-99.

[116] Felice De LUCA, Md. Iqbal HOSSAIN et Stephen G. KOBOUROV. « Symmetry Detection and Classification in Drawings of Graphs ». In : *International Symposium on Graph Drawing and Network Visualization (GD)*. Sous la dir. de Daniel ARCHAMBAULT et Csaba D. TÓTH. T. 11904. Lecture Notes in Computer Science. Springer, 2019, p. 499-513.

[117] Laurens van der MAATEN et Geoffrey HINTON. « Visualizing data using t-SNE ». In : *Journal of machine learning research* 9 (2008), p. 2579-2605.

[118] Kim MARRIOTT et al. « Removing node overlapping in graph layout using constrained optimization ». In : *Constraints* 8.2 (2003), p. 143-171.

[119] Riccardo MAZZA. *Introduction to Information Visualization*. Springer-Verlag, 2009.

[120] David MCCANDLESS. *Datavision*. Robert Laffont (traduit de l'édition anglaise originale : *Information is Beautiful*, Collins, 2010), 2011.

[121] David MCCANDLESS. *Datavision*^{2}. Robert Laffont (traduit de l'édition anglaise originale : *Knowledge is Beautiful*, Collins, 2014), 2014.

[122] Fintan MCGEE et al. « The State of the Art in Multilayer Network Visualization ». In : *Computer Graphics Forum* 38.6 (2019), p. 125-149.

[123] Yves MERCADIER et al. « #AIDS Analyse Information Dangers Sexualité : caractériser les discours à propos du VIH dans les forums de santé ». In : *Actes des Journées francophones d'Ingénierie des Connaissances (IC)*. 2018, p. 71-86.

[124] Wouter MEULEMANS. « Efficient Optimal Overlap Removal : Algorithms and Experiments ». In : *Computer Graphics Forum* 38.3 (2019), p. 713-723.

[125] Michel MEYBECK et Richard HELMER. « The quality of rivers : From pristine stage to global pollution ». In : *Global and Planetary Change* 1.4 (1989), p. 283-309.

[126] Miriah D. MEYER et al. « The nested blocks and guidelines model ». In : *Information Visualization* 14.3 (2015), p. 234-249.

[127] Kazuo MISUE et al. « Layout adjustment and the mental map ». In : *Journal of Visual Languages & Computing (JVLC)* 6.2 (1995), p. 183-210.

[128] Chris MUELDER et Kwan-Liu MA. « Rapid graph layout using space filling curves ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 14.6 (2008), p. 1301-1308.

[129] Chris MUELDER et al. « Egocentric storylines for visual analysis of large dynamic graphs ». In : *Proceeding of the Workshop on Big Data Visualization (BigDataVis)*. 2013, p. 56-62.

[130] Chris MUELDER et al. « Improved Cluster Tracking for Visualization of Large Dynamic Graphs ». In : *Actes de l'atelier Visualisation d'information, Interaction et Fouille de données (VIF), 13ième Conférence Internationale Francophone sur l'Extraction des Connaissances (EGC)*. 2013, p. 21-32.

[131] Tamara MUNZNER. « A Nested Process Model for Visualization Design and Validation ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 15.6 (2009), p. 921-928.

[132] Tamara MUNZNER. « Process and Pitfalls in Writing Information Visualization Research Papers ». In : *Information Visualization - Human-Centered Issues and Perspectives*. Sous la dir. d'Andreas KERREN et al. T. 4950. Lecture Notes in Computer Science. Springer, 2008, p. 134-153.

[133] Tamara MUNZNER. *Visualization Analysis and Design*. A.K. Peters visualization series. A K Peters, 2014. ISBN : 978-1-466-50891-0.

[134] Lev NACHMANSON et al. « Node overlap removal by growing a tree ». In : *Proceedings of the International Symposium on Graph Drawing and Network Visualization (GD)*. Springer, 2016, p. 33-43.

[135] Richard A NEHER et Trevor BEDFORD. « Nextflu : Real-Time Tracking of Seasonal Influenza Virus Evolution in Humans ». In : *Bioinformatics* 31.21 (2015), p. 3546-3548.

[136] Mark E. J. NEWMAN et Michelle GIRVAN. « Graph clustering ». In : *Physical Review E* 69.026113 (2004).

[137] Shaoliang NIE et al. « Visualizing Deep Neural Networks for Text Analytics ». In : *IEEE Pacific Visualization Symposium (PacificVis)*. 2018, p. 180-189.

[138] Andreas NOACK. « Modularity clustering is force-directed layout ». In : *Physical Review E* 79 (2 2009), p. 026102.

[139] Carolina NOBRE et al. « The State of the Art in Visualizing Multivariate Networks ». In : *Computer Graphics Forum* 38.3 (2019), p. 807-832.

[140] Jordi PETIT. « Experiments on the minimum linear arrangement problem ». In : *ACM Journal of Experimental Algorithmics* 8 (2003).

[141] Robert PIENTA et al. « Visual Graph Query Construction and Refinement ». In : *Proceedings of the International Conference on Management of Data (SIGMOD)*. 2017, p. 1587-1590.

[142] Catherine PLAISANT. « The challenge of information visualization evaluation ». In : *Proceeding of the Conference on Advanced visual interfaces (AVI)*. 2004, p. 109-116.

[143] Nissanka B. PRIYANTHA et al. « Anchor-free distributed localization in sensor networks ». In : *International Conference on Embedded Networked Sensor Systems (SenSys)*. 2003, p. 340-341.

[144] Helen C. PURCHASE. *Experimental Human-Computer Interaction - A Practical Guide with Visual Examples*. Cambridge University Press, 2012.

[145] Helen C. PURCHASE. « Metrics for Graph Drawing Aesthetics ». In : *Journal of Visual Languages & Computing (JVLC)* 13.5 (2002), p. 501-516.

[146] Oky PURWANTININGSIH et al. « Visual analysis of body movement in serious games for healthcare ». In : *Proceeding of the IEEE Pacific Visualization Symposium (PacificVis)*. 2016, p. 229-233.

[147] Paulo E. RAUBER et al. « Visualizing the Hidden Activity of Artificial Neural Networks ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 23.1 (2017), p. 101-110.

[148] Denis REDONDO et al. « Layer-Centered Approach for Multigraphs Visualization ». In : *Proceeding of the International Conference on Information Visualisation (iV)*. 2015, p. 50-55.

[149] Hans ROSLING et Zhongxing ZHANG. « Health Advocacy with Gapminder Animated Statistics ». In : *Journal of Epidemiology and Global Health* 1.1 (2011), p. 11-14.

[150] Timothé ROSSA et al. « Experimental Web Service and Eye-Tracking Setup for Unilateral Spatial Neglect Assessment ». In : *Proceeding of the Human-Computer Interaction International Conference (HCII)*. Sous la dir. de Vincent G. DUFFY. T. 11582. Lecture Notes in Computer Science. Springer, 2019, p. 141-155.

[151] Robert E. ROTH. « An Empirically-Derived Taxonomy of Interaction Primitives for Interactive Cartography and Geovisualization ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 19.12 (2013), p. 2356-2365.

[152] Sébastien RUFIANGE et Michael J. MCGUFFIN. « DiffAni : Visualizing Dynamic Graphs with a Hybrid of Difference Maps and Animation ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 19.12 (2013), p. 2556-2565.

[153] James A. RUSSELL. « A circumplex model of affect ». In : *Journal of personality and social psychology* 39.6 (1980), p. 1161-1178.

[154] Hiroaki SAKOE et Seibi CHIBA. « A dynamic programming approach to continuous speech recognition ». In : *International Congress on Acoustics*. T. 3. 1971, p. 65-69.
[155] Arnaud SALLABERRY, Chris MUELDER et Kwan-Liu MA. « Clustering, Visualizing, and Navigating for Large Dynamic Graphs ». In : *Proceeding of the International Symposium on Graph Drawing (GD)*. Sous la dir. de Walter DIDIMO et Maurizio PATRIGNANI. T. 7704. Lecture Notes in Computer Science. Springer, 2012, p. 487-498.
[156] Arnaud SALLABERRY et Pascal PONCELET. « La "DataViz" : Donnez vie à vos données ! » In : *Maths Mouvement Express, Comité International des Jeux Mathématiques*. 2018, p. 57-62.
[157] Arnaud SALLABERRY et al. « Contact Trees : Network Visualization beyond Nodes and Edges ». In : *PLoS One* 11.1 (2016), e0146368.
[158] Arnaud SALLABERRY et al. « Interactive Visualization and Navigation of Web Search Results Revealing Community Structures and Bridges ». In : *Proceeding of the Graphics Interface conference (GI)*. 2010, p. 105-112.
[159] Hans-Jörg SCHULZ. « Treevis.net : A Tree Visualization Reference ». In : *IEEE Computer Graphics & Applications (CG&A)* 31.6 (2011), p. 11-15.
[160] Sarah SCHÖTTLER, Tobias KAUER et Benjamin BACH. « Geographic Network Visualization Techniques : A Work-In-Progress Taxonomy ». In : *International Symposium on Graph Drawing and Network Visualization (GD)*. 2019.
[161] Michael SEDLMAIR, Miriah D. MEYER et Tamara MUNZNER. « Design Study Methodology : Reflections from the Trenches and the Stacks ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 18.12 (2012), p. 2431-2440.
[162] Edward SEGEL et Jeffrey HEER. « Narrative Visualization : Telling Stories with Data ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 16.6 (2010), p. 1139-1148.
[163] Anselm SPOERRI. « InfoCrystal : a Visual Tool for Information Retrieval ». In : *Proceedings of the IEEE Conference on Visualization (VIS)*. 1993, p. 150-157.
[164] Stanley S. STEVENS. *Psychophysics : introduction to its perceptual, neural, and social prospects*. Willey, 1975.
[165] Chris STOLTE, Diane TANG et Pat HANRAHAN. « Polaris : a System for Query, Analysis, and Visualization of Multidimensional Databases ». In : *Communications of the Association for Computing Machinery* 51.11 (2008), p. 75-84.
[166] Hendrik STROBELT et al. « LSTMVis : A Tool for Visual Analysis of Hidden State Dynamics in Recurrent Neural Networks ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 24.1 (2018), p. 667-676.
[167] Hendrik STROBELT et al. « Rolled-out Wordles : A Heuristic Method for Overlap Removal of 2D Data Representatives ». In : *Computer Graphics Forum* 31.3 (2012), p. 1135-1144.
[168] Guodao SUN et al. « EvoRiver : Visual Analysis of Topic Coopetition on Social Media ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 20.12 (2014), p. 1753-1762.
[169] Eniko SZEKELY et al. « A Graph-Based Method for Detecting Rare Events : Identifying Pathologic Cells ». In : *IEEE Computer Graphics and Applications (CG&A)* 35.3 (2015), p. 65-73.
[170] Roberto TAMASSIA, éd. *Handbook on Graph Drawing and Visualization*. Chapman et Hall / CRC, 2013.
[171] Alexandru C. TELEA. *Data Visualization : Principles and Practice*. A K Peters, Ltd., 2008.
[172] Dennis THOM et Thomas ERTL. « TreeQueST : A Treemap-Based Query Sandbox for Microdocument Retrieval ». In : *Proceedings of the Hawaii International Conference on System Sciences*. 2015, p. 1714-1723.
[173] James J. THOMAS et Kristen A. COOK. *Illuminating the Path*. IEEE Computer Society Press, 2005.
[174] Melanie TORY et Torsten MÖLLER. « Human Factors in Visualization Research ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 10.1 (2004), p. 72-84.
[175] Edward R. TUFTE. *Envisionning Information*. Graphics Press, 1990.
[176] Edward R. TUFTE. *The Visual Display of Quantitive Information*. Graphics Press, 1983.
[177] Edward R. TUFTE. *Visual Explanations*. Graphics Press, 1997.

[178] William T. TUTTE. « How to draw a graph ». In : *Proceedings of the London Mathematical Society* 13 (1963), p. 743-768.
[179] UNEP/WHO. *Water Quality Monitoring - A Practical Guide to the Design and Implementation of Freshwater Quality Studies and Monitoring Programmes*. UNEP/WHO publications, 1996.
[180] Manasi VARTAK et al. « SEEDB : Automatically Generating Query Visualizations ». In : *Proceedings of the Very Large Data Bases Endowment (VLDB)* 7.13 (2014), p. 1581-1584.
[181] Fernanda B VIEGAS et al. « ManyEyes : A Site for Visualization at Internet Scale ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 13.6 (2007), p. 1121-1128.
[182] Florence Ying WANG et al. « SentiCompass : Interactive visualization for exploring and comparing the sentiments of time-varying twitter data ». In : *Proceeding of the IEEE Pacific Visualization Symposium (PacificVis)*. 2015, p. 129-133.
[183] Yun WANG et al. « STAC : Enhancing Stacked Graphs for Time Series Analysis ». In : *Proceeding of the IEEE Pacific Visualization Symposium (PacificVis)*. 2016, p. 234-238.
[184] Matthew WARD, Georges GRINSTEIN et Daniel KEIM. *Interactive Data Visualization : Foundations, Techniques, and Applications*. A K Peters, Ltd., 2010.
[185] Colin WARE. *Information visualizatin : Perception for design*. Morgan Kaufmann, 2000.
[186] Colin WARE et al. « Cognitive measurements of graph aesthetics ». In : *Information Visualization* 1.2 (2002), p. 103-110.
[187] Martin WATTENBERG. « Baby Names, Visualization, and Social Data Analysis ». In : *Proceedings of the IEEE Symposium on Information Visualization (InfoVis)*. 2005, p. 1-7.
[188] Martin WATTENBERG et Jesse KRISS. « Designing for Social Data Analysis ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 12.4 (2006), p. 549-557.
[189] Duncan J. WATTS et Steven Henry STROGATZ. « Collective dynamics of 'small-world' networks ». In : *Nature* 393 (1998), p. 440-442.
[190] Eric WELCH et Stephen G. KOBOUROV. « Measuring Symmetry in Drawings of Graphs ». In : *Computer Graphics Forum* 36.3 (2017), p. 341-351.
[191] WHO/UNEP. *Water Pollution Control - A Guide to the Use of Water Quality Management Principles*. WHO publications, 1997.
[192] Jarke J. van WIJK. « Views on Visualization ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 12.4 (2006), p. 421-433.
[193] Leland WILKINSON. *The grammar of Graphics*. Springer-Verlag, Statistics et Computing, 1999.
[194] Yingcai WU et al. « OpinionFlow : Visual Analysis of Opinion Diffusion on Social Media ». In : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 20.12 (2014), p. 1763-1772.
[195] Peipei YI et al. « AutoG : a Visual Query Autocompletion Framework for Graph Databases ». In : *VLDB Journal* 26.3 (2017), p. 347-372.
[196] Faraz ZAIDI, Chris MUELDER et Arnaud SALLABERRY. « Analysis and Visualization of Dynamic Networks ». In : *Encyclopedia of Social Network Analysis and Mining, 2nd Edition*. Sous la dir. de Reda ALHAJJ et Jon G. ROKNE. Springer, 2018, p. 37-48.