Année 2014

École Doctorale du Pacifique

Thèse

présentée devant
l'Université de la Nouvelle-Calédonie

par
Jérémy Sanhes

pour obtenir le diplôme de
docteur spécialité informatique

Contribution à la fouille de données spatio-temporelles : application à l'étude de l'érosion

Soutenue publiquement le 25 Septembre 2014, devant le jury composé de :

Président
Henri Bonnel, Professeur ..... Université de la Nouvelle-Calédonie

Rapporteurs
Maguelonne Teisseire, Directrice de Recherche ..... IRSTEA, Université de Montpellier 2, CNRS
Alexandre Termier, Professeur ..... Université de Rennes 1

Examinateurs
Jérôme Azé, Professeur ..... Université de Montpellier 2
Frédéric Flouvat, Maître de Conférences ..... Université de la Nouvelle-Calédonie
Florence Le Ber, Directrice de Recherche ..... ENGEES, Université de Strasbourg

Directeurs
Jean-François Boulicaut, Professeur ..... INSA Lyon, CNRS
Nazha Selmaoui-folcher, Maître de Conférences, HDR ..... Université de la Nouvelle-Calédonie

Remerciements

Cette thèse a été financée par le projet FoSTEr *supporté par l'*anr.

Il est assez difficile d'écrire des remerciements concernant un travail de trois années entières. Le nombre de personnes impliquées dans l'accomplissement de la tâche va bien au-delà de ce que j'aurais pu penser en l'entamant. Ce fut un trajet loin d'être droit, et à plusieurs égards. Parfois joyeux, parfois douloureux. C'est pourquoi les lignes qui suivent expriment ma reconnaissance envers les personnes qui m'ont permis d'aller jusqu'au bout. Celles qui me connaissent sauront que les quelques mots qui leur sont consacrés valent bien plus que de longues phrases.

Je remercie Alexandre Termier et Maguelonne Teisseire d'avoir accepté de relire cette thèse et d'en être les rapporteurs. Je remercie également Jérôme Azé, Henri Bonnel, et Florence Le Ber d'avoir accepté d'être membres du jury.

Je remercie mes directeurs de thèse, Nazha Selmaoui-Folcher et Jean-François Boulicaut, ainsi que Frédéric Flouvat pour m'avoir encadré. Je remercie Nazha d'avoir créé les conditions nécessaires à l'élaboration de cette thèse. Je remercie Jean-François pour son soutien et sa sincérité. Je remercie Frédéric pour son implication, sa patience et sa constance.

Je remercie mes collègues et amis, de Nouvelle-Calédonie comme de métropole, pour leur présence et leur écoute. Nicolas[1], Clément, Arthur, Audrey, Claire, David, Mélanie, Étienne, Cheng Cheng, Aurélie, Pierre-Nicolas, Hai Nam, Claude, Baptiste, Elisabeth, Viviane, Hugo... j'en oublie forcément.

Je remercie ma famille, pour m'avoir soutenu dans mes choix, quels qu'ils furent.

Enfin, je remercie Stéphanie, sans qui rien n'aurait été possible.

1. La liste est non-ordonnée. L'opérateur « , » est commutatif. L'intersection des ensembles « amis » et « collègues » est non vide.

Sommaire

Liste des figures

Liste des algorithmes

Liste des acronymes

CTI	*Composite Threat Index* 95
DAG	*Directed Acyclic Graph* 11
ECD	Extraction de Connaissances à partir des Données 118
GPS	*Global Positioning System* 3
MIR	*Medium Infrared* 90
MNT	Modèle Numérique de Terrain 90
NDVI	*Normalized Difference Vegetation Index* 90
NIR	*Near Infrared* 90
REP	*Relative Erosion Prediction* 90
RMMF	*Revised Morgan-Morgan Finley* 90
RUSLE	*Revised Universal Soil Loss Equation* 90
SIG	Système d'Information Géographique 7
SOAP	*Spatial Object Association Pattern* 24
SPCOZ	*Spread Pattern of Spatio-Temporal Co-Occurrences over Zones* 23
SP-Tree	*Spread Pattern Tree* 23
USLE	*Universal Soil Loss Equation* 88
WDI	*Watershed Development Index* 90
WEPP	*Water Erosion Prediction Project* 90

Chapitre 1

Introduction

Sommaire

1. Contexte

La baisse des coûts des capteurs, ainsi que leur miniaturisation, ont été à l'origine d'une démocratisation sans précédent en terme de génération de données. On retrouve aujourd'hui ces capteurs de plus en plus dans nos appareils domestiques, tels que les accéléromètres dans les manettes de console de jeu, les capteurs infrarouges dans les caméras, les lecteurs d'empreinte digitale sur les ordinateurs, ou bien les balises GPS et boussoles dans les téléphones portables. Ces dispositifs, autrefois onéreux et réservés de fait à des projets industriels, ont permis le fantastique essor du marché de l'utilisation individuelle, s'adressant donc à un nombre considérablement plus important d'utilisateurs (potentiellement aussi grand que la population humaine mondiale) et augmentant d'autant le volume de données produites.

Conjointement à l'essor des capteurs, Internet a aussi bousculé les habitudes : l'information, qui est une ressource immatérielle, est décentralisée, potentiellement accessible de n'importe où et par n'importe qui, et pouvant être stockée et transférée presque à l'envi. Ce quasi-affranchissement des limites de stockage encourage la production de données. La tendance est d'ailleurs à la hausse : dans le domaine des sciences en particulier, la génération de données augmente de 30% par an selon Pryor (2012). Analyser en profondeur une telle quantité de données est un défi constamment renouvelé car sans cesse complexifié par la nature même des données, qu'il s'agisse de texte brut, de chiffres, d'images, de sons, de vidéos, de coordonnées GPS, de liens relationnels dans un réseau (entre personnes, ou entre stations hydrauliques, etc.), de relevés météorologiques, ou bien de structures moléculaires. Cette hétérogénéité des données doit nécessairement être prise en compte pour obtenir des analyses plus riches de sens. De plus, certaines données peuvent être imprécises (lorsqu'il s'agit de mesures notamment), parfois erronées, ou incomplètes.

C'est face à ces besoins de trier et synthétiser toutes ces sources d'information, de plus en plus nombreuses, diverses et complexes, et pourtant si profitables – voire indispensables – aux décideurs, que le développement de nouvelles méthodes d'analyse a vu le jour. L'enjeu est d'être capable de gérer à la fois le flot continu des données, leur nombre, leurs relations éventuelles, leur nature, tout en s'attachant à améliorer autant que faire se peut la finesse de l'analyse, et ce à l'aide de ressources calculatoires finies.

Des systèmes complets d'analyse ont été développés mettant en œuvre des processus décomposés en plusieurs étapes. En fouille de données en particulier, le processus d'Extraction de Connaissances à partir des Données (ECD) permet d'analyser des données et d'en extraire de la connaissance sans hypothèse *a priori*. Depuis Fayyad *et al.* (1996), le processus d'ECD est établi comme suit : dans un premier temps, les données doivent être choisies en fonction de leur adéquation avec le sujet d'étude (c'est l'étape de sélection présentée sur la figure 1.1) ; les données brutes ainsi sélectionnées devront ensuite être pré-traitées et éventuellement corrigées. En effet, certaines données peuvent être corrompues, contenir du bruit, ne pas être entièrement focalisées sur la zone d'étude voulue, ou bien contenir des « trous », tout cela à

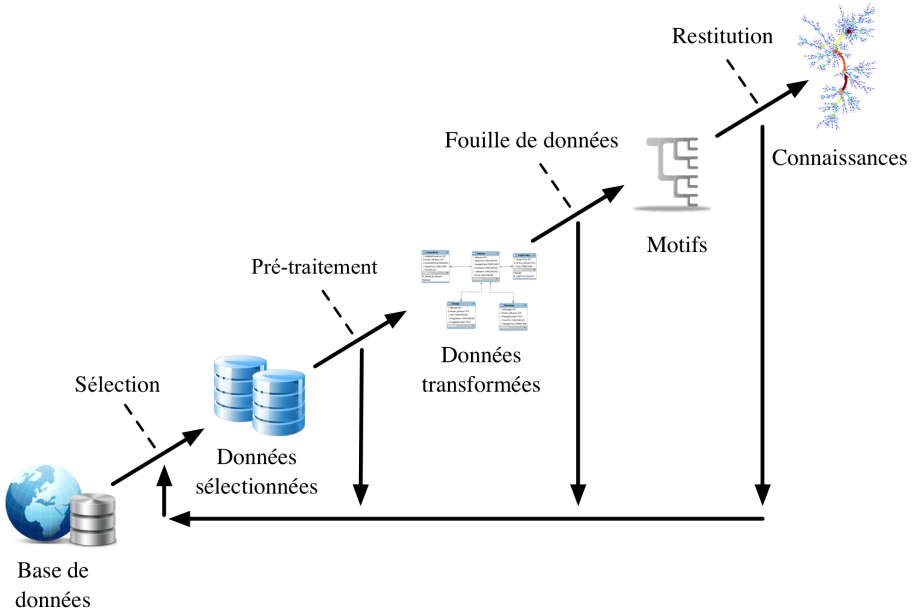


Figure 1.1 – Processus d'extraction de connaissances à partir des données (ECD) [1]

cause d'une éventuelle défaillance des capteurs, d'un problème de stockage de l'information, de données non disponibles, de phénomènes concomitants avec l'acquisition des données, de conventions et de normes analogiques ou numériques, ou bien de la simple impossibilité d'obtenir certaines données. Ces données devront aussi être formatées pour pouvoir être exploitables par la fouille de données, qui a pour objectif de donner de nouvelles informations permettant d'améliorer la connaissance humaine sur le sujet étudié. Bien que les étapes de sélection, pré-traitement et restitution font partie intégrante du procédé et sont absolument indispensables à l'acquisition de connaissances, le cœur de l'analyse se situe au niveau de la fouille des données. Chaque étape est elle-même itérative : le processus ECD peut et doit évoluer continuellement en fonction de l'ajout de nouvelles données, de nouvelles méthodes de fouille et des besoins d'analyse. Ainsi, il s'améliore constamment d'après les retours fournis par les connaissances acquises.

Plus généralement, la fouille de données est une discipline se trouvant à l'interface de l'intelligence artificielle, des statistiques, des bases de données et de l'algorithmie. Il s'agit :

— soit de traduire sous forme d'algorithme(s) les méthodes d'analyse humaine déjà utilisées – qui relèvent souvent elles-mêmes d'un processus intelligent complexe,

— soit de développer de nouvelles méthodes d'analyse pour extraire un ensemble d'informations plus petit que les données examinées, et qui possède des propriétés bien définies mathématiquement et reconnues comme présentant un intérêt potentiel.

Naturellement, déterminer le degré d'intérêt d'une propriété particulière peut relever de l'expertise des utilisateurs et du domaine étudié. Sur cet aspect, l'apport des informaticiens consistera à élaborer des mesures d'intérêt généralistes, non-spécifiques à une étude de cas

1. Schéma tiré de Fayyad *et al.*, 1996. Cette figure en particulier a été créée par Hugo Alatrista-Salas et apparaît dans son manuscrit de thèse (Alatrista-Salas, 2013).

précise. Il s'agira aussi de rendre le calcul de telles mesures – aussi simples soient-elles – réalisable à l'échelle des quantités de données fournies, qui sont de taille considérable par définition.

Même si l'on s'emploie à créer des algorithmes réutilisables sur plusieurs applications, on peut malgré tout discerner plusieurs contextes en fonction du type de la donnée étudiée. On distinguera, entre autres, les données catégorielles des données numériques, les données ensemblistes des données ordonnées et des données structurées. Dans ce mémoire, l'étude est portée sur des données spatio-temporelles, qui peuvent porter sur plusieurs de ces catégories de données.

2. Problèmes et motivations

Comme décrit précédemment, le potentiel informatif des données a amené les mentalités à accorder d'avantage de temps et d'énergie à leur collecte et leur analyse afin d'en exploiter les richesses informatives. C'est le cas, par exemple, des données dites « spatio-temporelles », à savoir des données référencées géographiquement et qui évoluent dans le temps. En l'occurrence, dans une base de données traditionnelle, une transaction (c'est-à-dire un enregistrement) se verra enrichie d'une information spatiale et d'une information temporelle. Là où nous avions auparavant une transaction sous la forme $T = (id_T$, informations), nous aurons T=(id_T, information spatiale, information temporelle, autres informations). Souvent, l'information spatiale et temporelle remplace ou s'ajoute à la clef unique de la transaction. De même, la mise au point toujours plus poussée de Systèmes d'Information Géographique (SIG) et le recours de plus en plus fréquents aux images satellites incitent au développement de méthodes spécifiques à des applications spatiales évoluant dans le temps. Par exemple, agriculteurs et urbanistes peuvent s'intéresser à l'évolution de parcelles agricoles ou immobilières.

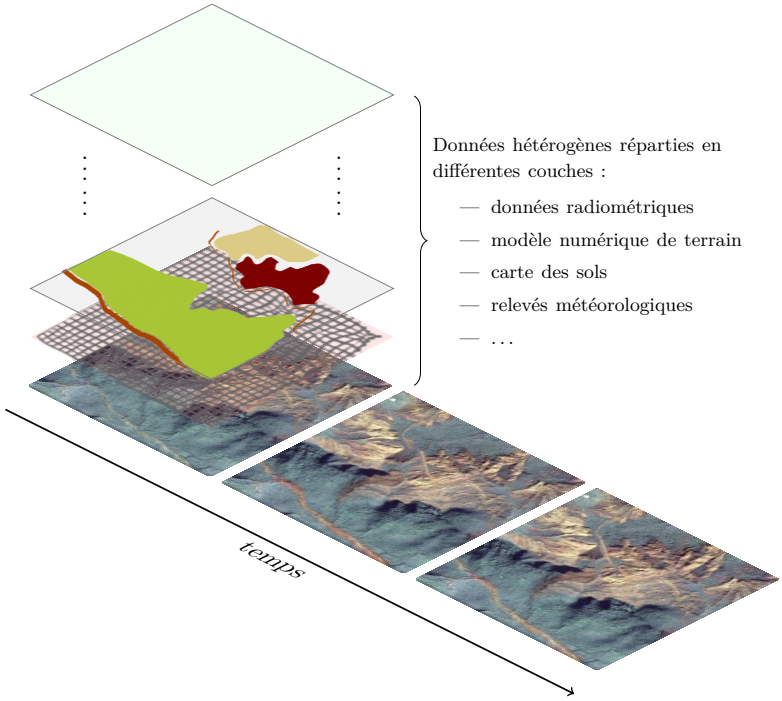


FIGURE 1.2 – Exemple de données spatio-temporelles : les données étudiées sont spatiales, et peuvent varier à chaque pas de temps. Dans cet exemple, les données sont aussi hétérogènes.

On peut citer aussi les données issues de flux migratoires : la valeur de telles données réside principalement dans la position géographique d'un objet d'étude (animal ou être humain) à un temps donné. De même, afin de pouvoir anticiper la propagation de maladies

vectorielles, d'autres chercheurs essayent d'analyser le comportement de phénomènes épidémiques à l'échelle d'une population et/ou d'un plus petit groupe d'individus. À des échelles spatiales et temporelles plus fines, certains s'intéressent à l'évolution de zones urbaines et péri-urbaines.

Cependant, les méthodes développées se cantonnent souvent à des problèmes spécifiques à une application précise, et ne couvrent en général qu'une facette du domaine spatio-temporel. En effet, d'autres besoins d'analyse existent, pour lesquels certaines modélisations proposées ne sont pas adaptées. Parmi ces besoins, nous traitons en particulier celui se focalisant non pas sur les objets d'étude, mais sur leur(s) propriétés, ainsi que leurs relations. Le cas du suivi de phénomènes géologiques fait partie de cette sous-classe de problème.

2.1 Les besoins en méthodes de fouille spatio-temporelle

Dans un grand nombre d'applications, la compréhension et la modélisation de la dynamique spatio-temporelle est un problème majeur (par exemple, pour la surveillance de l'érosion ou le suivi d'une épidémie). Yuan (2008) utilise le terme « dynamique » pour caractériser les forces qui influent sur le comportement d'un système dans l'espace et le temps. Par exemple, une épidémie de dengue est caractérisée par un ensemble de facteurs en interaction et causant la propagation de la maladie dans l'espace et le temps. Lorsque la dengue est déclarée dans un quartier, la question est de savoir comment, et en fonction de quels facteurs, elle va se propager dans les autres quartiers. Même si cette propagation semble dépendante de l'environnement direct des zones (points d'eau, mangroves à proximité, etc.) ainsi que d'un ensemble de circonstances évoluant dans le temps (humidité, chaleur, précipitation, etc.), la dynamique globale de propagation est loin d'être maîtrisée si on considère toutes les interactions possibles entre facteurs.

Face à ces questions, des méthodes formelles ont été développées afin d'aider les experts à valider ou à découvrir les dynamiques de propagation de ces phénomènes. Les méthodes de fouille de données spatio-temporelles visent à apporter des solutions pour mieux comprendre et décrire ces phénomènes complexes, le but étant de chercher des relations entre variables et événements sans hypothèse a priori. Parmi les méthodes de fouille de données, l'extraction de motifs caractérisant l'évolution d'un phénomène dans l'espace et dans le temps reste un problème ouvert en raison de la complexité des systèmes étudiés et des données disponibles.

Comme le montre Yao (2003), pendant longtemps, les travaux de recherche se sont focalisés sur une seule des dimensions (spatiale ou temporelle) sans prendre en compte l'autre. Historiquement, la fouille de données a vu le jour à partir d'un besoin d'analyse de consommation. Il s'agissait de mieux comprendre les comportements d'acheteurs, à partir de leur panier. Les premières méthodes de fouille ont donc porté sur le « panier de la ménagère » (Agrawal et Srikant, 1994) dont le type de données ensembliste est relativement simple. Quand il a ensuite fallu comprendre ce que les clients revenaient acheter, les chercheurs se sont mis à fouiller des séquences temporelles d'articles de consommation. Une composante temporelle venait d'être ajoutée. Notons cependant que les séquences ne se cantonnent pas à l'expression de la temporalité : ainsi, la fouille de séquences d'ADN en bio-informatique ne repose sur aucune composante temporelle.

En même temps, certains chercheurs se sont intéressés à l'analyse d'images statiques afin de les indexer dans une base de données donc à l'analyse de données spatiales. Les thèmes du spatial et du temporel ayant été développés parallèlement mais indépendamment, les chercheurs ont tout naturellement abordé le spatio-temporel comme un ajout de la composante spatiale à la composante temporelle, ou l'inverse. Ces dernières années, de plus en plus de travaux en fouille de données spatio-temporelles analysent conjointement l'aspect spatial et temporel. Généralement, ces travaux traitent deux types de problèmes : l'analyse des trajectoires d'objets en mouvement (Yao, 2003 ; Cao *et al.*, 2005 ; Yuan, 2008 ; Du *et al.*, 2009 ; Hai *et al.*, 2012) et l'analyse de la dynamique globale de phénomènes en fonction de leur environnement (Hsu *et al.*, 2009a ; Hsu *et al.*, 2009b ; Celik *et al.*, 2008 ; Celik *et al.*, 2006 ; Mohan *et al.*, 2010 ; Alatrista-Salas *et al.*, 2012).

Dans Cao *et al.*, 2005 ; Yuan, 2008 ; Du *et al.*, 2009, les auteurs caractérisent les trajectoires d'objets en mouvement par des séquences de tuples (l, t), où l est la localisation de l'objet au temps t. Dans ce type de problème, les trajectoires de chaque objet sont explicitement décrites dans les données. Les auteurs cherchent alors à extraire les trajectoires les plus fréquentes dans la base de trajectoires. Toutefois, dans un grand nombre d'applications, il ne s'agit pas de suivre le déplacement d'objets précis en mouvement, mais d'étudier l'évolution et les interactions globales, dans l'espace et dans le temps, d'ensembles d'événements en fonction de leur environnement. Un exemple d'application est l'étude de l'évolution des quartiers d'une ville en fonction de différents facteurs tels que le nombre de touristes, le type d'événements socio-culturels, ou le type d'attractions à visiter à proximité. Dans cet exemple, il est évident que les quartiers ne se déplacent pas. Il ne s'agit donc pas d'étudier des trajectoires mais d'étudier comment évolue l'ensemble des quartiers en fonction de leurs caractéristiques et des événements qui s'y produisent.

Dans le cas de l'étude de l'érosion, les objets spatiaux sont des zones qui non seulement ne sont pas fixes, mais peuvent en plus apparaître ou disparaître d'un temps à l'autre. Les phénomènes géologiques en jeu peuvent même amener plusieurs objets d'études à fusionner (quand il s'agit de lavakas [2] par exemple).

Face à ce problème, les séquences, et plus généralement les graphes, ont été utilisés pour représenter la *propagation* de phénomènes dans l'espace et dans le temps (Mabit *et al.*, 2011 ; Wang *et al.*, 2004 ; Wang *et al.*, 2005). Nous distinguerons les cas où les objets d'études sont précisément identifiables d'un temps à un autre (par exemple, grâce à des balises GPS pour le cas du suivi de trajectoires), des cas où une telle identification est soit difficile – voire impossible, soit non pertinente (par exemple, si les objets d'étude peuvent disparaître ou être profondément altérés dans le cas de l'étude de l'érosion).

2.1.1 Approches considérant les zones fixes dans le temps

Les travaux sur l'extraction des séquences, utilisés initialement pour rechercher des régularités temporelles, ont été appliqués et étendus au spatio-temporel. Par exemple, Tsoukatos et Gunopulos (2001) ont étendu les travaux sur les séquences d'itemsets afin d'extraire des séquences représentant l'*évolution* dans le temps de zones d'études (des quartiers, par

2. Ravines généralement larges et profondes ne résultant pas d'un simple glissement de terrain.

exemple). La base de données considérée est constituée de séquences d'itemsets (ensembles de caractéristiques environnementales) représentant l'évolution temporelle des différentes zones. Un algorithme effectuant un parcours en profondeur de l'espace de recherche est ensuite appliqué pour extraire les séquences les plus fréquentes (c'est-à-dire celles apparaissant dans le plus de zones).

Les auteurs ont également proposé une approche pour extraire les séquences fréquentes à une granularité spatiale plus élevée (à l'échelle de région par exemple) en exploitant les séquences fréquentes trouvées à une granularité plus faible (à l'échelle de la ville). Ils exploitent pour cela le fait que les séquences extraites à un niveau plus faible resteront fréquentes à un niveau de granularité plus élevé. La méthode recherche alors uniquement de nouvelles séquences fréquentes issues de l'agrégation spatiale. Dans le travail de Alatrista-Salas (2013), cette approche est étendue afin de prendre en compte les événements se produisant dans les zones voisines. Elle permet donc d'étudier l'évolution dans le temps de chaque zone en fonction des zones voisines. Par exemple, les motifs trouvés permettront de voir dans quel environnement avoisinant une séquence temporelle a eu lieu. Toutefois, comme pour l'approche précédente, les zones considérées sont fixes et ne se superposent pas. Elles correspondent typiquement aux quartiers d'une ville ou à des régions. Ces approches ne sont pas adaptées à des données utilisées où les objets d'intérêts sont très hétérogènes et peuvent apparaître, disparaître, ou se déformer tel que pour l'érosion.

2.1.2 Méthodes d'extraction de motifs spatio-temporels prenant en compte des dynamiques plus complexes

Mohan *et al.* (2010) étudient des graphes orientés acycliques de types d'événements. Cette approche permet d'étudier la propagation des événements pris individuellement (sans prendre en compte leur environnement). Ce modèle considère deux événements consécutifs s'ils sont spatialement proches (distance euclidienne inférieure à un seuil donné) et apparaissent dans la même fenêtre temporelle. Leurs motifs, appelés « motifs spatio-temporels en cascade », ne permettent malheureusement pas de prendre en compte l'environnement proche d'un événement.

Plus généralement, les graphes sont omniprésents dans divers domaines d'analyse de données (par exemple pour l'analyse de réseaux sociaux). Plusieurs algorithmes ont été proposés pour extraire les motifs à partir d'un ensemble de graphes (Inokuchi *et al.*, 2000 ; Borgelt et Berthold, 2002 ; Washio et Motoda, 2003 ; Wang *et al.*, 2006). La plupart de ces algorithmes d'extraction de graphes s'appliquent sur des graphes étiquetés, où chaque sommet ou arc n'a qu'une seule étiquette associée. Récemment, des modèles de graphes plus riches ont été considérés où les sommets et les arêtes sont étiquetés par plusieurs attributs ou propriétés (les *itemsets*), au lieu d'un seul attribut. Par exemple, un réseau social peut être représenté par un grand graphe où chaque sommet désignerait une personne ainsi que ses domaines d'intérêt. Ces graphes sont des graphes attribués, tels que décrits par Fukuzaki *et al.* (2010). Dans notre contexte, les dynamiques spatio-temporelles se modélisent naturellement via un graphe orienté acyclique attribué. Les sommets représentent des objets spatiaux, caractérisés par un ensemble d'attributs ou événements, tandis que les arcs peuvent exprimer leur proximité spatio-temporelle (c'est-à-dire des objets voisins dans des

temps consécutifs).

En effet, une des spécificités des données géographiques réside en ce que « *Everything is related to everything else, but near things are more related than distant things*[3] ». Ce concept illustre le concept de dépendance spatiale entre objets géographiques. Sous une telle considération, un graphe semble être une représentation naturelle pour modéliser les dépendances spatiales entre objets. Ces objets sont représentés par des sommets, et les dépendances spatiales par des arêtes. On pourra assigner divers attributs aux sommets pour représenter leurs caractéristiques respectives.

Si l'on veut rajouter une dimension temporelle au modèle, nous avons alors affaire à une suite ordonnée d'objets. Afin de modéliser les liens d'influence (c'est-à-dire les liens exprimant une transformation, ou une relation de cause à effet) qui peuvent exister entre objets appartenant à des temps successifs, l'utilisation d'arcs (c'est-à-dire, d'arêtes orientées) s'impose. Le graphe attribué devient donc un *Directed Acyclic Graph* (DAG) attribué, dont l'acyclisme découle du caractère causal du temps. Par exemple, on peut modéliser les dépendances spatio-temporelles entre zones habitées quand on se penche sur la propagation d'un virus, ou entre objets érodés et leur environnement quand on veut mieux comprendre les dynamiques d'érosion des sols. Dans un DAG attribué représentant l'étendue d'un virus de maladie vectorielle dans une ville (par exemple, la Dengue), les sommets pourraient représenter les quartiers urbains à un temps donné, et pourraient être caractérisés par un ensemble d'attributs ou d'événements. Ici, les arcs représenteraient la propagation du virus d'une zone à une autre sur des temps consécutifs. Si l'on considère maintenant l'analyse de l'érosion des sols, les sommets peuvent représenter différents objets géologiques, comme les ravines ou lavakas, observés à une date donnée. Leurs caractéristiques seraient exprimées par des attributs, et les arcs seraient utilisés pour symboliser les événements géologiques comme la fusion ou la division (voir figure 1.3).

Fouiller des graphes attribués mène à une explosion combinatoire : l'espace de recherche porte à la fois sur les graphes et les attributs. Peu d'études ont considéré les graphes attribués, encore moins les DAG attribués. MIYOSHI *et al.* (2009) fouillent un graphe étiqueté attribué, à savoir un graphe avec des étiquettes et des itemsets quantitatifs caractérisant les sommets. En gardant les étiquettes, la recherche de motifs fréquents est simplifiée et décomposée en deux étapes : extraction de graphes étiquetés puis d'itemsets. MOSER *et al.* (2009) ainsi que FUKUZAKI *et al.* (2010) se concentrent sur la fouille de motifs cohésifs et de motifs partageant des itemsets, c'est-à-dire des motifs représentant des sous-graphes avec des attributs partagés. Extraire des motifs fréquents dans un unique graphe attribué n'a donc pas encore été étudié. Développer une approche prenant en compte les dynamiques complexes entre les objets (apparition, disparition, fusion dans l'espace et le temps) reste donc un défi.

La richesse d'information d'une telle modélisation rend difficile la tâche d'extraction, de par la complexité du domaine de motifs que l'on cherche à extraire. En effet, si ce domaine de motifs utilise plusieurs types de données (structurelles, ordonnées, non ordonnées), le nombre de combinaisons possibles (c'est-à-dire le nombre de motifs) est grandement accru. Il est donc souvent nécessaire de réduire le nombre de motifs trouvés en éliminant ceux ne

3. Première loi de la géographie selon TOBLER (1970), que l'on pourrait traduire en français par « Tout interagit avec tout, mais les objets proches ont plus de chance de le faire que les objets éloignés ».

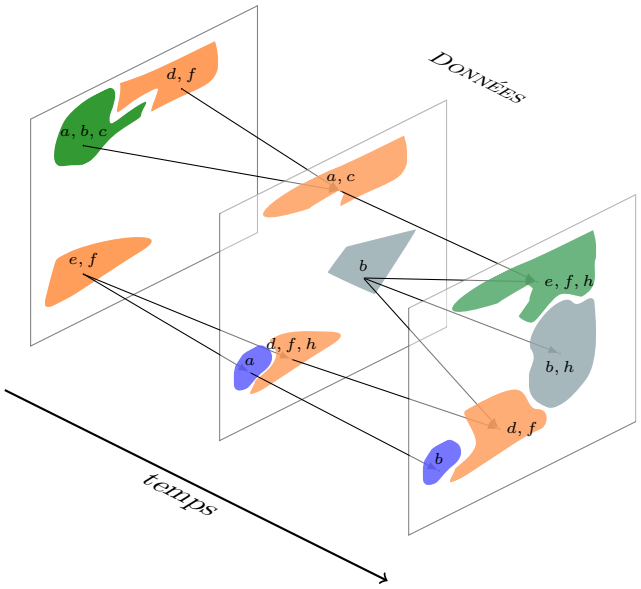


Figure 1.3 – Exemple de DAG attribué construit à partir de diverses images temporelles d'une même zone d'étude.

pouvant pas être intéressants pour l'utilisateur final.

2.2 Prise en compte de la connaissance du domaine pour améliorer la pertinence des motifs

Ainsi, dans ces contextes d'explosion des masses de données disponibles, il s'avère nécessaire d'utiliser les méthodes de fouille de données sous contraintes pour découvrir des motifs qui permettront ensuite d'assister la découverte de connaissances sur les phénomènes étudiés. L'ajout de contraintes lors de l'extraction de motifs a été largement étudié dans la littérature (McGarry, 2005). Différents domaines de motifs ont été considérés dans ce contexte, comme les ensembles (*itemsets*) et règles d'association dans des données booléennes, les motifs séquentiels ou les règles d'épisodes dans des (collections de) séquences, ou encore des sous-graphes dans des (collections de) graphes. La fouille sous contraintes a été étudiée pour permettre d'exprimer l'intérêt objectif ou subjectif des motifs, qu'il s'agisse de celui de motifs (Ng *et al.*, 1998 ; Pei *et al.*, 2001a)) ou de collections de motifs (Raedt et Zimmerman, 2007).

Les mesures objectives sont des mesures basées sur la fréquence, ou de manière plus générales sur des propriétés statistiques des motifs (suscitant ainsi l'usage de la contrainte populaire de la fréquence minimale introduite par Agrawal et Srikant (1994)). L'intérêt subjectif doit être spécifié grâces aux attentes des experts (voir, par exemple, Ng *et al.*, 1998 ; Pei *et al.*, 2001a ; Raedt et Zimmerman, 2007).

Ainsi, la notion subjective d'étonnement peut se spécifier en caractérisant tout de ce qui est attendu ou connu (Padmanabhan et Tuzhilin, 1998 ; Jaroszewicz et Simovici, 2004). De fait, la fouille sous contraintes permet non seulement de mieux gérer la pertinence

des motifs calculés mais encore, le plus souvent, d'exploiter les propriétés des contraintes (par exemple, des propriétés de monotonicité) pour réaliser des extractions complètes et efficaces (BOULICAUT et JEUDY, 2010). En effet, le passage à l'échelle de la fouille de données peut s'avérer délicat, puisque nous travaillons par définition sur de grandes masses de données. L'intégration du filtrage des motifs inintéressants durant le processus de fouille devient alors primordiale.

Travailler à l'intégration de la connaissance des experts dans les processus de fouille n'est pas nouveau (ANAND *et al.*, 1995 ; ANTUNES, 2008 ; DOMINGOS, 2007 ; CAO, 2010). Cette connaissance est souvent exprimée sous la forme de règles/contraintes expertes (de la forme *si ... alors ...*) définies manuellement. Ce type d'explicitation est difficile à obtenir et ne représente que très partiellement la connaissance du domaine : en pratique, elle reste limitée à quelques règles basiques.

L'utilisation de taxonomies ou ontologies améliore la définition de contraintes utiles et l'interprétation de motifs extraits (BRISSON *et al.*, 2005 ; PARENT *et al.*, 2013) même si leur acquisition auprès d'experts peut s'avérer très fastidieuse car soumise à l'interprétation humaine de concepts. On peut utiliser des modèles graphiques comme les réseaux bayésiens. Par exemple, JAROSZEWICZ *et al.* (2009) formalisent la connaissance essentielle d'experts sous forme de réseau bayésien de relations causales et de dépendances entre attributs. Ce réseau peut évoluer durant un procédé d'acquisition de connaissances. Il est possible d'exploiter de tels modèles durant la phase d'extraction de motifs plus intéressants. L'intérêt considéré est défini dans l'article comme « la divergence entre la fréquence attendue des motifs prévus par le modèle par rapport aux fréquences observées dans les données ».

D'un autre côté, les experts de domaines scientifiques variés (des géologues, des physiciens ou des épidémiologistes) expriment souvent une partie de leurs connaissances sur certains phénomènes au moyen de modèles mathématiques. Par exemple, les experts en érosion des sols ont développé des modèles permettant d'estimer le risque d'érosion en fonction d'un ensemble de paramètres environnementaux (par exemple la végétation, le type de sol, les précipitations et la pente (MORGAN, 2001 ; ATHERTON, 2005)). De même, les épidémiologistes ont développé des modèles mathématiques visant à estimer le nombre de personnes infectées par la Dengue en fonction du nombre d'habitants, du cycle de vie des moustiques, et des saisons (BAILEY, 1975 ; BURATTINI *et al.*, 2008 ; DE CASTRO MEDEIROS *et al.*, 2011). Même si ces modèles peuvent donner de bons résultats sur certaines zones d'études, leurs paramètres doivent être réajustés lorsque l'on change de zone. De plus, le nombre de variables en entrée est souvent limité. Par exemple, les modèles des épidémiologistes ne prennent que peu ou pas en compte l'influence de l'environnement proche (comme la géographie des quartiers, densité de population, conditions météorologiques fines). Ainsi, ces modèles sont fondamentalement des simplifications du réel. Or, du fait des progrès dans les capteurs et la collecte de données scientifiques, nous disposons souvent des valeurs de nombreux paramètres qui ne sont pas pris en compte dans les modèles disponibles (sans que l'on sache d'ailleurs si ces paramètres devraient l'être !).

2.3 Le cas de l'érosion en Nouvelle-Calédonie

La Nouvelle-Calédonie est un territoire français insulaire situé au large des côtes australiennes et néo-zélandaises. Sa nature géologique, son climat tropical ainsi que son activité anthropique entraînent certaines conséquences bien connues des géologues, dont l'érosion, qui altère à court et moyen termes l'occupation des sols. Ces modifications sont suffisamment intenses pour pouvoir changer les paysages, et par voie de conséquence les habitats naturels, l'équilibre du vivant, ainsi que l'organisation économique, urbaine et sociétale de la population. L'érosion peut en effet entraîner des glissements de terrains pouvant s'avérer dangereux – notamment à proximité de zones habitées, dégrader les paysages porteurs de valeur économique car touristiques, modifier la trajectoire des eaux de pluie et ainsi nécessiter la construction d'infrastructures d'adaptation coûteuses.

Bien que les mécanismes globaux (ruissellement, détachements de particules) impliqués dans les phénomènes érosifs sont connus par les experts, leurs modèles quantifiant cette érosion sont souvent centrés sur des cas particuliers, par exemple, sur des zones d'études agricoles, et leur généralisation à d'autres cas ne sont pas forcément adaptés ; il faut alors en développer de nouveaux. Pour cela, les experts font intervenir des coefficients arbitraires issus de leurs observations, lesquelles ne peuvent techniquement pas couvrir exhaustivement tous les cas d'érosion dénombrables sur la zone d'étude pouvant s'étendre sur plusieurs milliers de kilomètres carrés (la Nouvelle-Calédonie s'étalant sur plus de 18 000 km^2). L'automatisation, conjointe avec la puissance de calcul des ordinateurs et d'algorithmes performants, permet de déléguer la tâche fastidieuse de l'analyse exhaustive d'une certaine zone d'étude, tout en permettant de qualifier statistiquement les inférences que l'on peut en tirer. De plus, l'érosion étant un phénomène étalé dans le temps, l'analyse doit aussi porter sur plusieurs versions temporelles consécutives d'une même zone. La quantité de données spatiales se trouve ainsi multipliée par autant de tranches temporelles, et la complexité de l'analyse s'en trouve elle aussi fortement augmentée.

3. Contributions

3.1 Positionnement dans le processus d'ECD

Les principales contributions de ce mémoire se situent au niveau de la phase d'analyse. Comme expliqué précédemment, cela implique tout de même de réaliser en amont les étapes précédentes de l'ECD, même si la récupération et le pré-traitement des informations n'ont pas fait l'objet d'étude particulière. Ceci est aussi le cas de la restitution de l'information, qui a été nécessaire à l'évaluation de la qualité des motifs produits par nos algorithmes.

3.2 Organisation du mémoire

Dans le chapitre 2, nous détaillons l'état de l'art sur la fouille de données spatio-temporelles, qu'il s'agisse de méthodes dédiées à des problèmes spécifiques ou de méthodes génériques adaptables. Nous mettons en lumière le fait qu'aucun modèle ni méthode (spécifique ou générique) ne permet d'exprimer de façon complète certains phénomènes spatio-temporels, tels que l'érosion.

Le chapitre 3 présente les contributions de la thèse. Dans la première contribution, nous proposons un nouveau type de données pour modéliser des phénomènes spatio-temporels. Ce type de données consiste en une structure complexe (un DAG attribué) et suffisamment générale pour pouvoir être utile à d'autres modélisations. Nous proposons d'y extraire des chemins pondérés d'attributs afin d'observer les différentes évolutions de caractéristiques sur les sommets. Un formalisme a aussi été développé autour de cette base de données et de ce domaine de motifs. Afin de réduire le nombre de solutions extraites, une représentation condensée de l'ensemble des solutions est proposée ; le contexte de base de données sous forme de graphe unique diffère suffisamment des contextes classiques de fouille (bases de données transactionnelles) pour nécessiter un formalisme et une stratégie de fouille nouveaux. Cette représentation, proche de la fermeture, est décrite avec l'algorithme correspondant, dont la stratégie d'énumération des solutions repose sur deux étapes que nous décrirons précisément.

La deuxième contribution consiste à exploiter une connaissance experte particulière : les fonctions mathématiques à valeur dans $\mathbb{R}$ servant de modèle expert. Ces modèles présentent l'avantage d'être concis et d'être exprimés formellement dans la littérature du domaine de l'application étudiée. Utiliser ces modèles permet d'éviter de faire intervenir un utilisateur expert du domaine. À partir d'une mesure directement issue d'un tel modèle, nous exploitons une contrainte de seuil minimum. Nous dégageons des propriétés théoriques sur cette mesure afin d'intégrer la contrainte directement dans l'étape de fouille.

Afin de prouver l'intérêt de notre approche, nous élaborons dans le chapitre 4 deux scenarii de fouille de données. Cette étude de cas porte sur l'étude de l'érosion en Nouvelle-Calédonie, à partir de données issues d'images satellites et de données de terrain. Le premier scenario vise à rechercher les ensembles de caractéristiques radiométriques portées par des

pixels exprimant une forte érosion. Les expérimentations prouvent ainsi à la fois l'intérêt et l'efficacité de la contrainte de modèle développée dans la deuxième contribution. Dans le second scenario, nous exploitons les données spatio-temporelle issues d'images satellites prises à différentes dates. Nous décrivons toute la séquence de traitement menant à la génération d'un DAG attribué, dont nous extrayons les chemins pondérés condensés fréquents. Ces résultats sont restitués à l'utilisateur grâce à un logiciel de visualisation. Ceci nous permettent de retrouver des motifs traduisant un phénomène d'érosion, ou au contraire de re-végétalisation.

Enfin, nous concluons en résumant le travail effectué et en proposons de nombreuses perspectives.

Chapitre 2

La fouille de données spatio-temporelles : état de l'art

Sommaire

Dans ce chapitre sont présentées les études existantes portant sur la fouille de données spatio-temporelles. Les domaines de motif passés en revue sont soit dédiés aux données spatio-temporelles, soit des motifs plus généraux que l'on peut adapter aux données spatio-temporelles. Nous présenterons aussi des contraintes pouvant être associées à la recherche de ces motifs.

Ce mémoire porte sur l'étude des phénomènes spatio-temporels. Il en existe différents types, portant sur différents types d'objets d'étude. Par exemple, ces objets d'étude :

1. peuvent influer les uns sur les autres, selon la première loi de géographie de Tobler énoncée dans l'introduction.
2. peuvent être caractérisés par divers attributs. Ces caractéristiques sont des propriétés des objets, ou de leur environnement proche.
3. sont dynamiques par définition, à savoir qu'ils évoluent dans le temps. Le dynamisme peut porter à la fois sur :
   (a) les attributs de l'objet. Par exemple, si la couleur est l'une des propriétés d'un objet, celle-ci peut varier d'un temps à un autre.
   (b) l'emplacement de l'objet, modifiant sa position par rapport aux autres objets et donc modifiant l'influence qu'il peut avoir sur eux.
   (c) l'existence de l'objet. Dans certains cas, les objets d'étude peuvent apparaître ou disparaître entre deux temps consécutifs.
   (d) leur structure. Certains objets peuvent évoluer structurellement, à savoir fusionner pour ne former plus qu'un seul objet ou se scinder en plusieurs autres objets.

Les travaux sur la fouille de données spatio-temporelles sont souvent dédiés à certaines applications où plusieurs des considérations généralistes évoquées ci-dessus sont souvent prises en compte, bien qu'aucun des travaux revus ne les considère toutes à la fois. Par exemple, certains travaux se positionnent dans le cas où les objets d'étude sont spatialement statiques, dans le cas où ils ne sont composés que d'un seul attribut, et/ou le cas où ils sont constamment présents (aucune disparition, réapparition, fusion ou division) sur la fourchette temporelle choisie. Par exemple, lorsque l'on veut tracer des véhicules grâce à un échantillonnage régulier de leur position GPS (Giannotti et Pedreschi, 2008), ces véhicules existent et ne se transforment pas, et ce durant toute la durée de relevé des données. De fait, à aucun moment deux véhicules ne « fusionneront » ou ne s'assembleront pour n'en former qu'un seul. De plus, dans la mesure où seules les trajectoires sont étudiées, les véhicules ou groupes de véhicules ne portent aucune information supplémentaire pouvant les caractériser individuellement. Enfin, dans le cas de l'étude de la propagation de maladies à travers les différents quartiers d'une ville (Alatrista-Salas, 2013 ; Selmaoui-folcher *et al.*, 2013), ces quartiers ne se déplacent pas.

Nous nous concentrerons dans un premier temps sur les modélisations et les domaines de motifs portant explicitement sur du spatio-temporel. Puis, une fois que ces travaux seront démontrés non adaptés aux cas regroupant la totalité des considérations énoncées en début de chapitre, nous étudierons l'utilisation de représentations plus généralistes (c'est-à-dire, non spécifiques à la description de données spatio-temporelles). Nous décrirons les divers

domaines de motifs proposés dans la littérature et les techniques algorithmiques qui leur sont associées.

La seconde partie de ce chapitre porte sur l'utilisation de contraintes dans le processus de l'ECD, décrit dans l'introduction sur la figure 1.1 page 4. Il s'agit de créer des filtres permettant d'éliminer certains motifs jugés non intéressants et/ou redondants. La mesure d'intérêt mise en jeu peut être définie selon des propriétés statistiques d'un motif sur le jeu de données examiné – la mesure est alors dite « objective » car indépendante du cas d'étude. Elle peut aussi être définie selon le degré d'appartenance d'un motif à un modèle, schéma ou comportement attendu par les utilisateurs – le modèle dépendant du cas étudié, la mesure est alors qualifiée de « subjective ». Nous nous intéressons aussi à la réduction de l'ensemble des motifs trouvés via une représentation plus concise. Nous décrivons donc les principales contraintes existant dans la littérature et les techniques utilisées pour les exploiter directement dans les algorithmes de fouille, et non en simple post-traitement.

1. Données et approches de fouille spatio-temporelles

1.1 Domaines de motifs dédiés au spatio-temporel [1]

1.1.1 Suivi de trajectoires

L'un des domaines de fouille de données complètement dédié au spatio-temporel est la fouille de trajectoires, qui rencontre de nombreuses applications. L'analyse des objets en mouvement a également comme domaines d'applications la géographie socio-économique, le sport, l'analyse et le contrôle de la pêche, les prévisions météorologiques et l'analyse du mouvement (suivi des ouragans).

Ces données relatives aux déplacements, et à la mobilité en général (*mobility data*), se présentent généralement sous la forme de bases de données de trajectoires. Dans Mamoulis *et al.* (2004), Cao *et al.* (2005) et Giannotti *et al.* (2007), les auteurs définissent les trajectoires comme des objets en mouvement représentés par des séquences de tuples (l, t), où l est la localisation de l'objet au temps t. Autrement dit, une trajectoire est une collection de positions d'un même objet se déplaçant à différentes localisations spatiales.

La figure 2.1 représente schématiquement une base de données de trajectoires.

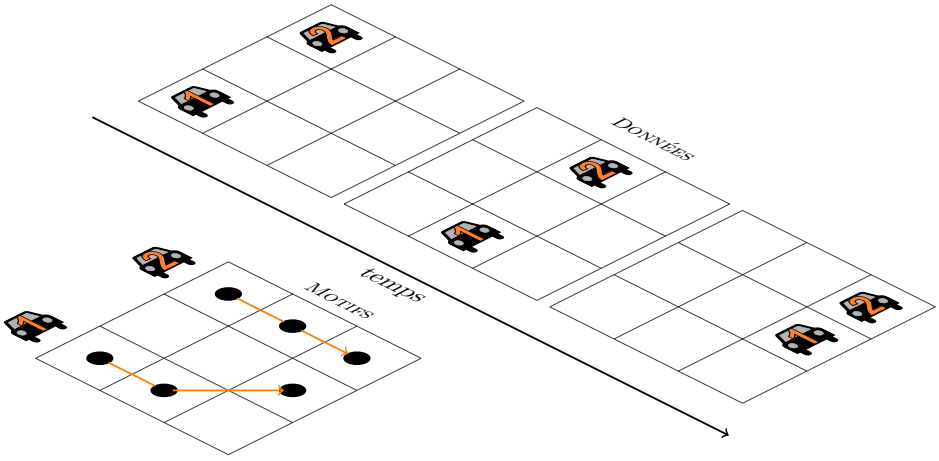


Figure 2.1 – Exemple de base de données de trajectoires de véhicules

Le nombre de trajectoires étant important, l'un des objectifs des méthodes d'extraction est de trouver les régularités les plus fréquentes. Face à cette problématique, beaucoup d'approches ont été proposées dans la littérature.

Mamoulis *et al.* (2004), Cao *et al.* (2005) et Cao *et al.* (2007) par exemple se sont intéressés à l'extraction de motifs périodiques dans une base de données de trajectoires.

1. La partie 1.1 de l'état de l'art est tirée de Selmaoui-Folcher *et al.* (2013) ainsi que du livrable 2.1 du projet ANR « FoSTEr » qui finance cette thèse.

Les objets étudiés (par exemple, des bus) ont la particularité de suivre approximativement la même route à intervalles de temps réguliers. Dans un premier temps, cette approche consiste à résumer l'ensemble des trajectoires d'un même objet par une seule séquence de segments. Les segments de trajectoires similaires sont regroupés en utilisant une fonction de similarité qui tient compte de la proximité spatiale, basée sur l'angle et la longueur spatiale des segments. Cette approche donne une meilleur abstraction des trajectoires et diminue la taille des données pour l'extraction. Contrairement à d'autres travaux, ces motifs intègrent aussi une notion de flou au niveau des localisations, ce qui permet d'extraire un motif même s'il ne se répète pas exactement au même endroit. Dans un second temps, un algorithme par niveaux est utilisé pour extraire les motifs fréquents. Cet algorithme dérivé de `A-Priori` a été optimisé grâce à l'utilisation d'une nouvelle structure de données (*substring tree*). Ce travail a été validé sur une base de données de trajectoires de bus, où chaque séquence correspondait aux déplacements d'un bus dans une journée.

Dans Fisher *et al.* (2005), les motifs étudiés sont des groupes d'objets partageant un type de mouvement (direction, vitesse) à une date donnée dans une certaine région de l'espace. Cinq types de motifs de trajectoires basées sur le mouvement, la direction et la localisation sont proposés (convergence, rencontre, troupeau, leadership et récurrence).

Les travaux présentés dans Gudmundsson *et al.* (2004) permettent de détecter les 4 premiers types de motifs définis dans Fisher *et al.* (2005) en utilisant des algorithmes de calcul approximatif. Les motifs spatio-temporels identifiés sont des sous-groupes d'objets ponctuels mobiles, avec des nombreux éléments localisés dans une région assez petite et présentant un mouvement similaire du point de vue de la direction, du but visé.

Hai *et al.* (2012) proposent d'exprimer les mouvements de groupe en tant qu'ensembles d'items. Pour chaque date, les auteurs regroupent les objets d'étude en *clusters* : un cluster représente un groupement d'objets ayant des propriétés similaires (par exemple, les objets d'un cluster sont spatialement proches les uns des autres). Un objet ne peut apparaître que dans un seul cluster à une date donnée. Comme la clusterisation est effectuée pour chaque date, un objet pourra appartenir à plusieurs clusters de dates différentes. On se retrouve ainsi dans le cas d'une base de données transactionnelle classique, dans le sens où les transactions représentent les objets d'étude, et les items représentent les différents clusters auxquels appartient chaque objet. Dans une telle base de données, Hai *et al.* montrent que la recherche d'itemsets fermés équivaut à rechercher des mouvements de groupe.

Tous ces travaux sur la fouille de trajectoires s'adressent à une problématique très spécifique. En effet, les objets sont identifiés de manière unique à travers tout l'intervalle de temps étudié. Ils ne disparaissent ni ne réapparaissent. Leur structure est d'ailleurs figée, au sens que ces objets ne fusionnent et ne se divisent pas. De plus, seule leur composante spatiale varie dans le temps ; aucune de leurs propriétés et/ou caractéristiques n'évoluent. En conclusion, bien qu'elle soit utile à de nombreuses applications spatio-temporelles, l'analyse de suivi de trajectoires s'adresse à des applications bien spécifiques, dont les hypothèses de départ sont trop fortes pour les problématiques considérées dans ce mémoire.

1.1.2 Introduction du temporel dans la recherche de motifs spatiaux.

Dans un certain nombre d'applications, l'objectif est différent de la fouille de trajectoires. En effet, il ne s'agit pas de suivre le déplacement d'objets précis en mouvement, mais d'étudier l'évolution et les interactions globales, dans l'espace et dans le temps, d'ensembles d'événements en fonction de leur environnement. Un exemple d'application est l'étude de l'évolution des quartiers d'une ville en fonction de différents facteurs tels que le nombre de touristes, le type d'événements socio-culturels, ou le type d'attractions à visiter à proximité. Dans cet exemple, il est évident que les quartiers ne se déplacent pas. Il ne s'agit donc pas d'étudier des trajectoires mais d'étudier comment évolue l'ensemble des quartiers en fonction de leurs caractéristiques et des événements qui s'y produisent.

Dans cette catégorie de méthodes, Celik *et al.* (2006) et Celik *et al.* (2008) ont généralisé le concept de co-localisation à des données spatio-temporelles. Initialement, les co-localisations (*colocations*) se focalisaient uniquement sur la dimension spatiale. Cette notion a été étudiée dans un grand nombre de travaux (Huang *et al.*, 2004 ; Qian *et al.*, 2009 ; Flouvat *et al.*, 2014a ; Selmaoui-Folcher *et al.*, 2011). Ces motifs spatiaux sont des ensembles de propriétés (ou types d'événements) fréquemment associés à des objets spatiaux voisins (c'est-à-dire dont les instances forment des cliques). Le motif {route, érosion, végétation faible} pourrait être un exemple de co-localisation associée à des zones où l'érosion serait étudiée. Il représenterait le fait que les propriétés « route », « érosion », et « végétation faible » sont souvent proches. Autrement dit, il y aurait une corrélation spatiale entre ces éléments.

Les co-localisations spatio-temporelles ont été définies comme une extension des co-localisations classiques afin de représenter des ensembles de propriétés associées à des objets voisins dans l'espace et dans le temps. Plus précisément, ces motifs sont tels que leurs instances sont spatialement proches pendant une fraction significative de temps. Ces contraintes ont été intégrées par l'intermédiaire d'une mesure d'intérêt monotone combinant prévalence spatiale et prévalence temporelle. Pour simplifier, seuls les motifs apparaissant un grand nombre de fois dans un grand nombre de pas temporels sont conservés. Par exemple, dans la figure 2.2, les motifs {érosion, piste} et {feu, vent, aire de repos} sont des co-localisations spatio-temporelles (les traits en pointillés représentent la relation spatiale). Toutefois, {feu, vent, aire de repos} aura une mesure d'intérêt plus forte que {érosion, piste} car il apparaît fréquemment dans un nombre de temps plus important ({érosion, piste} n'apparaît qu'une seule fois à chaque temps).

Le concept de co-localisation a aussi été étudié dans d'autres travaux. Par exemple, les travaux de Qian *et al.* (2009) se sont intéressés à l'extraction des *Spread Patterns of Spatio-Temporal Co-Occurrences over Zones* (SPCOZs). Ces motifs représentent la propagation (la « trajectoire » temporelle) de co-localisations spatiales. Qian *et al.* ont ainsi étendu les travaux sur les co-localisations afin d'intégrer la dimension temporelle. Plus précisément, ils suivent le déplacement d'éléments de propagation (*spread element*). Un élément de propagation est une co-localisation « fréquente » localisée et associée à une fenêtre temporelle. Dans la figure 2.3, la co-localisation fréquente {érosion, piste} associée à l'intervalle $[t_1, t_2]$ est un élément de propagation. Les éléments de propagation combinés deux à deux constituent des arbres représentant la propagation d'un motif : le *Spread Pattern Tree* (SP-Tree). La figure

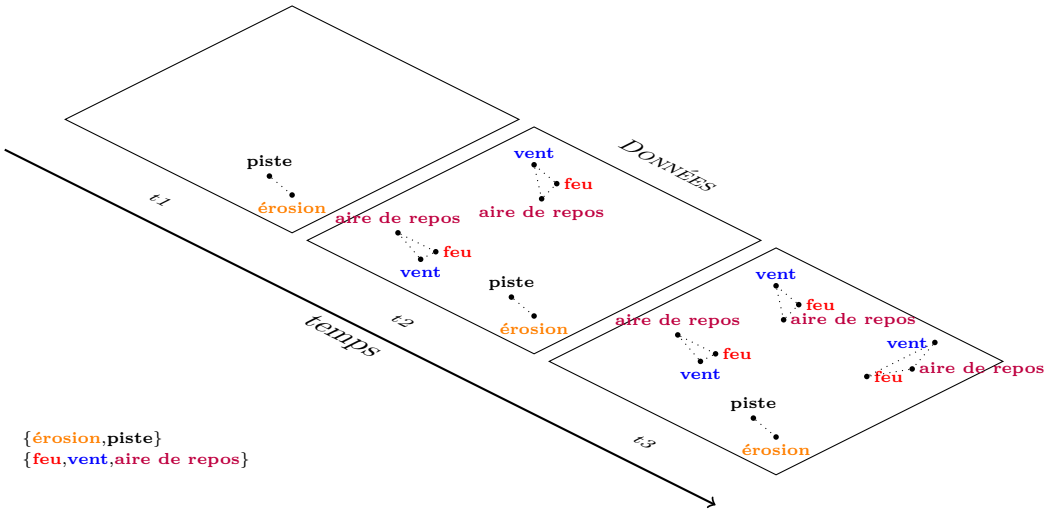


Figure 2.2 – Exemple de co-localisations spatio-temporelles

2.3 illustre deux exemples de motifs SPCOZ : le SP-Tree de {érosion, piste} et celui de {feu, vent, aire de repos}.

Yang *et al.* (2005) proposent un « *framework* » pour l'extraction de motifs spatiaux apparaissant fréquemment à différents temps, et une extension permettant de visualiser certaines évolutions. Ils ont validé leur approche sur un jeu de données étudiant l'évolution de molécules.

Les motifs étudiés, appelés *Spatial Object Association Pattern* (SOAP), peuvent être représentés sous la forme de graphes. Dans de tels graphes, chaque sommet correspond à une propriété, et chaque arête représente une relation de voisinage. Comme précisé par les auteurs, ce travail partage des similitudes avec les co-localisations car cette approche permet aussi d'extraire des ensembles de propriétés associées à des objets voisins, c'est-à-dire formant des cliques. Toutefois, à la différence de ces dernières, les auteurs considèrent des objets géométriques plutôt que des points pour les calculs de distances et de voisinage. De plus, ils permettent d'extraire trois autres types de configuration : étoile, séquence et *minLink*. Ce dernier type permet de définir des SOAPs plus généraux où seul le nombre minimum d'arêtes (*minLink*) associées à chaque sommet est fixé (c'est-à-dire le degré minimum des sommets). Par exemple, si $minLink = 1$, tous les SOAPs de type étoile, clique ou séquence sont générés. À noter que pour les séquences, les arêtes représentent une relation de voisinage et de direction. Par exemple, une arête (x, y) signifie « x est voisin et au dessus de y ».

Finalement, les auteurs montrent aussi comment utiliser (en post-traitement) les SOAPs fréquents pour visualiser l'évolution d'un même ensemble de propriétés F. Pour cela, ils définissent la notion d'épisodes comme un ensemble d'instances associées à un intervalle de temps où le motif est apparu puis a disparu. Les épisodes de tous les SOAPs fréquents associés à F sont recherchés (tous types confondus), puis ordonnés en fonction de leur date d'apparition. La séquence ainsi générée permet de visualiser l'évolution spatiale (étoile, clique, ...) de l'ensemble de propriétés étudiés. Toutefois, ce post-traitement ne permet pas de prendre en compte les évolutions de forme des objets étudiés, ainsi que les éventuelles relations de cause à effet. En effet, seul les changements globaux de « configuration » peuvent être observés.

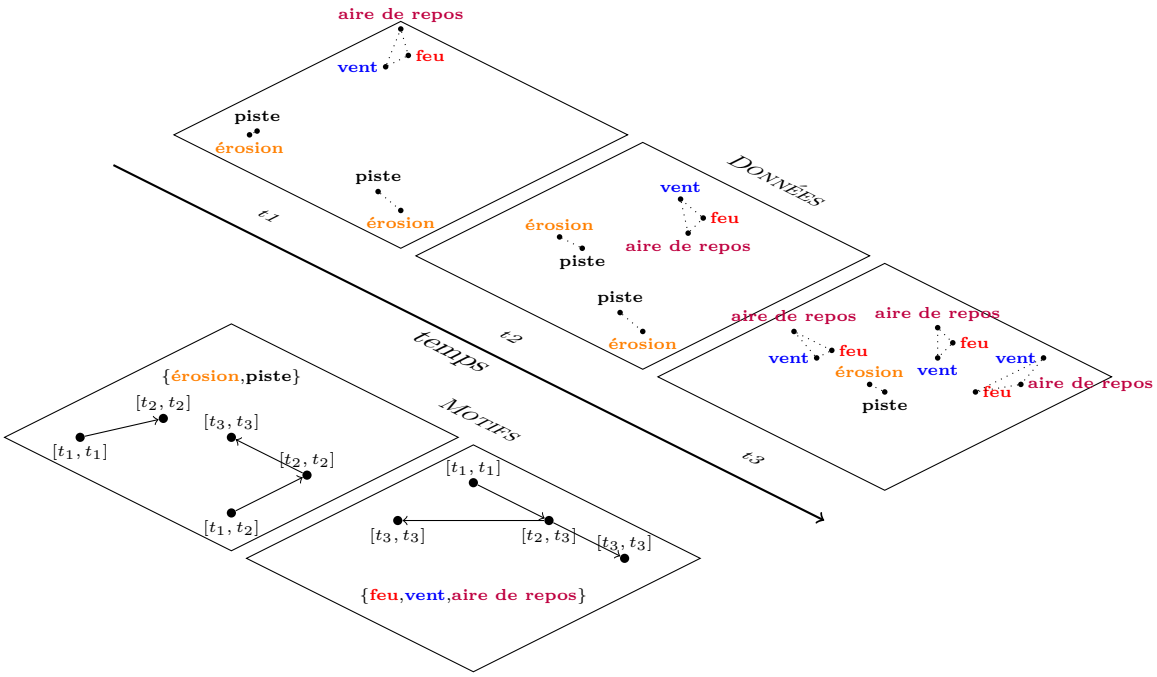


Figure 2.3 – Exemple de SPCOZ

De plus, le nombre d'épisodes peut être important pour un même ensemble de propriétés, ce qui rend très difficile l'analyse de la séquence. L'intégration d'informations supplémentaires telle que la météorologie est également délicate.

Même si ces méthodes sont vouées à étudier l'évolution (spatiale et temporelle) d'ensemble d'événements en fonction de leur environnement, elles se limitent à i) des caractérisations simples des événements : un seul attribut par événement servant d'identifiant à travers le temps (et qui n'évolue donc pas dans le temps), et ii) des relations entre événements temporellement statiques. Aucune de ces méthodes ne peut donc correspondre aux besoins de cette thèse fixés au début du chapitre.

1.1.3 Introduction du spatial dans la recherche de séquences temporelles.

De la même manière que les études présentées au paragraphe précédent, les travaux sur l'extraction des séquences, utilisées initialement pour rechercher des régularités temporelles, ont été appliqués et étendus au spatio-temporel. Par exemple, Tsoukatos et Gunopulos (2001) ont étendu les travaux sur les séquences d'itemsets afin d'extraire des séquences représentant l'évolution dans le temps de zones d'études (par exemple, des quartiers). La base de données considérée est constituée de séquences d'itemsets (ensembles de caractéristiques environnementales) représentant l'évolution temporelle des différentes zones. Un algorithme effectuant un parcours en profondeur de l'espace de recherche est ensuite appliqué pour extraire les sous-séquences les plus fréquentes (c'est-à-dire celles apparaissant dans le plus de zones). La figure 2.4 illustre un exemple de séquences pouvant être extraites.

Ces auteurs ont également proposé une approche pour extraire les séquences fréquentes à une granularité spatiale plus élevée (par exemple, à l'échelle d'une région) en exploitant

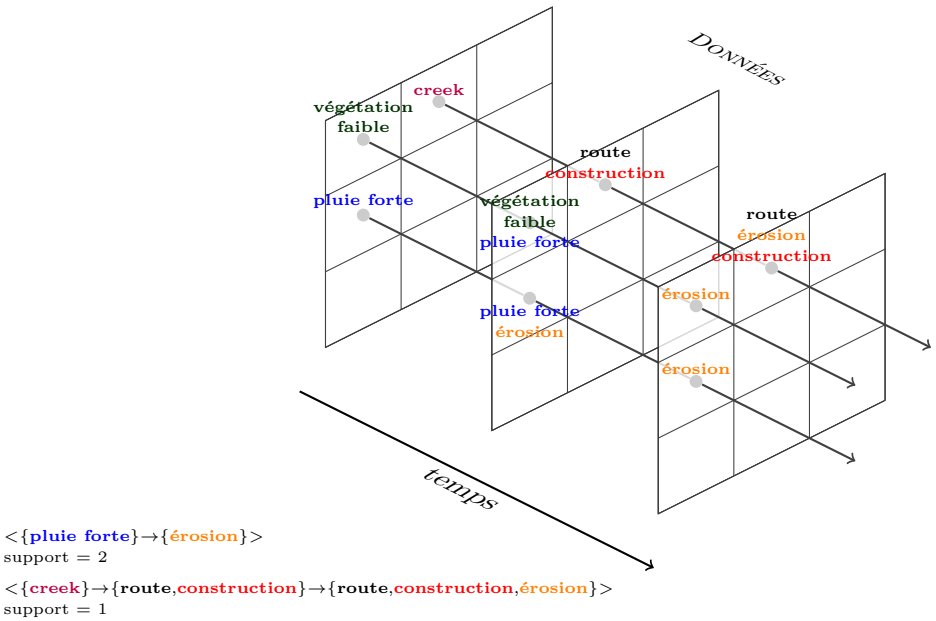


Figure 2.4 – Exemple de séquences représentant l'évolution de zones

les séquences fréquentes trouvées à une granularité plus faible (par exemple, à l'échelle d'une ville). Ils exploitent pour cela le fait que les séquences extraites à un niveau plus faible resteront fréquentes à un niveau de granularité plus élevé. La méthode recherche alors uniquement de nouvelles séquences fréquentes issues de l'agrégation spatiale.

Dans Wang *et al.* (2004), les auteurs se focalisent sur l'extraction de séquences représentant la propagation spatio-temporelle d'événements dans des fenêtres temporelles prédéfinies. Dans cet objectif, ils découpent la dimension temporelle en fenêtres d'une taille donnée (par exemple, 15 jours), divisent l'espace sous la forme d'une grille, et introduisent le concept de *flow pattern*. Un flow pattern est une séquence d'ensembles d'événements de la forme $< E_1 \rightarrow E_2 \rightarrow ... \rightarrow E_k >$ où E_i est un ensemble d'événements de la forme $e(localisation)$, avec e un type d'événements (par exemple pluie, vent). Chaque ensemble d'événements est composé d'événements spatialement voisins apparaissant au même temps. Deux ensembles d'événements E_p et E_q sont consécutifs dans la séquence, si leurs événements appartiennent à la même fenêtre temporelle, s'ils sont tous voisins et apparaissent à deux temps consécutifs. L'objectif de ce travail est de trouver les séquences d'événements apparaissant fréquemment, c'est-à-dire celles apparaissant un grand nombre de fois suivant les mêmes localisations. La figure 2.5 montre un exemple de jeu de données et quelques flow patterns avec leur support (dans cet exemple, deux événements sont voisins si leur distance euclidienne est inférieure ou égale à 1). Les événements sont localisés par des coordonnées (X, Y), tels que l'événement pluie(0,1) au temps t_1. Le motif <{piste(0,0), pluie(0,1)}→ {aire de repos(1,2), feu(1,2)}> a un support de 0 car les événements {aire de repos(1,2), feu(1,2)} ne sont pas voisins de tous les événements de {piste(0,0), pluie(0,1)} (aire de repos et feu du temps t_2 ne sont pas voisins de piste du temps t_1). Il est aussi intéressant de noter que les motifs <{piste(0,0),pluie(0,1)}→ {érosion(1,0)}> et <{piste(0,0),pluie(1,1)}→

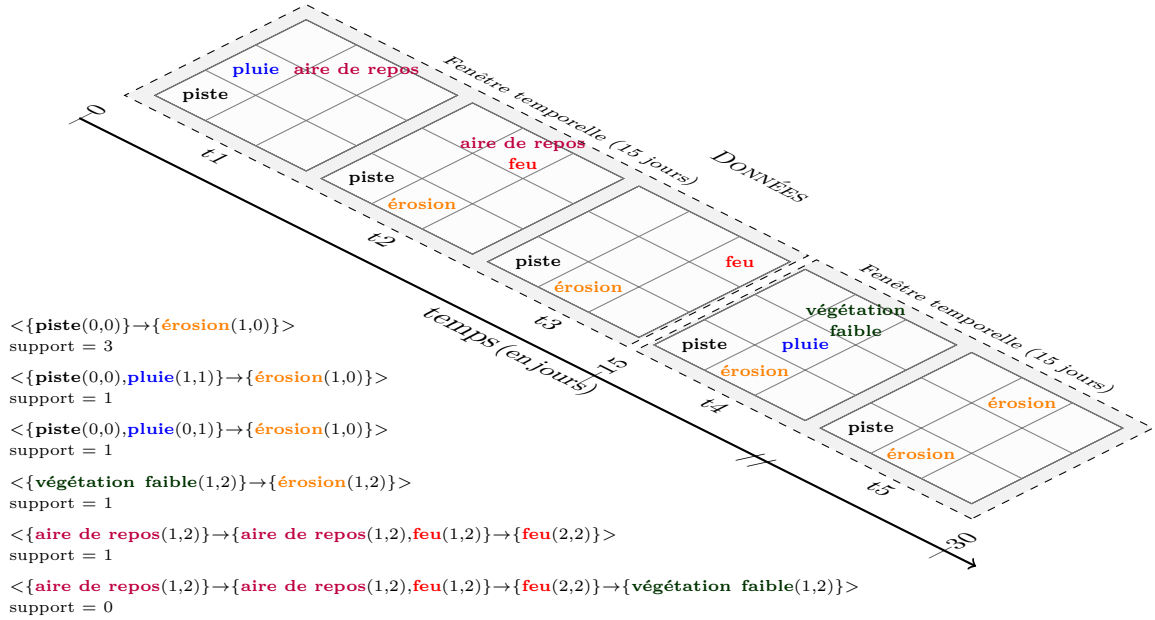


Figure 2.5 – Exemple de *flow patterns*

{érosion(1,0)}> sont considérés comme deux motifs différents bien que représentant des phénomènes similaires (seule la localisation de l'événement pluie est légèrement différente). Autre point important, le support du motif <{aire de repos(1,2)}→ {aire de repos(1,2), feu(1,2)}→ {feu(2,2)}→ {végétation faible(1,2)}> est égal à 0 car il n'est pas inclus dans une unique fenêtre temporelle. Pour les mêmes raisons, le flow pattern <{piste(0,0)}→ {érosion(1,0)}> apparaît 3 fois et non 4 fois .

Pour extraire ces motifs, les auteurs adaptent des techniques d'extraction de motifs séquentiels classiques. Plus précisément, leur algorithme applique une stratégie par niveaux pour trouver les séquences de taille 1 et 2, puis utilisent les motifs fréquents trouvés comme point de départ à un parcours en profondeur de l'espace de recherche.

Wang *et al.* (2005) étendent cette notion et définissent les motifs spatio-temporels généralisés (*generalized spatio-temporal pattern*) comme des séquences de *relative eventsets*. Un *relative eventset* est un ensemble d'événements dont la localisation est remplacée par un positionnement relatif à une localisation de référence. Un motif spatio-temporel généralisé est fréquent s'il a au moins $t-minsup$ (support temporel) occurrences dans le temps et qu'il a au moins $s-minsup$ (support spatial) occurrences dans l'espace (les localisations peuvent être différentes mais la localisation relative doit être identique). Pour extraire ces motifs, les auteurs proposent un nouvel algorithme appelé `GenSTMiner` qui utilise une approche dérivée de `PrefixSpan` (Pei *et al.*, 2001b).

Huang *et al.* (2008) se sont concentrés sur le problème d'extraction de séquences de propriétés représentant la propagation de certains types d'événements. Ces séquences sont de la forme $< f_1 \rightarrow f_2 \rightarrow ... \rightarrow f_k >$, où f_i est un type d'événements. Cette approche permet donc d'étudier la propagation d'événements pris individuellement (sans prendre en compte leur environnement). Ce modèle considère deux événements consécutifs s'ils sont

spatialement proches (distance euclidienne inférieure à un seuil donné) et apparaissent dans la même fenêtre temporelle. Les auteurs ont également étudié d'autres relations de voisinage dépendant du temps. Ces relations permettent de représenter un rétrécissement de la zone d'influence d'un événement (son voisinage) au fur et à mesure que le temps passe (c'est ce qui passe lors de la propagation d'une maladie infectieuse).

Les auteurs proposent aussi une nouvelle mesure d'intérêt pour ces séquences, appelée *sequence index*. Elle s'appuie sur des travaux antérieurs (Cressie, 1993) exploitant des statistiques spatiales pour étudier l'indépendance de phénomènes. Pour extraire ces séquences, les auteurs proposent un nouvel algorithme `Slicing-STS-Miner` basé sur un traitement incrémental des différentes fenêtres temporelles et une extension des séquences à chaque étape. Pour cela, ils s'appuient sur une propriété du *sequence index* : si une séquence est intéressante, toutes les sous-séquences ayant le même préfixe sont intéressantes.

Alatrista-Salas *et al.* (2012) proposent un domaine de motif appelé « motif spatio-séquentiel », afin de non seulement représenter l'évolution de plusieurs propriétés décrivant une zone, mais aussi de prendre en compte leur voisinage. Comme son nom l'indique, un tel motif permet d'ajouter une information spatiale aux motifs séquentiels traditionnels. Ainsi, dans les séquences de la forme $< f_1 \rightarrow f_2 \rightarrow ... \rightarrow f_k >$, chaque f_i représentera non pas un seul attribut a_i comme précédemment, mais plusieurs attributs caractérisant la zone et les zones voisines. Les auteurs introduisent pour cela un opérateur spatial noté $\bullet$. On aura alors $f_i = \{a_1, a_2, ..., a_i\} \bullet \{b_1, b_2, ..., b_j\}$ où les a_i et b_j représentent respectivement les attributs de l'objet d'étude et ceux de ses voisins. Ces motifs sont donc plus riches que ceux présentés précédemment (multiples attributs, information spatiale). Les auteurs montrent l'intérêt de ces motifs sur des problèmes liés à la Dengue ainsi que sur des données hydrologiques.

Toutefois, comme pour chacun des motifs dérivés des séquences temporelles, chaque objet d'étude est clairement identifiable à chaque pas de temps, et reste géographiquement statique. Leurs éventuelles évolutions plus complexes (telles qu'une division ou une fusion entre plusieurs objets, ou encore une propagation de caractéristiques vers d'autres zones) ne peuvent pas être prises en compte. L'utilisation de structures plus complexes (telles que des graphes) semble nécessaire pour pouvoir exprimer cette information que recèlent les données. Des méthodes d'analyse de graphes ont déjà été proposées dans la littérature, indépendamment du domaine d'application qui nous intéresse (le spatio-temporel). La partie qui suit présente ces différentes méthodes généralistes que l'on peut adapter à nos besoins.

1.2 Utilisation de structures arborescentes et de graphes

Parmi les approches utilisables pour le spatio-temporel, il en existe des plus généralistes – c'est-à-dire, non dédiées à la fouille spatio-temporelle. De par leur généricité, certaines peuvent être adaptables éventuellement à un contexte spatio-temporel. Nous pouvons distinguer, entre autres :

— les méthodes de diffusion d'information, qui intègrent déjà une notion de dynamique (et donc de temps), mais qui sont très peu liées à l'extraction de motifs ;

— les méthodes de fouilles sur des données structurées (notamment des graphes) qui furent développées au départ pour des applications n'ayant pas de composante spatio-

temporelle ; on peut citer entre autres les études d'arbres XML (Zaki, 2002), de recherche de molécules (Borgelt et Berthold, 2002), ou de structures communautaires.

1.2.1 Propagation de phénomènes et diffusion d'information

On peut aborder certains problèmes faisant intervenir une dynamique interne sous forme d'équations différentielles. C'est ce que font par exemple les épidémiologistes (Bailey, 1975) lorsqu'ils essayent d'estimer comment évoluera l'emprise d'une maladie sur une population, en sachant que le nombre d'infectés à temps t dépendra de ce même nombre à $t-1$ le temps précédent, ainsi que du nombre d'individus sains. Par exemple, le modèle SIR, qui consiste en équations différentielles, considère qu'un individu passe d'un état sain (S) à un état infecté (I) et contagieux, puis finit par un état final de rémission (R) dans lequel il est soigné et immunisé [2]. Le modèle utilise les variables $S(t)$, $I(t)$ et $R(t)$ pour quantifier le nombre d'individus dans chacun des trois états à un temps t. À n'importe quel temps t, le nombre total d'individus N est fixe : $N = S(t) + I(t) + R(t)$. Le modèle est le suivant :

$$\frac{dS(t)}{dt} = -\beta.S(t).I(t) \qquad \frac{dI(t)}{dt} = \beta.S(t).I(t) - \gamma.I(t) \qquad \frac{dR(t)}{dt} = \gamma.I(t)$$

où β et γ sont des paramètres propres à la maladie étudiée. Ce modèle inclue bien le temporel, mais ne donne qu'une vue macroscopique du problème : il n'est pas question d'étudier à l'échelle de l'individu quelles sont les causes d'une telle propagation. Or, nous savons que la transmission de pathologies virales dépend de la proximité des individus. La prise en compte du spatial est donc primordiale, et nécessite d'utiliser une échelle plus fine, par exemple en découpant la ville en quartiers, voire en blocs de maisons, puis en prenant en compte l'environnement.

Les graphes permettent naturellement de faire une telle modélisation : en faisant l'analogie entre la transmission d'un virus et celle d'opinions (Kimura *et al.*, 2009), les informaticiens se sont penchés sur les réseaux sociaux, où chaque action d'une personne peut influencer celles avec laquelle elle est connectée. Kempe *et al.* (2003) et Goyal *et al.* (2011) ont ainsi cherché à savoir quelles étaient, dans un réseau donné, les personnes influentes, c'est-à-dire les « graines » à partir desquelles une action est susceptible de se propager à travers un maximum de sommets du réseau. Certaines hypothèses sont émises sur la façon dont se propage l'information ; par exemple, le modèle en cascade (de l'anglais « *Information Cascade* ») établit entre autres qu'un sommet ne peut que « contaminer » un de ses voisins, et que cet état binaire est définitif. En faisant l'hypothèse d'un modèle de propagation, les méthodes de maximisation d'influence permettent donc seulement de répondre à la question « qui sont les personnes les plus influentes du réseau ? », mais pas « pourquoi le sont-elles ? ». Or l'objectif de cette thèse est pourtant de caractériser la ou les causes d'une telle diffusion dans un réseau. En clair, nous voulons partir sans hypothèses sur le modèle de propagation, puisque c'est justement une connaissance que nous ne détenons pas et que nous voudrions acquérir (en totalité ou en partie) grâce à la fouille de données. De plus, dans une telle modélisation, seul l'état des individus (les sommets) évolue dans le temps ; ceux-ci ne sont

2. Les individus n'ayant pas survécu à la maladie sont comptabilisés dans la variable $R(t)$

donc caractérisés que par un seul attribut. Les relations entre individus (les arêtes) restent statiques, et les sommets sont supposés exister sur toute la fourchette temporelle étudiée.

1.2.2 Les arbres

La recherche de sous-arbres dans un ensemble d'arbres (aussi appelé « forêt ») a fait de l'objet de plusieurs études (Zaki, 2002 ; Asai *et al.*, 2003 ; Zaki, 2004 ; Termier *et al.*, 2004 ; Balcázar *et al.*, 2010 ; Chehreghani, 2011 ; Pasquier *et al.*, 2013 ; Pasquier *et al.*, 2014), qu'il s'agisse de recherche d'arbres XML dans des pages web ou de recherche de polymères communs à des structures ARN (Acide Ribo-Nucléique). Les structures arborescentes peuvent cependant être utilisées pour d'autres applications. En l'occurrence, en remarquant que le terme « diffusion » connote une propagation jusqu'à un nombre de cibles supérieur au nombre de sources, les arbres peuvent apparaître pertinents pour la modélisation de tels phénomènes : un arbre est défini comme une structure où il n'existe qu'un seul sommet racine r, et qu'il n'existe qu'un seul chemin entre r et chacun des autres sommets.

On pourrait alors faire correspondre la racine avec le point d'émission originel, et les autre sommets (notamment les « feuilles », c'est-à-dire les sommets sans fils) avec les points touchés par le phénomène en jeu. La différence d'approches, entre cette modélisation et celle utilisée dans la maximisation d'information, est que l'on essaye d'exprimer tous les embranchements possibles de la contamination, car on ne connaît pas *a priori* le modèle de propagation. On essaye ensuite de faire émerger ce modèle inconnu à partir de caractéristiques observées dans les données structurées. L'approche proposée ici est l'inverse de celle utilisée pour la maximisation de propagation, pour laquelle on fait d'abord l'hypothèse d'un modèle de propagation, et à partir duquel on essaye ensuite de tirer certaines caractéristiques particulières (en l'occurrence, l'ensemble de sommets appelés « graines »). L'approche proposée dans ce paragraphe semble donc mieux correspondre aux besoins de cette thèse.

Il existe toutefois certains inconvénients à représenter des phénomènes sous forme d'arbres, car chaque sommet n'a qu'un seul sommet parent. Or il est possible que certains phénomènes puissent être engendrés par plusieurs causes, non nécessairement regroupées sur un seul objet. Par exemple, dans le cas de la propagation d'un virus dans une ville, on pourrait s'apercevoir qu'un quartier ne devient dangereux que si, une semaine avant, un quartier voisin Q_1 est contagieux et qu'un autre quartier voisin Q_2 abrite plusieurs vecteurs de la maladie (des moustiques par exemple).

De façon plus générale, il n'existe par définition qu'un seul chemin entre la racine et chacun des sommets ; il n'est donc pas possible d'avoir deux branches pointant sur le même objet, et donc de représenter par exemple l'un des liens illustrés par la figure 2.6. Seules certaines arêtes peuvent être gardées afin de respecter la structure arborescente (un exemple est donné avec les arêtes pleines), bien que d'autres arêtes soient tout aussi intéressantes (arêtes en pointillés). Une solution, étudiée dans une étude préliminaire (Selmaoui-Folcher et Flouvat, 2011 ; Sanhes *et al.*, 2012), serait de dupliquer l'objet-cible doublement pointé par deux sommets différents du même arbre. Similairement, deux arbres différents peuvent partager un même objet d'étude ; dans ce cas, cet objet sera exprimé plusieurs fois dans la modélisation, qui contiendra donc une redondance d'information. Comment alors, pendant la fouille, gérer les problèmes de comptage d'occurrence que cela implique ? Les auteurs se

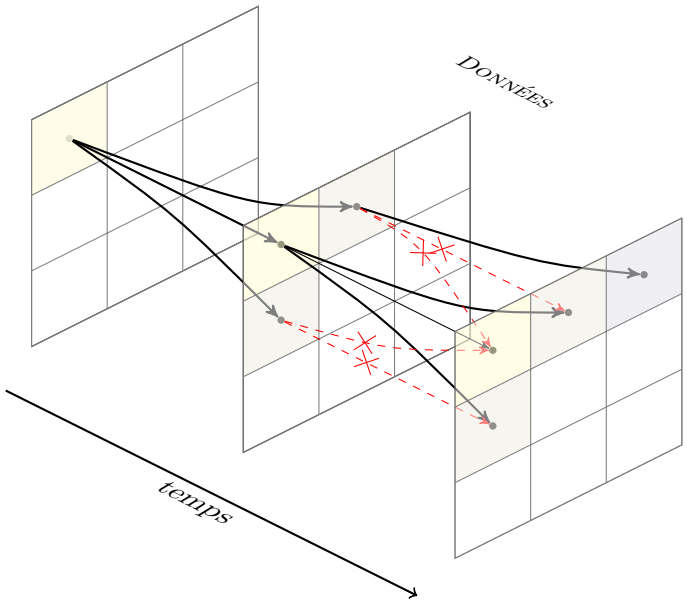


Figure 2.6 – Utilisation d'arbres pour la modélisation de phénomènes spatio-temporels et problèmes en découlant.

sont heurtés à ces exemples dans leur recherche de sous-arbres fréquents.

S'il est donc difficile (impossible ?) de gérer les occurrences de cette modélisation qui semble alors inadaptée, peut-être qu'une autre modélisation serait plus appropriée. Si l'on ne duplique pas les sommets en question, plusieurs chemins sont créés entre une racine et un sommet de la structure. Nous ne traitons alors plus des arbres, mais des graphes au sens général.

1.2.3 Les graphes

La recherche de motifs dans les graphes a vu le jour grâce aux besoins d'analyse de données structurelles plus complexes que les séquences ou les arbres ; par exemple, chimistes et biologistes étaient intéressés par la recherche de composantes chimiques (c'est-à-dire de sous-graphes) communes à certaines molécules (les graphes) (Dehaspe *et al.*, 1998 ; Borgelt et Berthold, 2002 ; Yan et Han, 2002 ; Poezevara *et al.*, 2011). Les types de graphes étudiés dans la littérature, ainsi que les domaines de motifs recherchés, sont nombreux. On peut, par exemple, distinguer les graphes labellisés des graphes attribués. Dans le premier cas, les sommets (ou arêtes) sont caractérisés par un seul attribut (ou label) ; dans le deuxième cas, plusieurs attributs peuvent être assignés à chacun des sommets, sans toutefois que leur nombre soit le même pour chaque sommet. Dans le cas de l'analyse de molécules, les sommets constituaient ainsi un élément chimique, et les arêtes, l'existence d'une liaison entre ces atomes. Dans ce contexte, il est important de noter que l'on examine un ensemble de graphes, parmi lesquels on essaye d'extraire des sous-ensembles de sommets labellisés connexes. Ce contexte est donc le même que la fouille classique d'itemsets ou de séquences, dans le sens où les données en entrée sont de type « transactionnel » (comme le montre la figure 2.7), et

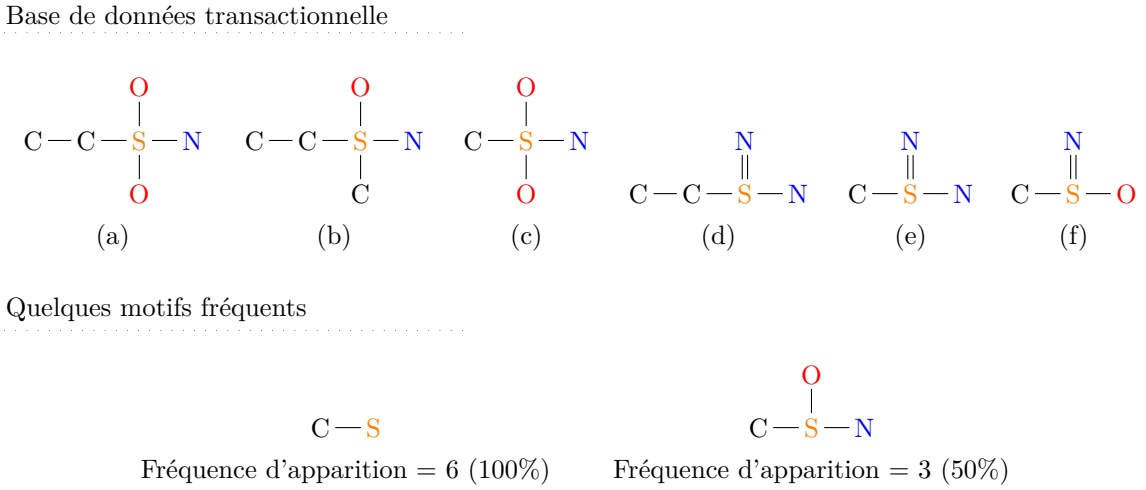


Figure 2.7 – Exemple de base de données transactionnelle de graphes. Ici, les graphes représentent des molécules. Un sommet est un atome, une arête est une liaison de covalence entre deux atomes [3].

où l'on recherche des motifs du même type que les données en entrée, en utilisant des relations d'inclusion bien définies. Lorsque, dans le cas de Poezevara *et al.* (2011), ces graphes sont catégorisés (par exemple, « molécule toxique » ou « molécule neutre »), rechercher les sous-graphes parmi un ensemble de graphes permet de mettre en évidence des structures qui sont récurrentes dans les molécules appartenant à une classe mais pas l'autre. Ainsi, on peut établir des sous-molécules discriminatoires et estimer, par exemple, la toxicité de certaines molécules. Les exemples de motifs donnés dans la figure 2.7 sont des sous-graphes dits « induits » : les sous-graphes extraits respectent les relations parents-fils des sommets des graphes de la base de données en entrée. D'autres auteurs, tels que Termier *et al.* (2007) et Pasquier *et al.* (2013) cherchent en plus des sous-graphes « encastrés » : dans ce cas, une relation parent-fils dans un tel motif peut aussi représenter une relation ancêtre-descendant dans un graphe en entrée.

Certains auteurs (Miyoshi *et al.*, 2009 ; Desmier *et al.*, 2013) proposent d'étudier des graphes dynamiques. Ce type de données peut s'apparenter à des séquences de graphes, à l'exception près que chaque sommet est identifié sur chacun des graphes ; seuls les arêtes ainsi que les labels ou attributs changent (évoluent ou apparaissent et disparaissent) dans le temps. En pratique, Miyoshi *et al.* (2009) supposent les arêtes fixes, et fouillent ainsi un seul grand graphe labellisé attribué, à savoir un graphe dont chaque sommet possède un label l, ainsi que plusieurs items $i_1, i_2, ..., i_n$ avec $l \notin \{i_k\}_{1 \leq k \leq n}$. Cet ensemble d'items regroupe tous les items présents sur le sommet à travers tous les pas de temps. En gardant les labels dans un ensemble à part, la recherche de motifs fréquents est simplifiée et décomposée en deux étapes : extraction de graphes labellisés, puis d'itemsets sur les sous-graphes trouvés. La fouille d'itemsets à l'étape suivante ne prend en compte aucune information structurelle. Leur approche ne gère donc pas conjointement la fouille de sous-graphes et la combinatoire

3. Exemple tiré de Borgelt et Berthold, 2002.

induite par l'utilisation des itemsets dans les sommets.

Desmier (2014) propose d'étudier des graphes dynamiques particuliers : ces graphes comportent sur leurs sommets de multiples attributs à valeur numérique, variant dans le temps. L'ensemble des attributs est fixe, seules les valeurs varient. Les sommets sont par définition fixés pour toute la fourchette temporelle étudiée : les objets d'étude sont donc géographiquement statiques. En revanche, les arêtes entre sommets peuvent varier (apparition/disparition), et ainsi exprimer une évolution relationnelle. Dans Desmier *et al.* (2012) et Desmier *et al.* (2013), les auteurs cherchent les groupes de sommets ayant un voisinage similaire ou étant connectés, et dont les attributs varient de la même façon.

Comme nous l'avons déjà vu pour certains problèmes spatio-temporels, les objets d'intérêts à étudier peuvent se déplacer, se transformer, disparaître, s'influencer les uns les autres, en fonction de la distance spatiale qui les sépare. Le cas où les objets d'étude sont clairement identifiés, et où seul leur comportement spatial importe (fût-il un comportement de groupe), a déjà été étudié dans les travaux de suivi de trajectoires. Ici, nous nous intéressons d'avantage au cas où les objets d'études ne sont pas précisément identifiés d'un temps à un autre, et où leurs propriétés (pas forcément spatiales) peuvent varier d'un temps à un autre, éventuellement grâce à (ou à cause de) la proximité d'autres objets. En ce sens, la représentation d'un tel problème sous forme de graphes paraît tout autant adéquate que pour les travaux présentés précédemment. Cependant, le fait qu'un sommet à un temps t_i soit en relation avec un autre sommet au temps suivant t_{i+1} donne une certaine particularité à ces graphes, comme expliqué dans le paragraphe suivant.

1.2.4 Les graphes orientés acycliques

Vouloir insérer un contexte temporel dans la fouille de graphe nous amène à nous intéresser en particulier à l'étude des graphes orientés, où le temps peut être naturellement traduit par des arcs (dirigés) au lieu d'arêtes ; l'acyclisme d'un tel graphe temporel découle ensuite du caractère « causal » intrinsèque du temps. On appelle ces graphes des *Directed Acyclic Graph*s (DAGs)[4].

Par exemple, Mohan *et al.* (2010) étendent les travaux de Huang *et al.* (2008) présentés précédemment en étudiant des graphes orientés acycliques de types d'événements. Leurs motifs, appelés « motifs spatio-temporels en cascade », ne permettent pas de prendre en compte l'environnement proche d'un objet, tout comme le fait qu'un objet puisse être caractérisé par plusieurs propriétés. Ils introduisent aussi une nouvelle mesure d'intérêt appelée « *cascade participation index* » construite à partir de la mesure proposée dans Huang *et al.* (2004). Cette mesure est anti-monotone, propriété importante dans un algorithme par niveaux de type « A-Priori » pour extraire les motifs les plus intéressants.

L'exemple de la figure 2.8 page suivante illustre comment la structure de DAG découle naturellement des données observées : à un temps t_1, un ensemble d'objets d'étude sont délimités. Aux temps suivants t_2 et t_3, on identifie d'autres objets, sans pour autant être capable de dire s'ils sont nouveaux ou si ce sont les mêmes objets du temps précédent qui ont évolué. Leur apparition ou modification sont toutefois sûrement conditionnées par la

4. Graphes orientés acycliques

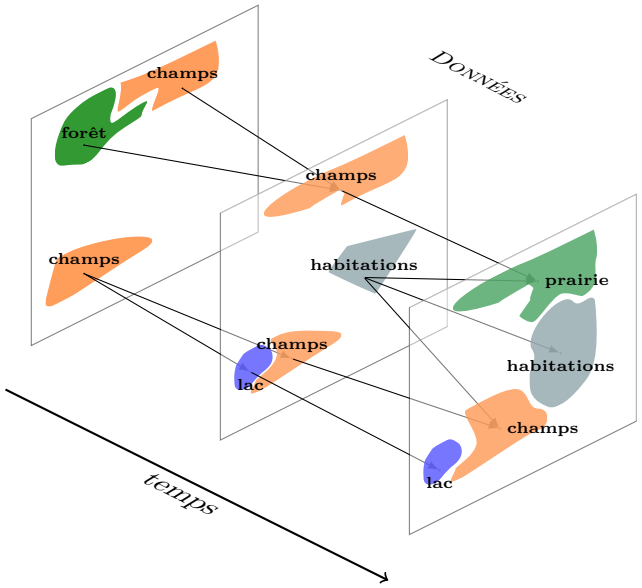


Figure 2.8 – Exemple de DAG construit à partir de diverses images temporelles d'une même zone d'étude. Un arc représente la proximité spatio-temporelle d'objets entre deux temps consécutifs.

présence d'objets voisins, toujours selon la loi de Tobler. Ainsi, on peut voir que du temps t_1 au temps t_2, une forêt et une zone de champs ont fusionné pour ne faire qu'un seul champs, qui ensuite a été transformé en prairie alors qu'il était à proximité d'habitats. On peut alors supposer que la proximité de ces habitats peut expliquer l'apparition d'une prairie là où se tenait auparavant un champs. C'est donc une information qu'il est important d'exprimer, et qui est contenue dans la structure de DAG nouvellement créée.

Plusieurs travaux se sont penchés sur l'étude de tels graphes, tels que Werth *et al.* (2008) proposant l'algorithme `Dagma`, Termier *et al.* (2007) proposant `DigDag`, Chen *et al.* (2004), ou bien, dans une moindre mesure, Günnemann et Seidl (2010) dont la méthode d'extraction peut s'adapter au contexte plus général des graphes (cycliques ou acycliques). Ces algorithmes extraient des sous-graphes, parmi un ensemble de graphes (c'est-à-dire une base de données transactionnelle). En outre, chaque sommet est caractérisé par une seule étiquette (ou *label*).

Termier *et al.* (2007) proposent de rechercher des sous-DAGs « encastrés » pour le cas particulier où chaque DAG de la base de données est constitué de labels distincts (il ne peut y avoir plusieurs fois le même label dans un DAG $\mathcal{D}$ en entrée). Ainsi, on peut ramener le problème à une fouille d'itemsets dans une base de données transactionnelle, en transformant chaque arête d'un DAG $\mathcal{D}$ en item : une fois trouvés les itemsets fréquents (en réalité les arêtes fréquentes), il est facile de les assembler pour reconstituer des sous-DAGs, puisque chaque sommet contient un label qui lui est unique. Par exemple, si l'on trouve pour les DAGs $\mathcal{D}_1$ et $\mathcal{D}_2$ les arêtes $A \rightarrowtail B$ et $B \rightarrowtail C$ formant un itemset fréquent, on peut directement en déduire que $A \rightarrowtail B \rightarrowtail C$ forme un sous-DAG fréquent, présent dans les DAGs $\mathcal{D}_1$ et $\mathcal{D}_2$. Comme le précisent les auteurs, cette hypothèse peut laisser penser que le problème est trivial ; ils

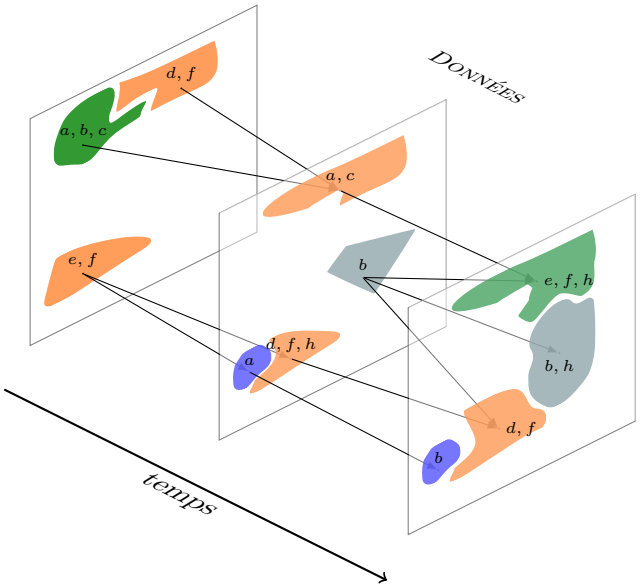


Figure 2.9 – Exemple de DAG attribué construit de la même façon que sur la figure 2.8 page précédente. Ici, les objets d'étude ne portent aucune sémantique, mais sont caractérisés par plusieurs attributs symbolisés par des lettres.

montrent toutefois que la recherche de sous-DAGs encastrés (et non induits) rend malgré tout le processus particulièrement long et gourmand en ressources, au point de ne pas pouvoir passer à l'échelle. Il faut en effet transformer en items non seulement les arêtes, mais aussi tous les chemins pour exprimer les relations ancêtre-descendant. Pour résoudre ce problème, les auteurs regroupent ces relations en *tiles* (des arbres de profondeur 1) présentant des propriétés (anti-)monotones grâce auxquelles une heuristique peut trouver les *tiles* maximaux contenant les motifs recherchés. L'algorithme `DigDag` recherche ainsi les sous-DAGs encastrés fréquents dans une collection de DAGs dont les labels sont uniques *par* DAG. En conséquence, chaque sous-DAG ne peut apparaître qu'une seule fois par DAG.

Chen *et al.* (2004) s'insèrent dans la même configuration : à partir d'une collection de DAGs, ils recherchent les sous-DAGs fréquents. Les auteurs se concentrent sur les sous-DAGs induits et non encastrés. Cependant, plusieurs attributs caractérisent cette fois chaque sommet, et ils ne sont pas nécessairement uniques dans un DAG donné. Ces DAGs sont appelés « DAGs attribués ». Le nombre de sous-DAGs que l'on peut en extraire est donc beaucoup plus grand : dans un ensemble de n items, on peut générer $2^n - 1$ itemsets différents. Cependant, dans cette configuration de base de données transactionnelle, l'utilisation de multiples attributs ne complique pas fondamentalement la tâche de recherche : comme l'expliquent les auteurs, on peut déjà dans un premier temps éliminer les items non fréquents. De plus, les auteurs contournent finalement le problème causé par l'explosion des combinaisons possibles, en se contentant de ne générer que les itemsets présents sur les sommets en entrée. La base de données initiale peut donc être ramenée à une collection de DAGs labellisés, et non attribués.

Dans les cas qui nous intéressent, nous aurions besoin de pouvoir analyser une structure où les sommets comportent plusieurs caractéristiques. Lorsque les objets ne peuvent pas

clairement être identifiés, mais que l'on peut tout de même les caractériser par divers attributs, les sommets deviennent ainsi attribués, et la structure créée est un DAG attribué. La figure 2.9 illustre les différences par rapport à la figure 2.8. Notons qu'il s'agit là d'un graphe unique. La configuration de fouille est donc sensiblement différente de celle où l'on possède une collection de graphes. En effet, dans cette dernière, la fréquence d'apparition d'un motif est simplement le nombre de transactions où apparaît ce motif. S'il apparaît plusieurs fois dans une même transaction, le motif n'est compté qu'une seule fois. Dans les travaux précédents, la question de trouver les diverses occurrences d'un motif dans une même transaction est donc mise à l'écart. Une telle simplification facilite grandement la fouille.

Dans le domaine de la compression de code, l'analyse de graphes peut aussi s'avérer utile. Werth *et al.* (2008) expriment ainsi les instructions processeur sous forme de sommets et leurs appels sous forme d'arcs ; les sous-graphes permettent de mettre en avant les morceaux de code fréquemment répétés. On peut alors les factoriser sous une seule instruction pour réduire la consommation d'énergie, qui est une ressource très limitée dans les systèmes embarqués. Werth *et al.* fouillent donc des collections de DAGs labellisés en considérant que plusieurs occurrences d'un sous-DAG peuvent apparaître dans le même DAG. On peut ainsi assimiler les différents DAGs donnés en entrée comme les diverses composantes connexes d'un seul grand graphe. Bien que les auteurs se concentrent sur la recherche de sous-DAGs induits seulement, la complexité du problème (montrée polynomiale) est suffisamment forte pour que les tests d'isomorphismes s'avèrent trop coûteux lors d'un passage à l'échelle. Afin de réduire le nombre et le coût de ces tests, les auteurs proposent une représentation canonique simple des DAGs, sous forme d'une chaîne de caractères. Ainsi, leur algorithme de recherche peut déterminer à chaque pas de temps si un nouveau sous-DAG est sous sa forme canonique. S'il ne l'est pas, la partie courante de l'espace de recherche est coupée, car ce sous-DAG sera retrouvé dans une autre branche de recherche. Bien que leur approche n'élimine pas tous les duplicats, elle en réduit une bonne portion, ainsi que le temps nécessaire pour fouiller l'ensemble de tous les sous-DAGs fréquents.

En outre, les auteurs évoquent – sans la traiter – une complexité supplémentaire induite par la présence de plusieurs sous-DAGs dans un même DAG en entrée : ces sous-DAGs peuvent en effet se superposer. Dans leur application, les données en entrée ne peuvent pas présenter un tel cas, et les problèmes de superposition ne sont donc pas abordés. Cependant, quand on fouille un graphe unique, où le support est fondé sur le nombre d'occurrences, il est nécessaire de gérer les problèmes de superposition.

Le type de données à étudier dans cette thèse porte plusieurs types d'information : on y retrouve de multiples attributs (sur les sommets) ainsi qu'une information structurelle (entre les sommets) variant dans le temps. Malheureusement, tous ces aspects ne sont pas traités simultanément dans la littérature.

De plus, même si une telle richesse sur les données apporte plus d'informations potentiellement intéressantes sur les motifs, elle rend d'autant plus difficile leur extraction. L'explosion du nombre de combinaisons possible entre attributs et arcs augmente considérablement le nombre de motifs potentiellement extraits. Il devient alors nécessaire de réduire ce nombre de solutions, en essayant de ne garder que ceux satisfaisant des critères dépendant de l'application.

2. Évaluation et amélioration de la pertinence de motifs

Les contraintes d'extraction sont apparues simultanément avec les premiers algorithmes de fouille de données (Agrawal et Srikant, 1994 ; Mannila *et al.*, 1997). Il s'agissait de réduire le volume d'information porté par la base de données analysée, en ne sélectionnant qu'une partie seulement des motifs que l'on pouvait en extraire. Le nombre de motifs extraits sans contraintes est en général bien supérieur à la taille de la base de données (voire supérieur). Par exemple, pour la recherche d'itemsets dans une base de données transactionnelles d'itemsets, le nombre de combinaisons est de $2^n - 1$ avec n le nombre d'items de la base. Ce nombre est potentiellement bien supérieur au nombre de transactions. L'utilisation de contraintes permettant de ne sélectionner qu'une partie des motifs devient ainsi primordiale.

Ces contraintes peuvent être appliquées durant la fouille ou *a posteriori* : les résultats sont les mêmes, et leur pertinence n'est donc pas remise en question. Cependant, les problèmes de passage à l'échelle ne permettent pas toujours de faire un post-traitement. Il devient alors crucial de pouvoir insérer ces contraintes dans le processus de fouille, afin d'économiser en mémoire et en temps d'exécution, qui sont des ressources limitées. De fait, la fouille sous contraintes permet non seulement d'avoir des motifs plus intéressants mais encore, le plus souvent, d'exploiter les propriétés des contraintes (par exemple, des propriétés de monotonicité) pour réaliser des extractions complètes et efficaces (Boulicaut et Jeudy, 2010). Il s'agit donc d'utiliser ces contraintes à la fois pour répondre aux besoins d'une analyse, mais aussi pour s'assurer que la tâche est applicable (en terme de passage à l'échelle) sur des données qui sont par définition nombreuses et complexes.

Mannila *et al.* (1997) ont proposé un cadre formel pour la fouille de données sous contraintes. Soient DB une base de données, $\mathcal{L}$ un langage fini pour exprimer des motifs (ou « mots ») ou définir des sous-groupes de données, et q un prédicat permettant d'évaluer si un motif $l \in \mathcal{L}$ est vrai, ou « intéressant » dans DB. Trouver $\mathcal{T}h(\mathcal{L}, DB, q)$[5], appelée « théorie de DB par rapport à $\mathcal{L}$ et q », consiste à trouver l'ensemble des motifs $\varphi \in \mathcal{L}$ qui satisfont un prédicat de sélection q sur une base de données DB. De façon formelle, $\mathcal{T}h(\mathcal{L}, DB, q) = \{\varphi \in \mathcal{L} \mid q(\varphi, DB) \text{ est vrai}\}$.

La difficulté d'utiliser les contraintes durant le processus de fouille dépend de la difficulté à énumérer les motifs φ et de la difficulté à calculer q. φ et q peuvent, selon le type de contraintes, présenter des propriétés théoriques qui permettent de parcourir l'espace de recherche de façon intelligente : certaines branches de cet espace de recherche pourront être évitées. Lorsqu'une contrainte est proposée, il est ainsi utile (voire nécessaire) d'en dégager des propriétés théoriques. Ainsi, on peut trouver plus efficacement les motifs d'un langage donné en les examinant selon un certain ordre. La notion de généralisation / spécialisation peut être formalisée par un ordre partiel $\preceq$ sur les motifs d'un langage $\mathcal{L}$: un motif φ sera plus général qu'un autre motif θ si $\varphi \preceq \theta$. Pour qu'un prédicat de sélection q soit monotone

5. La base de données DB est notée **r** dans l'article original

décroissant [6], il faut vérifier que $\varphi \preceq \theta \wedge q(DB, \theta) \Rightarrow q(DB, \varphi)$. Comme montré par Mannila *et al.*, ce type de contraintes peut ensuite être utilisé dans un algorithme d'extraction (les auteurs proposent une stratégie de recherche par niveaux, baptisée *levelwise algorithm*) pour élaguer l'espace de recherche, et ainsi améliorer les performances.

De cette propriété anti-monotone, Mannila *et al.* ont aussi dérivé la notion de « bordures », qui consiste à ne garder qu'un sous-ensemble des motifs satisfaisant le prédicat q. Par exemple, on peut s'intéresser exclusivement aux motifs les plus spécifiques selon $\preceq$. La « bordure positive » de $\mathcal{T}h(\mathcal{L}, DB, q)$ notée $\mathcal{B}d^+$ est la suivante :

$$\mathcal{B}d^+(\mathcal{T}h(\mathcal{L}, DB, q)) = \{\theta \in \mathcal{T}h(\mathcal{L}, DB, q) \mid \forall \varphi \in \mathcal{L} \text{ tel que } \theta \preceq \varphi, \text{ on a } \varphi \notin \mathcal{T}h(\mathcal{L}, DB, q)\}$$

Si l'on s'intéresse plutôt aux motifs les plus généraux de de $\mathcal{L}$ ne satisfaisant pas q, on prendra la « bordure négative » de $\mathcal{T}h(\mathcal{L}, DB, q)$ notée $\mathcal{B}d^-$:

$$\mathcal{B}d^-(\mathcal{T}h(\mathcal{L}, DB, q)) = \{\varphi \in \mathcal{L}\ \mathcal{T}h(\mathcal{L}, DB, q) \mid \forall \theta \in \mathcal{L} \text{ tel que } \theta \preceq \varphi, \text{ on a } \theta \in \mathcal{T}h(\mathcal{L}, DB, q)\}$$

Notons que la relation d'ordre $\preceq$ garantit la spécificité/généralité maximale des motifs de $\mathcal{L}$, mais ne s'applique pas à leur(s) mesure(s) éventuelle(s). En d'autres termes, il est possible à partir de $\mathcal{B}d^+(\mathcal{T}h(\mathcal{L}, DB, q))$ de retrouver exactement tous les motifs intéressants plus généraux ; il est cependant impossible de retrouver exactement les valeurs de mesure d'intérêt de ces motifs à partir de $\mathcal{B}d^+(\mathcal{T}h(\mathcal{L}, DB, q))$. La représentation condensée $\mathcal{B}d^+$ est ainsi qualifiée de « représentation avec perte ».

En outre, nous pouvons distinguer plusieurs types de contraintes :

1. Les contraintes indépendantes de toute application,
2. Les contraintes utilisant de la connaissance experte et qui dépendent ainsi de l'application étudiée

Parmi les contraintes dépendant de l'application, nous détaillerons les contraintes sémantiques sur le langage de motifs, les contraintes statistiques portant sur un prédicat de sélection q appliqué à un motif, et les contraintes de représentation condensées qui portent sur un ensemble de motifs extraits. Nous verrons que d'autres représentations condensées que les bordures ont donc été développées dans le but de pouvoir reconstituer exactement l'ensemble des motifs solutions de $\mathcal{T}h(\mathcal{L}, DB, q)$. On peut citer notamment l'ensemble des fermés introduit par Pasquier *et al.* (1999). Nous verrons cependant que la notion de fermeture n'est applicable que lorsque la base de données est de type « transactionnel ». En outre, bien que d'autres travaux aient été menés sur la réduction de la base de données elle-même (Raissi et Poncelet, 2007 ; Mathioudakis *et al.*, 2011), nous nous focalisons dans cette thèse sur la réduction de l'ensemble des motifs extractibles dans une base de données.

Enfin, nous discuterons des techniques développées pour contraindre la fouille de données à trouver des motifs estimés intéressants par rapport à une connaissance experte de référence.

6. le terme utilisé dans la littérature est très souvent « anti-monotone »

Ici, l'intérêt d'un motif pourra être évalué en fonction soit de son caractère étonnant, soit de sa conformité par rapport aux résultats attendus.

2.1 Contraintes indépendantes de toute application

2.1.1 Contraintes sémantiques sur le langage de motifs

Les contraintes sémantiques regroupent les contraintes portant sur le langage de motifs $\mathcal{L}$ recherché. Prenons par exemple le langage de motifs des itemsets : pour un ensemble d'items $\mathcal{I}$, on a $\mathcal{L}_{itemset} = \{I \in 2^{\mathcal{I}}\}$ avec n le nombre d'items binaires. On peut vouloir ne garder que les itemsets d'une taille minimale. Le langage de motifs exprimant cette contrainte est $\mathcal{L}_{itemset,k} = \{I \in 2^{\mathcal{I}} \mid |I| \geq k\}$ où k est la taille minimum de l'itemset.

Prenons un deuxième exemple. Lorsqu'on recherche des sous-séquences dans un ensemble de séquences, on peut vouloir se limiter à trouver l'ensemble des sous-séquences qui sont des chemins (c'est-à-dire dont le « *gap* » entre itemsets est nul). Prenons $gap = 0$ pour une BD constituée d'une séquence d'itemsets $S =< I_1 \rightarrow I_2 \rightarrow I_3 >$ avec $I_i \in \mathcal{L}_{itemset}$. On ne recherchera donc que les sous-séquences $< \mathcal{P}(I_1) \rightarrow \mathcal{P}(I_2) >$, $< \mathcal{P}(I_2) \rightarrow \mathcal{P}(I_3) >$ et $< \mathcal{P}(I_1) \rightarrow \mathcal{P}(I_2) \rightarrow \mathcal{P}(I_3) >$, mais pas $< \mathcal{P}(I_1) \rightarrow \mathcal{P}(I_3) >$[7]. Cette contrainte a été étendue pour la recherche de sous-arbres ou de sous-graphes ; on dit alors que l'on recherche les sous-arbres ou sous-graphes « induits[8] » (Chen *et al.*, 2004) à l'opposé des sous-arbres ou sous-graphes « encastrés[9] » (Termier *et al.*, 2007 ; Pasquier *et al.*, 2013 ; Pasquier *et al.*, 2014). On peut généraliser cette contrainte, en affirmant qu'elle revient à rechercher des structures dont les sommets sont connexes (Desmier *et al.*, 2012 ; Mougel *et al.*, 2012).

Ng *et al.* (1998) et Pei *et al.* (2001a) proposent des contraintes pour n'extraire que des itemsets dont chaque item est issu d'une même classe (contrainte de classe) ou dont chaque item a une valeur supérieure à un seuil donné. Par exemple, si l'on définit un ensemble de classes $\mathcal{C} = \{C_1, C_2, ..., C_{|\mathcal{C}|}\}$, le langage de motifs sera $\mathcal{L}_{itemset,\mathcal{C}} = \{I \in 2^{\mathcal{I}} \mid \exists C \in \mathcal{C}$ telle que $\forall i \in I, i \in C\}$.

2.1.2 Contraintes s'appuyant sur des mesures statistiques

Au delà des contraintes sémantiques, les contraintes imposées par des prédicats de sélection q portent sur l'évaluation de certaines propriétés d'un motif $\varphi \in \mathcal{L}$ par rapport à la base de données DB.

Dans leur article précurseur, Agrawal et Srikant (1994) ont posé le problème de la recherche des motifs fréquents : ne garder que les solutions apparaissant plus de σ fois dans la base de données d'itemsets (ici, un item représente un article acheté). L'intérêt de ne garder que les motifs fréquents est de pouvoir dégager les tendances principales d'un jeu de données. Les auteurs définissent deux concepts : une mesure d'intérêt (en l'occurrence, la fréquence d'apparition ou « support »), et une contrainte s'appuyant sur cette mesure (en l'occurrence, un seuil minimum de fréquence à atteindre pour chaque motif). Ces concepts ont été ensuite formalisés par Mannila *et al.* (1997) plus haut. Agrawal et Srikant donnent aussi une

7. $\mathcal{P}$ désigne tous les sous-ensembles d'un ensemble
8. « *induced* » dans la littérature anglophone
9. « *embedded* » dans la littérature anglophone

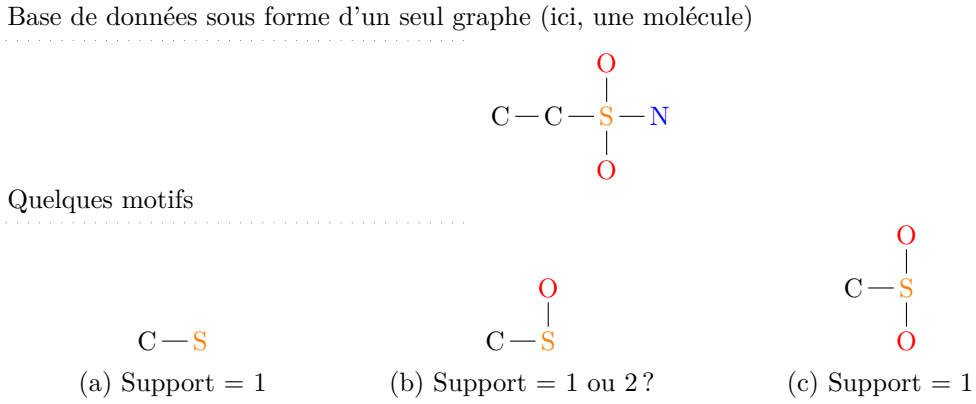
méthode efficace de recherche des motifs fréquents. Au lieu de tous les énumérer puis de ne garder que ceux satisfaisant cette contrainte de fréquence minimale, les auteurs proposent de ne parcourir qu'une partie de l'espace de recherche en évitant de générer les itemsets qui ne peuvent être fréquents. Pour cela, ils utilisent le comportement monotone de la fréquence : si un motif a un support supérieur à σ, alors tout motif plus général aura aussi un support supérieur à σ. Par contraposée, tout motif sera non fréquent si un motif plus général est non fréquent. Cette propriété est donc très pratique, et permet de rendre plus rapide la recherche de motifs, qu'il s'agisse d'itemsets, de séquences, ou de graphes. Dans le cas de la recherche d'itemsets (c'est-à-dire, $\varphi, \theta \in \mathcal{L}_{itemset}$) fréquents, s'assurer du caractère anti-monotone de la fréquence revient à l'équivalence suivante : $\varphi \preceq \theta \Leftrightarrow \varphi \subseteq \theta$.

De façon générale, on peut catégoriser les mesures s'appuyant sur la fréquence comme des mesures statistiques. McGarry (2005) liste plusieurs des mesures entrant dans cette catégorie, telles que le support, la confiance, le *lift* ou l'entropie pour les plus connues. Toutes ont en commun d'utiliser la fréquence, ou bien des probabilités calculées grâce à la fréquence du motif. Par exemple, le support d'un motif est sa fréquence d'apparition divisée par le nombre de transactions ; le support $support(A)$ donne ainsi une fréquence relative. Pour évaluer des règles d'association (basiquement, des inductions de type $itemset\, A \rightarrow itemset\, B$), la confiance $C = \frac{support(A \cup B)}{support(A)}$ donne le taux de vérification de la règle, tandis que le *lift* $L = \frac{C}{support(B)}$ donne le degré de dépendance probabiliste entre A et B (si $L = 1$, A et B ont des probabilités d'apparitions indépendantes). Notons qu'ici, le support d'un itemset A est considéré comme étant équivalent à sa probabilité d'apparition $P(A)$ dans la base de données, et que la confiance d'une règle $A \rightarrow B$ est considérée comme étant équivalente à $P(B|A)$. De façon générale, les contraintes statistiques, qui s'appuient sur ce genre de mesures, se résument souvent à la définition de seuils minimaux (ou maximaux) que ces mesures doivent dépasser (ou pas) afin que les motifs mesurés puissent être considérés intéressants.

La notion de fréquence paraît intuitive pour les bases de données transactionnelles : pour chaque transaction, on compte 1 si le motif recherché s'y trouve, 0 sinon ; elle l'est moins lorsque l'on recherche des sous-graphes dans une base de données composée d'un seul graphe, c'est-à-dire d'une seule transaction. Dans ce cas, la définition de fréquence n'est pas triviale, et peut avoir un impact important en terme de passage à l'échelle. En effet, le calcul de cette fréquence peut être complexe. De plus, ses propriétés théoriques peut être difficilement exploitables par les algorithmes d'extraction. Par exemple, dans la figure 2.10 page ci-contre, certaines occurrences d'un motif peuvent s'avérer contenir une partie commune. Cette partie apparaît donc globalement moins souvent que les occurrences en question, car elle est commune à plusieurs instances. Dans la figure 2.10, si le support du motif (b) est 2, la mesure n'est pas monotone par rapport à l'ordre de spécificité/généralité des graphes. Cette mesure ne peut pas être utilisée pour élaguer l'espace de recherche.

Bringmann et Nijssen (2008) se sont posé la question de savoir comment évaluer la fréquence d'un motif dans un unique graphe. Ils passent en revue les mesures utilisables dans ce contexte, ainsi que leurs complexités, qui sont beaucoup plus grandes que dans le cas des bases de données transactionnelles. Par exemple, Kuramochi et Karypis (2005) génèrent un graphe intermédiaire (le « *overlapping graph* ») des occurrences d'un motif, et

recherchent le plus grand ensemble de sommets complètement déconnectés les uns des autres. Cette mesure, appelée « *maximal independant set* », est utilisée par Miyoshi *et al.* (2009) présenté précédemment. Même si une telle mesure sélectionne les motifs et réduit ainsi la taille de l'ensemble des solutions, elle est NP-complète et pose donc des problèmes de passage à l'échelle. Face à ce problème, Bringmann et Nijssen proposent une autre mesure, celle du « *most restricted node* ». Il s'agit de trouver, dans le graphe fouillé, le plus petit nombre de sommets ou d'arêtes uniques qu'un sous-graphe recouvre. Prenons l'exemple du graphe unique donné par la figure 2.10, et le motif C—O—S (2.10b). Deux occurrences peuvent être trouvées pour ce motif (celle où l'oxygène O est orienté vers le haut, et celle où il est orienté vers le bas). L'atome d'oxygène du motif peut correspondre à deux atomes d'oxygène différents du graphe. Par contre, l'atome de carbone C du motif correspond dans tous les cas à un seul atome C du graphe (idem pour le soufre S). Le support du motif est donc $min(2, 1, 1) = 1$. Cette mesure présente l'avantage d'évaluer les motifs d'une façon simple à interpréter pour les utilisateurs. De plus, elle présente l'intérêt d'être monotone. La contrainte de seuil qui en découle pourra donc être utilisée pour améliorer le passage à l'échelle de l'extraction.



(a) Support = 1 (b) Support = 1 ou 2 ? (c) Support = 1

Figure 2.10 – Exemple de base de données sous forme d'un seul graphe.

2.1.3 Représentations condensées

Pour réduire encore la taille de l'ensemble des motifs solutions, les chercheurs ont proposé plusieurs représentations condensées des résultats d'une fouille, le but étant d'obtenir des résultats plus concis et donc plus facilement interprétables. Il s'agit d'éliminer toute redondance d'information dans un ensemble de motifs solutions. Parmi ces représentations, on peut distinguer entre autres les motifs maximaux, les motifs libres et les motifs clos. Certaines représentations condensées permettent de garder la totalité de l'information contenue par l'ensemble des solutions sans la dégrader ; on parle alors de représentation sans perte, ou représentation exacte. Il est à noter que, contrairement à un prédicat de sélection q qui s'applique sur un seul motif à la fois et indépendamment des autres motifs, les contraintes de non redondance d'information doivent considérer les motifs solutions dans leur ensemble.

Afin d'illustrer les différentes représentations condensées suivantes, nous prendrons l'exemple des bases de données transactionnelles d'itemsets $DB = (\mathcal{T}, \mathcal{I}, \mathcal{R})$, où $\mathcal{T}$ représente

les transactions, $\mathcal{I} = 2^n$ les items, et $\mathcal{R} \subseteq \mathcal{T} \times \mathcal{I}$ l'ensemble des relations décrivant la base : une relation $(t, i) \in \mathcal{R}$ est vraie si et seulement si l'item i est présent dans la transaction t.

Motifs maximaux (bordure positive) Les motifs maximaux sont nés du constat que dans l'ensemble des solutions, certains motifs sont plus spécifiques que d'autres. Dans le cas de la recherche d'itemsets, on peut se retrouver avec m itemsets intéressant $I^1, I^2, ..., I^m \in \mathcal{L}_{itemset}$, qui apparaissent tous dans la base de données (c'est-à-dire $\forall i \in I^1 \cup I^2 \cup ... \cup I^m, \exists t \in \mathcal{T} \mid (t, i) \in \mathcal{R}$), et tels qu'un de ces itemsets I^k regroupe tous les autres, c'est-à-dire $I^1 \subset I^k$ et $I^2 \subset I^k$ et , ..., et $I^m \subset I^k$. En d'autre termes, nous sommes dans le cas général où $I^1 \preceq I^k$, $I^2 \preceq I^k$, ..., $I^m \preceq I^k$. Seul I^k a besoin d'être conservé, puisqu'il représente à lui seul tous les autres.

Notons toutefois que I^k n'apparaît pas forcément dans exactement toutes les transactions que les autres itemsets $I^1, I^2, ..., I^m$. Si l'on ajoute une mesure basée sur ces transactions (c'est-à-dire des mesures statistiques), la mesure de chacun des $I^1, I^2, ..., I^m$ ne pourra pas être déduite de celle de I^k. Il sera nécessaire de reparcourir la base de données pour retrouver les valeurs de cette mesure. Cette condensation perd donc de l'information.

Motifs fermés Les motifs « clos » ou motifs « fermés » sont quant à eux une représentation condensée des solutions sans perte d'information. En observant que certains motifs étaient inclus dans d'autres (tout en conservant la même mesure d'intérêt), le travail précurseur de Pasquier *et al.* (1999) s'est attelé à supprimer les premiers de l'ensemble des solutions, et ce durant la fouille. Pour cela, ils partent du constat qu'il existe une correspondance de Galois entre itemsets et transactions. Plus précisément, reprenons notre exemple de base de données DB. Définissons une fonction $f : 2^{\mathcal{T}} \to 2^{\mathcal{I}}$ telle que $f(T) = \{i \in \mathcal{I} \mid \forall t \in \mathcal{T}, (t, i) \in \mathcal{R}\}$, qui associe à un ensemble de transactions T l'ensemble des items présents dans toutes ces transactions. La fonction $g : 2^{\mathcal{I}} \to 2^{\mathcal{T}}$ telle que $f(I) = \{t \in \mathcal{T} \mid \forall i \in \mathcal{I}, (t, i) \in \mathcal{R}\}$ associe quant à elle l'ensemble des transactions où apparaissent l'itemset I. Il est montré que (f, g) est une connexion de Galois, et $\phi = f \circ g : \mathcal{I} \to \mathcal{I}$ est un opérateur de clôture, à savoir qu'il est extensif, idempotent, et monotone croissant. Le fermé $\phi(I)$ d'un itemset I est le plus grand des sur-ensembles de I tel que les transactions le supportant sont les mêmes que celles qui supportent I. En conséquence, leurs fréquences d'apparition sont aussi identiques. Il est donc inutile de garder les itemsets fréquents qui ne sont pas des fermés, car l'information qu'ils portent se retrouve dans l'ensemble des fermés – qui, lui, est plus petit. Outre le fait de garder la totalité et l'intégrité de l'information contenue par l'ensemble des solutions, l'ensemble des motifs fermés est aussi la plus utilisée.

En outre, la recherche efficace de motifs condensés a été largement étudiée. L'algorithme `Close-by-One` de Kuznetsov et Obiedkov (2002) explore l'espace de recherche de façon à ne pas générer deux fois un motif clos, ce qui évite d'effectuer des vérifications coûteuses. La méthode la plus rapide pour les itemsets, nommée `LCM` (pour *Linear time Close itemset Mining*, Uno *et al.* (2003)), repose sur un parcours optimisé de l'espace de recherche exploitant le concept de « *core prefix* ». Il faut aussi pouvoir définir un ordre sur les items (l'ordre alphabétique, par exemple). Intuitivement, le *core prefix* d'un itemset fermé I sert de « noyau » d'extension pour générer un autre itemset fermé I'. Le *core prefix* d'un itemset I est le plus

petit préfixe (selon l'ordre sur les items) qui apparaît dans toutes les transactions où apparaît I. Par exemple, sur une base de donnée où $\mathcal{R} = \{(t_1, a), (t_1, c), (t_2, a), (t_2, b), (t_2, c)\}$, le *core prefix* de l'itemset abc est ab (on ne peut pas juste prendre a, car a apparaît en plus dans t_1). Dans une base de données où $\mathcal{R} = \{(t_1, b), (t_1, c), (t_2, a), (t_2, b), (t_2, c)\}$, le *core prefix* de abc est a.

Les propriétés de fermeture ne sont hélas valables que dans le contexte des bases de données transactionnelles. En effet, un item (ou attribut) portant une sémantique de présence (donc binaire), une transaction ne peut comporter plus d'une fois le même attribut. En conséquence, un sous-itemset ne peut apparaître qu'une fois par transaction. Cette considération est d'ailleurs retenue pour la plupart des autres méthodes de fouille de motifs fréquents (voir Yan *et al.*, 2003 pour les séquences d'itemsets, Yan et Han, 2003 pour les graphes d'itemsets), et c'est elle qui permet de réduire l'ensemble des transactions en un nombre : ce nombre est la valeur de la fréquence d'apparition d'un itemset, appelée « support ». Pourtant, dans le cas des séquences par exemple, on pourrait aisément considérer qu'une sous-séquence puisse apparaître plusieurs fois par transaction. Par exemple, dans la séquence $S =< a \rightarrow a \rightarrow b \rightarrow b >$, on pourrait supposer que le nombre d'apparitions de la sous-séquence $S' =< a \rightarrow b >$ peut être 1 ou bien 4, alors qu'il n'y a qu'une seule transaction S. De façon plus générale, utiliser le nombre de transactions comme nombre d'apparitions n'est plus valable dans une base de données sous forme d'un unique graphe. La fermeture ne peut donc pas être appliquée dans ce contexte, même si le concept de représentation condensée y garde tout son intérêt.

2.2 Contraintes dépendantes du domaine d'application

L'avantage des contraintes basées sur des mesures statistiques telles que présentées précédemment est qu'elles sont assez générales pour pouvoir être réutilisées sur n'importe quelle base de données transactionnelle. La mesure utilisée porte sur la présence ou l'absence des éléments recherchés durant la fouille, et non sur le sens de ces éléments. On pourrait alors qualifier ces mesures statistiques de mesures « objectives », tel que dans McGarry (2005). Bien entendu, utiliser telle ou telle valeur de support minimum dépendra de l'utilisateur, c'est-à-dire de l'expert de l'application étudiée. Définir, dans une contrainte, une valeur précise de support minimum n'a alors rien d'objectif : cette valeur est choisie en fonction de l'application.

Appliquer uniquement des contraintes utilisant des mesures objectives est toutefois insuffisant. Dans certains cas, il serait par exemple intéressant de filtrer certains motifs en fonction du sens qu'ils peuvent porter. En effet, pour parler de « découverte » de connaissances, encore faut-il être capable de dire si une connaissance (issue d'une information) est nouvelle ou non pour une application donnée. On aimerait donc que les motifs mettent en avant une information que l'expert ne possédait pas. De façon générale, on cherche donc une information étonnante. Il peut aussi être intéressant pour l'expert d'un domaine de s'assurer qu'un modèle de connaissances initial s'applique bien au scénario étudié. Dans ce cas, les motifs triviaux sont intéressants car rassurants : ils confortent la validité de la connaissance de référence. Ainsi, la notion subjective d'étonnement peut se spécifier en caractérisant tout de ce qui est attendu ou connu (Padmanabhan et Tuzhilin, 1998 ; Jaroszewicz et Simo-

vici, 2004). Tout ceci implique non seulement d'être capable d'utiliser l'information portée dans un motif, mais aussi de pouvoir comparer motifs et connaissances du domaine d'application. Or, il existe plusieurs façons de modéliser une connaissance experte, de même qu'il existe plusieurs domaines de motifs. L'insertion, dans le processus de fouille, de contraintes dérivées de la connaissances du domaine d'application n'est pas triviale non plus. Elle est pourtant indispensable si l'on souhaite garantir le passage à l'échelle.

Règles « si... alors... » L'une des façons d'exprimer une connaissance est de former des règles simples de type « si ..., alors ... ». Par exemple, « si *route* et *forte pente* alors *forte érosion* ». Ces règles sont définies manuellement. La simplicité de ces règles rend un tel modèle compréhensible et constructible par n'importe quel expert de n'importe quel domaine. Elle limite aussi hélas son utilisation à des modèles de connaissances simples : il est difficile de demander aux experts d'être exhaustifs quand le nombre de variables se fait conséquent. Cela représente beaucoup de travail de leur part, et une intervention humaine augmente les probabilités d'erreurs (humaines). Ainsi, en pratique, ce type d'explicitation ne représente que très partiellement la connaissance du domaine, et se limite souvent à quelques règles basiques.

Ontologies On peut exprimer des relations plus complexes à l'aide de taxonomies, ou plus généralement d'ontologies (Brisson *et al.*, 2005 ; Antunes, 2008), qui sont basiquement des graphes relationnels. Les contraintes qui y sont associées sont alors les mêmes que l'on peut définir sur des graphes orientés : connexité, distance, descendance, etc. Ces contraintes peuvent aussi être insérées directement dans le processus de fouille lors de la génération de motifs candidats : à chaque extension (ou agrandissement) d'un motif, on vérifie que ce changement satisfait les contraintes énoncées. Antunes (2008) donne un cadre formel à ces contraintes, et montre qu'elles peuvent être insérées dans n'importe quel type d'algorithme de fouille : soit de type « générer & tester » (souvent, des algorithmes effectuant des recherches en largeur tels que `A-Priori`), soit des algorithmes utilisant des projections directes (avec des recherches en profondeur, tels que `FP-Growth` ou `PrefixSpan`).

Autres modèles On peut aussi utiliser des modèles comme les réseaux bayésiens. Par exemple, Jaroszewicz *et al.* (2009) formalisent la connaissance experte en tant que réseau bayésien de relations causales et de dépendances entre attributs. Ce réseau peut évoluer durant le processus d'ecd. Il est possible d'exploiter un tel modèle pendant la phase de recherche de motifs afin d'extraire des motifs plus intéressants. La mesure d'intérêt considérée est définie ici comme la « divergence entre la fréquence des motifs prédits par le modèle par rapport à la fréquence observée dans les données. »

Kontonasios *et al.*, 2013 utilisent le principe d'« Entropie Maximale » pour modéliser des informations statistiques (moyenne, variance et histogrammes) connues par les experts sur une partie des données. Puis, ils proposent une mesure pour évaluer l'apport subjectif en information des motifs.

Finalement, ces diverses techniques demandent toutes l'intervention d'un utilisateur expert dans le domaine étudié, afin de pouvoir exprimer un ensemble de connaissances de

ce domaine dans le cadre de la recherche de motifs « intéressants ». Il serait avantageux d'arriver à se dispenser d'une telle intervention, et de pouvoir adopter un processus global d'exploitation de la connaissance experte.

Chapitre 3

Contributions

Sommaire

Comme discuté dans l'état de l'art, les différentes modélisations de problèmes spatio-temporels sont souvent dédiés à un type d'application. Pour l'étude de phénomènes spatio-temporels notamment, il est nécessaire de prendre en compte les relations entre objets d'études, leur comportement (fusion ou division), et leurs multiples caractéristiques.

D'autre part, les graphes deviennent omniprésents dans beaucoup de problèmes d'analyse de données. Récemment, des modèles de graphes plus riches ont été étudiés, où par exemple les sommets et arêtes sont attribués contrairement aux graphes simplement labellisés (un seul attribut). Ces graphes ont été baptisés « graphes attribués » ou « graphes associés à des itemsets » (« *itemset-associated graphs* » de Fukuzaki *et al.* (2010)), et permettent de prendre en compte les multiples caractéristiques des objets d'étude. Quand une notion temporelle intervient, les arêtes sont orientées (on parle d'arcs) et le graphe devient acyclique du fait de l'aspect causal du temps. Nous proposons donc l'utilisation d'un unique graphe orienté acyclique attribué (a-DAG) pour modéliser des phénomènes spatio-temporels : les sommets sont des objets spatiaux caractérisés par un ensemble d'attributs et/ou événements, et les arcs dénotent la proximité spatio-temporelle entre ces objets (par exemple, le voisinage spatial entre deux occurrences de deux temps consécutifs).

Le but de la fouille de données dans un tel graphe consiste à trouver les transitions (ou cheminements) de caractéristiques pouvant expliquer un phénomène particulier (lui-même caractérisé par un ensemble d'attributs). Cela revient à chercher dans un a-DAG les chemins fréquents d'attributs. L'utilisation d'ensemble d'attributs au lieu de labels uniques mène à une explosion combinatoire. Le nombre de motifs extraits est alors beaucoup trop grand pour être exploitable. En observant que certains motifs expriment la même information, il devient judicieux de vouloir réduire l'ensemble des solutions en ne générant pas les motifs porteurs d'une information redondante. Nous proposons donc d'extraire des chemins condensés.

Dans Sanhes *et al.* (2013a) et Sanhes *et al.* (2013b), nous proposons une méthode correcte pour extraire de tels motifs. Cependant, cette méthode a été prouvée incomplète après publication de l'article. Nous donnons donc dans ce chapitre un cadre formel plus étendu permettant de mettre en avant les raisons de l'incomplétude de l'algorithme proposé dans l'article. Nous donnons ainsi une solution permettant de trouver l'ensemble complet des motifs recherchés.

Dans un deuxième temps, nous proposons de développer des contraintes s'appuyant sur la connaissance établie d'un domaine d'étude. Les différentes méthodes existantes pour exprimer les connaissances d'experts nécessitent toujours l'intervention d'un utilisateur. En exploitant les connaissances formalisées par un modèle mathématique, non seulement nous profitons d'une connaissance élaborée (plus complexe, par exemple, que simples règles « si... alors... »), concise et déjà traduite dans un langage universel (langage mathématique), mais nous nous dispensons aussi d'impliquer en amont un utilisateur expert. Dans Flouvat *et al.* (2014c) et Flouvat *et al.* (2014b), nous proposons une nouvelle contrainte de seuil issue de modèles mathématiques.

Finalement, le chapitre qui suit est scindé en deux parties. Dans la première partie, notre contribution consiste à extraire les successions d'attributs dans un graphe orienté acyclique attribué, le tout sous une forme condensée pour éviter la redondance d'information et aussi gagner en synthèse et performances d'exécution. Un premier algorithme, dont la correction

est démontrée, est construit dans un contexte formel, où motifs et notions de « condensés » sont décrits et mis en perspectives par rapport à leurs apports respectifs et aux travaux existants. Cet algorithme s'appuie sur des équivalences avec la fouille de données dans une base de données n-aire d'items (Cerf *et al.*, 2008). Nous donnons ensuite un deuxième algorithme permettant de trouver l'ensemble complet des solutions. Les performances sont évaluées sur plusieurs jeux de données synthétiques et réelles.

La deuxième partie propose d'exploiter la connaissance experte d'un domaine d'étude qui est formalisée sous forme de modèle mathématique. Nous nous servons de ce modèle comme une mesure d'intérêt, à partir de laquelle nous dérivons une contrainte de seuil minimum. Cette contrainte permet de filtrer, durant le processus de fouille de données, les motifs exprimant une mesure de modèle inférieure à un seuil expert. Nous dégageons plusieurs propriétés théoriques afin de les exploiter pour parcourir plus efficacement l'espace de recherche. Bien que les exemples d'utilisation de la contrainte soient donnés pour la recherche d'itemsets, la méthode reste générique. Les performances et évaluations de pertinence ne seront données qu'au chapitre suivant, dans une étude de cas pour laquelle nous disposons à la fois de données et d'un modèle expert.

1. Recherche de chemins d'attributs dans un unique DAG attribué

Comme discuté dans l'état de l'art, les graphes deviennent omniprésents dans beaucoup de problèmes d'analyse de données. Récemment, des modèles de graphes plus riches ont été étudiés, où par exemple les sommets et arêtes sont attribués contrairement aux graphes simplement labellisés (un seul attribut). Par exemple, un réseau social peut être représenté comme un grand graphe où chaque sommet correspond à une personne, auquel sont associés ses domaines d'intérêt. Ces graphes ont été baptisés « graphes attribués » ou « graphes associés à des itemsets » (« *itemset-associated graphs* » de FUKUZAKI *et al.* (2010)). Quand une notion temporelle intervient, les arêtes sont orientées (on parle d'arcs) et le graphe devient acyclique du fait de l'aspect causal du temps.

Dans notre contexte spatio-temporel, les données sont représentées par un unique graphe orienté acyclique attribué (a-DAG) : les sommets sont des objets spatiaux caractérisés par un ensemble d'attributs et/ou événements, et les arcs dénotent la proximité spatio-temporelle entre ces objets (par exemple, le voisinage spatial entre deux occurrences de deux temps consécutifs). Le but étant de trouver les transitions (ou cheminements) de caractéristiques pouvant expliquer un phénomène particulier (lui-même caractérisé par un ensemble d'attributs), cela revient à chercher dans un a-DAG les chemins fréquents d'attributs.

1.1 Cadre théorique

Un **a-DAG** est un DAG $G = (V_G, E_G, \lambda_G)$ attribué par un ensemble d'items $\mathcal{I}$, et consiste en un ensemble de sommets V_G, un ensemble d'arcs $E_G \subseteq V_G \times V_G$ et une fonction d'attribution $\lambda_G : V_G \to \mathcal{P}(\mathcal{I})$ qui fait correspondre à chaque sommet du DAG G un sous-ensemble de $\mathcal{I}$. Par souci de concision et cohérence avec la littérature, les items sont simplement juxtaposés pour former un itemset. Ainsi, dans le a-DAG de la figure 3.1 page suivante, l'itemset $\{a, c\}$ associé au sommet ① est écrit ac.

Notons P une succession d'itemsets $I_i \in \mathcal{P}(\mathcal{I})$, c'est-à-dire $P = I_1 \rightarrowtail I_2 \rightarrowtail \cdots \rightarrowtail I_{|P|}$. On écrira par la suite P_i pour désigner le $i^{\text{ème}}$ itemset de P. P est un **chemin** si et seulement s'il existe une succession de sommets $v_1, v_2, \ldots, v_{|P|} \in V_G$ satisfaisant $\forall P_i \in P$, $P_i \subseteq \lambda_G(v_i)$ et telle que chaque v_i est un parent de v_{i+1} dans G. La succession de sommets $O = v_1 \rightarrowtail v_2 \rightarrowtail \cdots \rightarrowtail v_{|P|}$ est une **occurrence** du chemin P. Par exemple, sur le a-DAG donné en figure 3.1, les occurrences du chemin **de taille 3** $ah \rightarrowtail cd \rightarrowtail i$ sont ② ↣ ③ ↣ ⑥, ② ↣ ③ ↣ ⑧, ② ↣ ③ ↣ ⑩, ② ↣ ④ ↣ ⑦, ② ↣ ⑤ ↣ ⑦, et ⑤ ↣ ⑦ ↣ ⑧. La seule occurrence ② ↣ ③ ↣ ⑥ **supporte** quant à elle les chemins $ah \rightarrowtail bcd \rightarrowtail bi$, $a \rightarrowtail bcd \rightarrowtail bi$, $h \rightarrowtail bcd \rightarrowtail bi$, $h \rightarrowtail b \rightarrowtail bi$, et ainsi de suite. Nous noterons O_i pour désigner le $i^{\text{ème}}$ sommet de O. L'**ensemble des occurrences** de P dans G sera noté $occur_G(P)$.

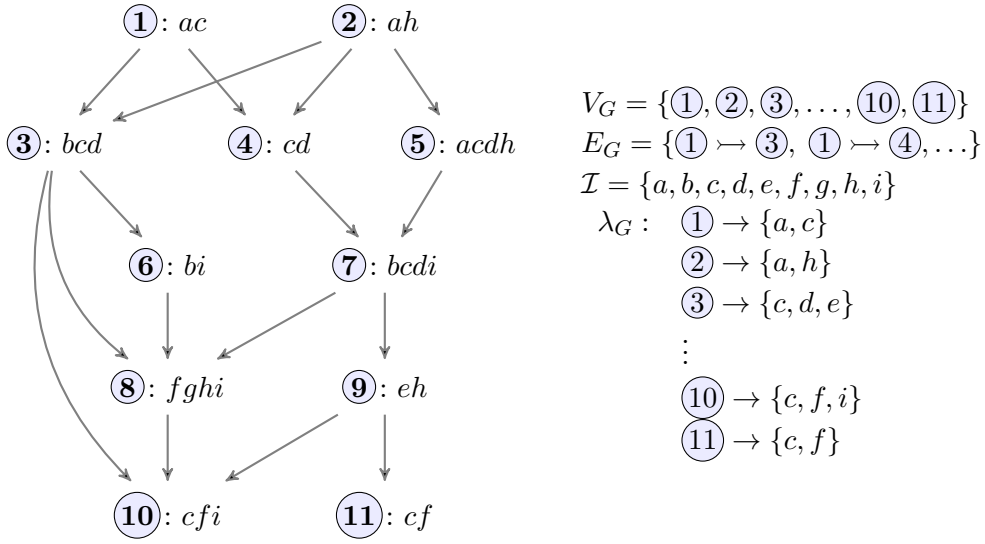


Figure 3.1 – Exemple de a-DAG.

Un tel chemin décrit correctement une séquence d'événements dans un a-DAG. Cependant, il serait utile de connaître la contribution des occurrences de chaque arc du chemin. En effet, un simple chemin peut apparaître de différentes façons, tel que montré par la figure 3.2. Dans le a-DAG A de cette figure, le chemin $a \rightarrowtail b \rightarrowtail c \rightarrowtail d$ apparaît à diverses occurrences, même s'il part toujours du même sommet 1. Dans le a-DAG B, ce même chemin apparaît de façon différente (avec moins d'embranchements). Nous proposons de différencier ces deux chemins en pondérant chaque arc pour qualifier leur participation aux occurrences. Nous définissions alors un nouveau domaine de motifs : les **chemins pondérés**.

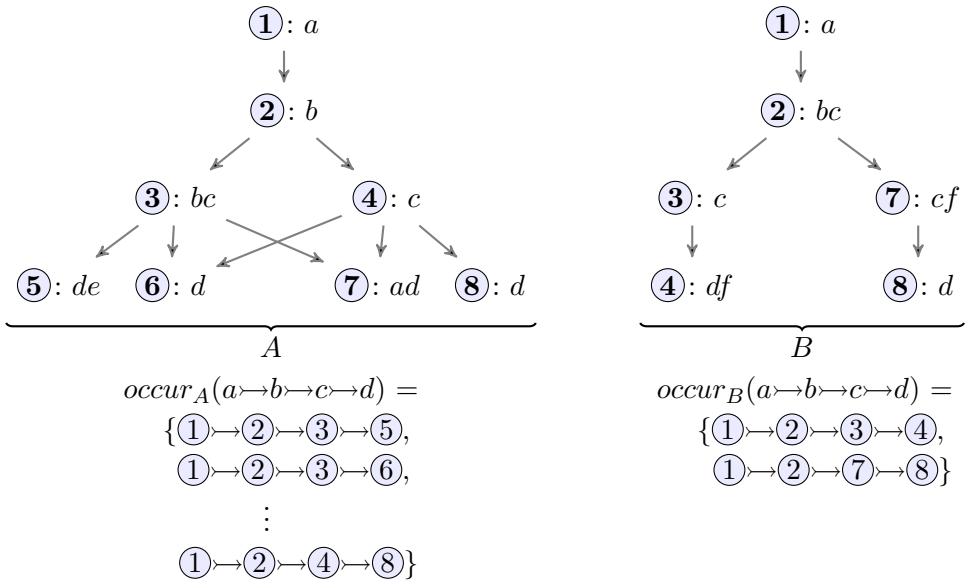


Figure 3.2 – Deux a-DAGs où le chemin $a \rightarrowtail b \rightarrowtail c \rightarrowtail d$ apparaît de différentes façons.

Chemin pondéré. Les chemins pondérés sont des chemins avec un poids sur chaque arc, représentant le nombre d'occurrences différentes de cet arc parmi toutes les occurrences du chemin. Dans les données de l'exemple de la figure 3.1, le chemin $P = ah \rightarrowtail cd \rightarrowtail i$, dont les occurrences ont été énumérées précédemment, nous donne le motif [1] :

$$ah \overset{4}{\rightarrowtail} cd \overset{6}{\rightarrowtail} i.$$

En effet, les différentes occurrences de $ah \rightarrowtail cd$ dans $occur_G(P)$ sont au nombre de 4, et les différentes occurrences de $cd \rightarrowtail i$ dans $occur_G(P)$ sont au nombre de 6. Une telle représentation permet de voir que l'itemset ah apparaît 4 fois avant l'apparition du chemin $cd \rightarrowtail i$, et que l'itemset i apparaît 6 fois après l'apparition du chemin $ah \rightarrowtail cd$. Dorénavant, $\omega_G(P_i \rightarrowtail P_{i+1})$ désignera le **poids** de l'arc entre les itemsets P_i et P_{i+1}. On notera $wp(G)$ [2] l'ensemble des chemins pondérés présents dans un a-DAG G.

Nous définissons aussi la notation $P_{i,j}$ avec $1 \leq i < j \leq |P|$ pour désigner un **fragment**, c'est-à-dire la portion d'un chemin pondéré P de P_i à P_j. Par exemple, si $P = ah \overset{4}{\rightarrowtail} cd \overset{6}{\rightarrowtail} i$, $P_{1,2} = ah \overset{4}{\rightarrowtail} cd$.

Relation d'inclusion. L'opérateur $\sqsubseteq$ sur un couple de chemins pondérés est défini de la manière suivante : $P \sqsubseteq P'$ si et seulement si $|P| \leq |P'|$ et $\exists k \in [0, |P'| - |P|]$ tel que :

$$\text{(inclusion d'itemsets) } 1. \quad \forall i \in [1, |P|],\ P_i \subseteq {P'}_{k+i} \tag{3.1}$$

$$\text{(préservation occurrences) } 2. \quad \forall j \in [1, |P|[,\ \omega_G(P_j \rightarrowtail P_{j+1}) = \omega_G({P'}_{k+j} \rightarrowtail {P'}_{k+j+1}) \tag{3.2}$$

Un chemin pondéré P' contient un autre chemin pondéré P si nous pouvons trouver P dans une sous-séquence de P' avec les mêmes poids (c'est-à-dire les mêmes occurrences). Nous disons ainsi que P' est un **super-chemin pondéré** de P.

Remarque 1.1

Un fragment Pi, j d'un chemin pondéré P est contenu par ce motif P (trivial). Dans ce cas particulier, les ensembles d'itemsets dont on teste l'inclusion (équation 3.1) sont en fait égaux.

Un des problèmes populaires en fouille de données est la recherche de motifs fréquents, où il s'agit de trouver des motifs dont le support/la fréquence est supérieur(e) à un seuil donné. À partir de la mesure proposée par BRINGMANN et NIJSSEN (2008) [3], nous définissons une mesure de support pour un motif P dans un a-DAG G, notée $\sigma_G(P)$:

$$\begin{aligned}\sigma_G(P) &= \min_{1 \leq i < |P|} \left| \{ (O_i) \rightarrowtail (O_{i+1}) \,/\, O \in occur_G(P) \} \right| \\ &= \min_{1 \leq i < |P|} \omega_G(P_i \rightarrowtail P_{i+1})\end{aligned}$$

1. Dans cette section, le mot « motif » référera systématiquement à un chemin pondéré. Un itemset sera désigné par son nom, afin d'éviter les confusions.
2. w pour *weighted*, p pour *path*
3. Cette mesure peut tout aussi bien s'appliquer sur des sommets ou des arcs

En d'autres termes, σ_G retourne le plus petit poids d'un chemin pondéré. Tout en étant (anti-)monotone et facilement calculable, ce support correspond bien à la notion de fréquence dans un unique DAG : il discrimine les chemins apparaissant à divers endroits dans un DAG de ceux commençant ou finissant par quelques arêtes. En d'autres termes, les chemins ayant des occurrences complètement distinctes seront mieux évalués que ceux qui partagent une ou plusieurs arêtes.

Les chemins pondérés aident à mieux décrire l'évolution d'un itemset vers un autre itemset (voir figure 3.2). De simples chemins auraient rendu floue, voire invisible, une telle distinction entre chemins ayant le même support mais dont les apparitions diffèrent.

La collection de motifs fréquents dans G est l'ensemble des motifs P tels que $\sigma_G(P) > minsup$, $minsup$ étant un seuil défini arbitrairement. Cependant, le nombre de chemins fréquents dans G peut être très grand. Dans une telle situation, il peut être judicieux de se pencher vers une représentation condensée des chemins fréquents (Calders *et al.*, 2004).

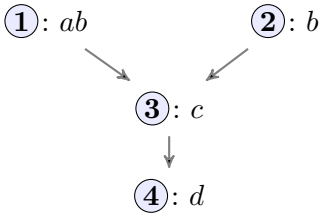
1.2 Contrainte de non-redondance

Dans un a-DAG G donné, nous cherchons une représentation condensée notée $cond(G)$ de tous les chemins pondérés : chaque chemin pondéré fréquent *ainsi que son support* doit être déductible de $cond(G)$. De plus, aucun élément de $cond(G)$ ne doit pouvoir être déductible à partir d'un autre élément de $cond(G)$. En d'autres termes, nous avons besoin à la fois de la maximalité, de l'unicité et de la complétude des solutions dans la collection $cond(G)$.

Plusieurs auteurs ont travaillé sur la fouille de motifs fermés fréquents (voir, par exemple, Pasquier *et al.*, 1999 ; Yan *et al.*, 2003 ; Yan et Han, 2003), formant une représentation condensée exacte des motifs fréquents. La collection des motifs fermés est beaucoup plus petite, mais garde la même information ; il reste donc possible d'en déduire tous les motifs ainsi que leur fréquence.

Fermeture et représentation condensée. Avant d'aller plus loin, il est nécessaire de bien distinguer les différences entre les représentations condensées faites à partir de transactions de graphes, et celles faites à partir d'un seul graphe. Dans la première configuration, la forme la plus populaire de représentation condensée est la collection des motifs fermés (ou motifs clos). Cette notion exploite la connexion de Galois qui existe entre transactions et motifs. Une propriété importante de l'opérateur de clôture est la préservation de la propriété du support : un motif et son fermé ont le même support. Cette propriété repose sur la définition du support, où un motif n'est compté qu'une fois par transaction.

Dans le cas d'une base de données constituée d'un seul grand graphe, la définition de support est différente. Elle s'appuie sur le nombre minimum d'arêtes qui supportent chaque fragment du chemin. De plus, la connexion de Galois définie pour un motif clos ne peut pas être appliquée puisqu'il n'y a aucune transaction en jeu. Un autre problème est qu'un motif n'est dans la classe d'équivalence que d'un seul clos, alors qu'un chemin pondéré peut être déduit à partir de plusieurs super-chemins pondérés, comme montré figure 3.3. Il s'avère que l'on peut donc très difficilement utiliser une approche basée sur la fermeture dans notre cas.



chemin pondéré	super-chemins pondérés
$c \overset{1}{\rightarrowtail} d$	$ab \overset{1}{\rightarrowtail} c \overset{1}{\rightarrowtail} d$ $b \overset{2}{\rightarrowtail} c \overset{1}{\rightarrowtail} d$

FIGURE 3.3 – Un chemin pondéré peut être inclus dans plusieurs super-chemins pondérés.

Une représentation condensée des chemins. Un motif étant caractérisé par une mesure (le plus souvent, le support), il est nécessaire que l'ensemble des solutions d'une extraction contienne l'information de cette mesure pour chacun des motifs. Comme le support d'un chemin pondéré est directement encodé dans le motif lui-même, nous pouvons définir l'ensemble des chemins condensés dans un a-DAG comme suit :

Définition 1.1 (*Ensemble des chemins condensés d'un a-DAG* $\boldsymbol{G}$)

$$cond(G) = \{P \in wp(G) \mid \nexists P' \in wp(G) \text{ tel que } P \sqsubseteq P'\}$$

Théorème 1.1 (*Représentation sans perte d'information*)

Chaque chemin pondéré de G ainsi que son support peut être déduit de $cond(G)$.

Démonstration :

i) Par définition, σ_G est le poids minimum d'un chemin pondéré. On déduit le support σ_G directement à partir du chemin pondéré.

ii) La relation d'inclusion utilise à la fois l'inclusion au niveau des attributs, mais aussi l'égalité des poids du chemin ; ainsi, un chemin pondéré peut être déduit à partir de ses super-chemins pondérés. ■

1.3 Une première stratégie d'énumération fondée sur la recherche d'itemsets clos dans une base de données n-aire

Nous allons dans un premier temps étudier les propriétés des chemins condensés de taille 2 (c'est-à-dire ne comportant qu'une arête), et faire le parallèle avec l'extraction d'itemsets clos dans une base de données ternaire. Nous allons pour cela notamment nous appuyer sur les travaux de CERF *et al.* (2008).

À partir de cette étude, nous proposons dans un deuxième temps un algorithme en deux étapes pour extraire les chemins pondérés condensés. Cette stratégie commence par une extraction de chemins pondérés de taille 2 par l'algorithme de CERF *et al.* (2008). Puis, nous étendrons ces chemins de façon à obtenir les chemins condensés du a-DAG.

À l'inverse de MABIT *et al.* (2011) qui cherchent d'abord à aplanir le graphe, notre approche évite la génération coûteuse et non-nécessaire de candidats. On peut ainsi gérer la grande complexité inhérente au problème (comme montré plus loin, section 1.5 page 67). De

plus, l'information structurelle est gardée, ce qui nous permet éventuellement d'insérer des contraintes structurelles dans le processus de fouille.

Tout d'abord, nous caractérisons les chemins pondérés de taille 2 (avec un seul arc) qui sont condensés par rapport à l'ensemble des chemins pondérés de taille 2 seulement. Ces chemins sont appelés « graines », car il suffit de les étendre simplement (par projections successives du a-DAG analysé) pour en trouver le(s) condensé(s) correspondant(s). Comme nous allons le montrer dans la suite de cette section, ces « graines » se rapprochent de la notion d'itemset clos dans une base de données ternaire. Ce parallèle permet d'utiliser un algorithme d'extraction d'itemsets clos dans une base de données n-aire pour rechercher ces chemins de taille 2. L'étape de recherche de ces graines est ensuite suivie par une recherche en profondeur dans le a-DAG pour obtenir l'ensemble des condensés. Cette méthode présente deux avantages : on se limite dans un premier temps à étudier un ensemble réduit de chemins (ceux de taille 2), puis on fait des projections successives par lecture directe dans le graphe en entrée, technique qui a déjà fait ses preuves lors de la recherche de séquences fréquentes (voir par exemple Pei *et al.*, 2001b).

1.3.1 Extensions et graines : définitions

L'efficacité de la stratégie choisie dépend, entre autres, de l'ensemble des graines que l'on prend au départ, qui dépend lui-même de la façon dont on génère les extensions ultérieures. La raison pour laquelle il est intéressant de scinder la stratégie en deux parties est qu'il est plus facile (donc plus rapide) de se focaliser sur un problème à la fois. La partie « extraction de graines » recherche des petits motifs. La partie « extension » est par définition celle qui générera les motifs condensés *in fine*. Afin que cette partie soit la plus rapide possible, il convient de la rendre la plus simple possible. D'où l'intérêt de faire, dans le a-DAG en entrée, des projections successives par rapport au chemin étendu : chaque étape d'extension se fait indépendamment des projections passées. Nous proposons donc de trouver l'ensemble des graines telles qu'il suffit de les étendre en longueur récursivement pour trouver les motifs condensés. Pour cela, nous introduisons tout d'abord la notion de concaténation.

Définition 1.2 (*Concaténation*)

La concaténation d'un itemset I et d'un chemin pondéré P résulte en un chemin pondéré P' tel que $P' = P \overset{\omega}{\rightarrowtail} I$ ou $P' = I \overset{\omega}{\rightarrowtail} P$, avec ω un poids arbitraire.

Propriété 1.1

Les poids et les attributs de P sont conservés après concaténation.

Définition 1.3 (*Extension simple*)

Un chemin pondéré P est **extensible simplement** dans G si et seulement si P peut être étendu récursivement par rapport à G par concaténation : chaque chemin pondéré P' obtenu après concaténation est tel que $P' \in wp(G)$.

On notera $ext(P)$ l'ensemble des chemins constituant les extensions simples d'un chemin pondéré P.

Nous insistons bien ici sur le fait qu'une concaténation n'est effectuée que dans la longueur du chemin : il ne s'agira en aucun cas d'ajouter un item à un itemset appartenant au chemin étendu : cet itemset de P serait alors modifié, ce qui est interdit par la définition.

Propriété 1.2

Un chemin pondéré $P' \in ext_G(P)$, obtenu après une extension simple d'un chemin pondéré P, contient P ($P \sqsubseteq P'$). P' est donc un super-chemin pondéré de P.

Il existe très souvent plusieurs extensions simples possibles d'un chemin pondéré. Par exemple, dans le a-DAG de la figure 3.1 page 52, le chemin pondéré $c \overset{3}{\rightarrowtail} bi$ (dont les occurrences sont ③↦⑥, ④↦⑦, et ⑤↦⑧) a comme extensions simples :

- $a \overset{5}{\rightarrowtail} c \overset{3}{\rightarrowtail} bi$
- $h \overset{3}{\rightarrowtail} c \overset{3}{\rightarrowtail} bi$
- $ah \overset{3}{\rightarrowtail} c \overset{3}{\rightarrowtail} bi$

mais aussi :

- $c \overset{3}{\rightarrowtail} bi \overset{2}{\rightarrowtail} f$
- $c \overset{3}{\rightarrowtail} bi \overset{3}{\rightarrowtail} h$
- $c \overset{3}{\rightarrowtail} bi \overset{2}{\rightarrowtail} fghi$

⋮

- $c \overset{3}{\rightarrowtail} bi \overset{2}{\rightarrowtail} fghi \overset{1}{\rightarrowtail} cfi$
- $c \overset{3}{\rightarrowtail} bi \overset{2}{\rightarrowtail} h$
- $c \overset{3}{\rightarrowtail} bi \overset{2}{\rightarrowtail} h \overset{3}{\rightarrowtail} cf$

ainsi que les extensions combinant des concaténations d'itemsets par le bas ou par le haut, telles que $ah \overset{3}{\rightarrowtail} c \overset{3}{\rightarrowtail} bi \overset{2}{\rightarrowtail} fghi \overset{1}{\rightarrowtail} cfi$ par exemple. Notons que le chemin pondéré $cd \overset{3}{\rightarrowtail} bi$ n'est pas une extension simple de $c \overset{3}{\rightarrowtail} bi$ (voir définition 1.2 et propriété 1.1).

Définition 1.4 (*Extension simple maximale*)

Les plus grandes des extensions simples d'un chemin pondéré P (au sens de la relation d'inclusion) seront appelées **extensions simples maximales**.
Formellement, parmi toutes les extensions simples $ext_G(P)$, les maximales sont les extensions P' telles que $\forall P' \in ext_G(P), \nexists P'' \in ext_G(P), P' \sqsubseteq P''$.

Dans l'exemple précédent, il existe plusieurs extensions simples maximales de $c \overset{3}{\rightarrowtail} bi$: par exemple, $ah \overset{3}{\rightarrowtail} c \overset{3}{\rightarrowtail} bi \overset{2}{\rightarrowtail} fghi \overset{1}{\rightarrowtail} cfi$ ou bien $a \overset{5}{\rightarrowtail} c \overset{3}{\rightarrowtail} bi \overset{3}{\rightarrowtail} h \overset{3}{\rightarrowtail} cf$ entre autres.

Maintenant que nous avons défini comment effectuer les extensions de manière simple, nous pouvons définir les graines que nous devons chercher.

Définition 1.5 (*Graine*)

Un chemin pondéré P est appelée **graine** ssi :

1. P est de taille 2 ;
2. les extensions simples maximales de P sont des éléments de $cond(G)$.

1.3.2 Extraction des graines et extraction d'itemsets fermés dans une relation n-aire

Il s'agit maintenant d'extraire l'ensemble des graines afin de les étendre pour trouver toutes les solutions de $cond(G)$. Le premier point de la définition d'une graine montre qu'il nous suffit de ne fouiller que l'ensemble des chemins pondérés de taille 2 du a-DAG donné en entrée, que nous appellerons $L2W$ [4]. Le deuxième point est plus difficile à vérifier : comment savoir si les extensions simples maximales d'un chemin de taille 2 seront bien des condensés avant même d'avoir effectué ces extensions (et évidemment sans savoir par avance quel est l'ensemble des solutions) ?

Pour répondre à cela, essayons de prendre le problème à l'envers en partant d'un chemin pondéré condensé, comme expliqué dans la remarque suivante :

Remarque 1.2

Soit un chemin pondéré condensé P. Par la définition d'un fragment (section 1.1 page 53), chacun de ses fragments $P_{i,j}$ est extensible simplement pour donner P. Celui-ci étant un condensé, il fait partie des extensions simples maximales du fragment $P_{i,j}$.

Cette remarque est formalisée par le lemme suivant :

Lemme 1.1

Tout fragment $P_{i,j}$ d'un motif condensé P suffit pour trouver P par des extensions simples successives.

Démonstration : Par définition d'un fragment (section 1.1 page 53), chacun des fragments $P_{i,j}$ d'un chemin pondéré P peut être concaténé successivement par des itemsets pour former P. $P_{i,j}$ est donc extensible simplement : on a $P \in ext_G(P_{i,j})$. Comme P est en plus un condensé, alors $\nexists P' \in wp(G), P \sqsubset P'$. Comme $ext_G(P_{i,j}) \subseteq wp(G)$, on a $\nexists P' \in ext_G(P_{i,j}), P \sqsubset P'$. De par la définition 1.4, P fait donc partie des extensions simples maximales du fragment $P_{i,j}$. ■

Lemme 1.2 (*Un fragment de taille 2 est une graine*)

Tout fragment $P_{i,i+1}$ d'un condensé P est une graine.

Démonstration : Le fragment $P_{i,i+1}$ d'un condensé P est par définition de taille 2. De par le lemme 1.1, chaque extension simple maximale de $P_{i,i+1}$ est un condensé. Par définition (définition 1.5), $P_{i,i+1}$ est une graine. ■

4. L pour *list*, 2 pour la taille, W pour *weighted*

Le problème est reformulé : comment identifier les fragments de taille 2 des condensés dans l'ensemble $L2W$?

Seule une fraction des chemins pondérés de taille 2 ($L2W$) sont des fragments de condensés. Comme observé dans la remarque 1.1 page 53, la différence entre ces fragments et de simples chemins est que les itemsets des fragments correspondent exactement aux itemsets du(des) motif(s) pondéré(s) condensé(s). En résumé, les fragments de taille 2 que nous cherchons sont maximaux au niveau des itemsets, comme expliqué par le théorème suivant.

Théorème 1.2

Soit F un fragment de condensé C de taille 2. On a donc $F = C_{i,i+1}$ avec $C \in cond(G)$, et F est une graine. Parmi tous les chemins pondérés de taille 2 que l'on peut former à partir de $occur_G(F)$, F est celui qui est maximal au niveau des itemsets :

$$\text{Si } F \text{ est une graine}, \quad \forall P \in L2W \text{ tel que } occur_G(P) = occur_G(F), \ P \sqsubset F.$$

Démonstration : Par l'absurde.
Supposons qu'il existe un chemin $P \in L2W$ tel que $occur_G(P) = occur_G(F)$ et $F \sqsubseteq P$.
Comme $occur_G(P) = occur_G(F)$, les extensions maximales de P seront issues des mêmes concaténations que celles des extensions maximales de F. Comme $F \sqsubseteq P$, nous aurons donc une extension maximale P' de P qui sera un super-chemin pondéré d'une extension maximale F' de F, c'est-à-dire $F' \sqsubseteq P'$. Or F est une graine : par définition, toute extension maximale de F est un condensé, c'est-à-dire que $\nexists P' \in wp(G)$ tel que $F' \sqsubseteq P'$. Cette contradiction rend l'hypothèse initiale est invalide. CQFD. ■

Pour des besoins de clarté, écrivons $Q(F)$ l'ensemble décrit dans le théorème 1.2 : $Q(F) = \{P \subset L2W \mid occur_G(P) = occur_G(F)\}$. On vient de voir que F est le plus grand chemin pondéré de cet ensemble. Le théorème suivant met en avant la relation entre $Q(F)$ et $L2W$ (l'ensemble des chemins pondérés de taille 2) :

Théorème 1.3

$$L2W = \bigcup_{F \text{ est une graine}} Q(F)$$

Démonstration :

1. $\supset$: Chaque F étant une graine, F est de taille 2. Donc chaque chemin pondéré de $Q(F)$ est aussi de taille 2. Ces chemins pondérés appartiennent donc à $L2W$.
2. $\subset$: Prenons un chemin pondéré fréquent $P \in L2W$. P est de taille deux. Notons son condensé P'. Il existe au moins un fragment F de P' de taille 2 (F est une graine) tel que $occur_G(P) = occur_G(F)$ puisque F et P ont le même condensé. Donc, par définition de l'ensemble $Q(F)$, on a $P \in Q(F)$. ■

Nous savons, grâce au théorème 1.2, que pour extraire toutes les graines, nous devons trouver tous les chemins pondérés qui sont maximaux par rapport aux mêmes ensembles d'occurrences. Il nous faudrait donc d'abord trouver ces ensembles d'occurrences. Grâce au

théorème 1.3, nous savons que trouver dans $L2W$ les chemins pondérés fréquents maximaux sur les itemsets revient à chercher (une partie) des graines.

Notons aussi qu'un chemin pondéré de taille 2 $P = P_1 \rightarrowtail P_2$ peut naturellement être représenté par un triplet $\{occur_G(P), P_1, P_2\}$. En exprimant de cette façon ces chemins, on voit que l'on peut travailler dans des bases de données ternaires (ou tri-dimensionnelles), telles que décrites par Cerf *et al.* (2008). Le théorème suivant décrit ce constat.

Théorème 1.4

Chercher dans $L2W$ les chemins pondérés qui sont maximaux sur les itemsets revient à chercher des itemsets clos dans la base de données ternaire suivante :

1. La première dimension est E_G, la seconde et la troisième sont $\mathcal{I}$;
2. Pour un arc $(v_1) \rightarrowtail (v_2) \in E_G$ donné, le tuple correspondant $T \subset (E_G, \mathcal{I}, \mathcal{I})$ est $T = ((v_1) \rightarrowtail (v_2), \lambda_G((v_1)), \lambda_G((v_2)))$.

On appellera $LC2W$ l'ensemble de ces motifs clos.

Démonstration : Reprenons la définition d'un fermé de dimension 3 et les notations de Cerf *et al.* (2008) : un motif de dimension 3 $S = \{S^1, S^2, S^3\}$ est fermé si et seulement s'il n'existe aucun autre motif S' de dimension 3 tel que $\forall i \in \{1, 2, 3\}, S^i \subseteq S'^i$. Comme les dimensions 2 et 3 représentent respectivement les itemsets des sommets d'origine et de destination, la définition des motifs fermés sur $(E_G, \mathcal{I}, \mathcal{I})$ est la définition des chemins pondérés condensés de taille 2 (mais condensés seulement par rapport aux chemins pondérés *de taille 2*). Leur poids est le nombre d'arcs différents, qui est en fait la taille de S^1. ■

Par exemple, à partir du motif de dimension 3 $S = \{\{③ \rightarrowtail ⑥, ④ \rightarrowtail ⑦, ⑤ \rightarrowtail ⑦\}, \{c, d\}, \{b, i\}\}$, on obtient le chemin pondéré $P = cd \overset{3}{\rightarrowtail} bi$. Avoir un tel ensemble de chemins pondérés de taille 2 nous permet de procéder à une extension via une recherche en profondeur dans le a-DAG.

Nous possédons maintenant un ensemble de graines nommé $LC2W$.

1.3.3 Extension des graines vers des chemins pondérés condensés

Le but de la deuxième étape est d'étendre les graines pour générer les chemins pondérés condensés fréquents finaux. Pour ce faire, nous exploitons les chemins pondérés précédemment extraits. Ces chemins pondérés sont assurés d'être maximaux au niveau des itemsets des sommets d'origine et de destination. Comme les fragments trouvés précédemment sont des portions de condensés qui interviennent soit au début, au milieu ou à la fin de chemins pondérés, l'extension doit être tentée à la fois vers le bas (en ajoutant des fils) et le haut (en ajoutant des pères). Un sommet dans un a-DAG peut avoir plusieurs fils et pères. Pour gérer toutes les combinaisons possible d'extension, nous stockons les solutions dans un graphe (lui-même orienté acyclique), tel que montré dans la figure 3.4. La construction de cette figure sera expliquée en détails un peu plus loin. Les deux extensions étant fondamentalement les mêmes, nous ne présentons que l'extension vers le bas. Celle vers le haut est faite en suivant les mêmes principes (en considérant simplement les sommets pères au lieu des sommets fils).

Le graphe solution contient des sommets identifiés par un couple d'itemset et de ses occurrences dans le a-DAG G donné en entrée.

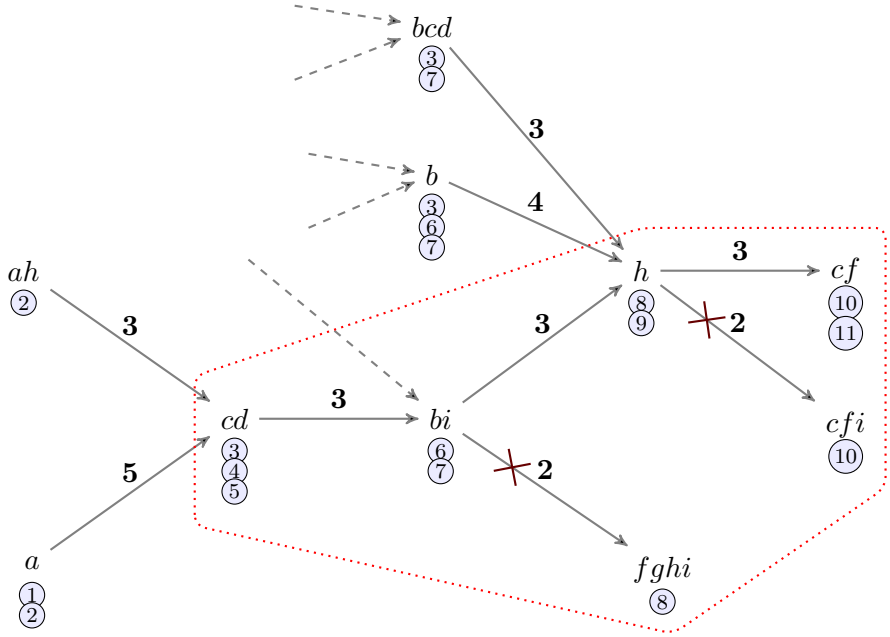


FIGURE 3.4 – Graphe solution partiel, construit à partir de la graine $cd \overset{3}{\rightarrowtail} bi$ (figure 3.1, $minsup = 3$).

Pour un motif P, nous définissons maintenant :

— $V_{dest} = \{(v_{|P|}) \in occur_G(P)\}$, c'est-à-dire les derniers sommets des occurrences de P. L'extension se fera à partir de ces sommets ;

— $Li = \{i \in \mathcal{I} \mid \forall (v) \in V_{dest}, v \text{ a au moins un fils } (u) \mid i \in \lambda_G((u))\}$. Li est la liste des items qui sont dans au moins un fils de chaque dernier sommet de P.

La méthode d'extension est fondée sur la proposition suivante, dérivée du théorème 1.2 :

Théorème 1.5

Soit P un chemin pondéré à étendre (vers le bas). Soit $DB|_P$, la base de données binaire projetée par rapport à P, définie comme suit :

1. La première dimension est E_G, la seconde est Li,
2. Pour un arc $(v_1) \rightarrowtail (v_2) \in E_G \mid (v_1) \in V_{dest}$, la transaction correspondante $T \subset (E_G, Li)$ est $T = ((v_1) \rightarrowtail (v_2), \lambda_G((v_2)) \cap Li)$

Les chemins pondérés étendus $\{P \rightarrowtail I \mid I \text{ est fermé dans } DB|_P\}$ sont des super-chemins pondérés fréquents de P. Ainsi, il suffit d'étendre récursivement les super-chemins pondérés pour en obtenir le(s) condensé(s).

Démonstration :

i) Premièrement, l'extension proposée garantit l'inclusion des itemsets et la préservation des poids. En effet, si un chemin pondéré P doit être simplement étendu par concaténation

(donc sans que son préfixe soit modifié), alors chaque dernier sommet de P doit être extensible. Si une partie seulement de ses occurrences peut être étendue, les poids du préfixe ne sont plus conservés avec extension de cette partie (l'inverse est vrai).

ii) Suite à l'extension, nous obtenons donc un super-chemin pondéré de P. Si celui-ci ne peut plus être étendu, alors il est condensé. En effet, il ne peut pas exister de sur-motifs ayant les mêmes poids (de par la définition des fermés dans $DB|_P$, chaque itemset étant un fermé fréquent obtenu par projections successives de la base). Les derniers super-chemins pondérés générés ainsi récursivement sont donc des condensés fréquents. ■

Corollaire 1.1

Si $Li = \emptyset$, alors le chemin pondéré n'est plus extensible.

Remarque 1.3

Si $\exists$ⓥ $\in V_{dest}$ | ⓥ n'a pas de fils, alors $Li = \emptyset$ et le chemin pondéré n'est plus extensible.

Algorithme Chaque chemin pondéré de l'ensemble des graines $LC2W$ est récursivement étendu jusqu'à ce qu'il devienne condensé (algorithme 1 lignes 2 à 7)[5]. Pour étendre un chemin P, nous regardons V_{dest}, les derniers sommets de $occur_G(P)$. Si un chemin P peut être étendu avec un itemset I, alors chaque dernier sommet de $occur_G(P)$ doit avoir au moins un fils dont l'itemset associé inclut I (remarque 1.3). Cet ensemble d'itemsets obtenu à partir de $occur_G(P)$ constitue la base de données projetée de P ($DB|_P$, algorithme 2 ligne 4).

A partir de cette projection, nous étendons le chemin avec les itemsets satisfaisant la définition de la représentation condensée. Pour ce faire, nous extrayons les itemsets fermés dans la base de données projetée $DB|_P$ (algorithme 2, ligne 5), tel qu'expliqué dans le théorème 1.5. L'extension est alors effectuée (algorithme 2, lignes 6 à 11) pour chaque itemset généré vérifiant la contrainte de support. Si la nouvelle extension est une graine (algorithme 2 ligne 7), celle-ci devra être traitée comme telle : une extension dans le sens opposé est alors effectuée, et la graine n'aura pas besoin d'être étendue ultérieurement (algorithme 2 lignes 8 et 9).

Algorithme 1 : Rechercher chemin pondérés condensés fréquents.

Entrées : $LC2W$: Ensemble de graines, G : a-DAG

1 Créer un graphe solution G_{sol}
2 **tant que** $LC2W \neq \emptyset$ **faire**
3 Piocher une graine F dans $LC2W$
4 Ajouter F à G_{sol}
5 `EtendreChemin` (G,$LC2W$,F,G_{sol}) `// extension vers le bas`
6 `EtendreChemin` (G^{-1},$LC2W^{-1}$,F^{-1},G_{sol}) `// extension vers le haut`
7 Générer solutions à partir de G_{sol}

5. Par souci de lecture, l'exposant $\cdot^{-1}$ appliqué à un ensemble d'arcs signifie que nous avons inversé leur direction.

Algorithme 2 : `EtendreChemin`

Entrées : $LC2W$: Ensemble de graines, G : a-DAG, P : chemin pondéré « candidat » à extension, G_{sol} : graphe solution à remplir

Sorties : G_{sol}

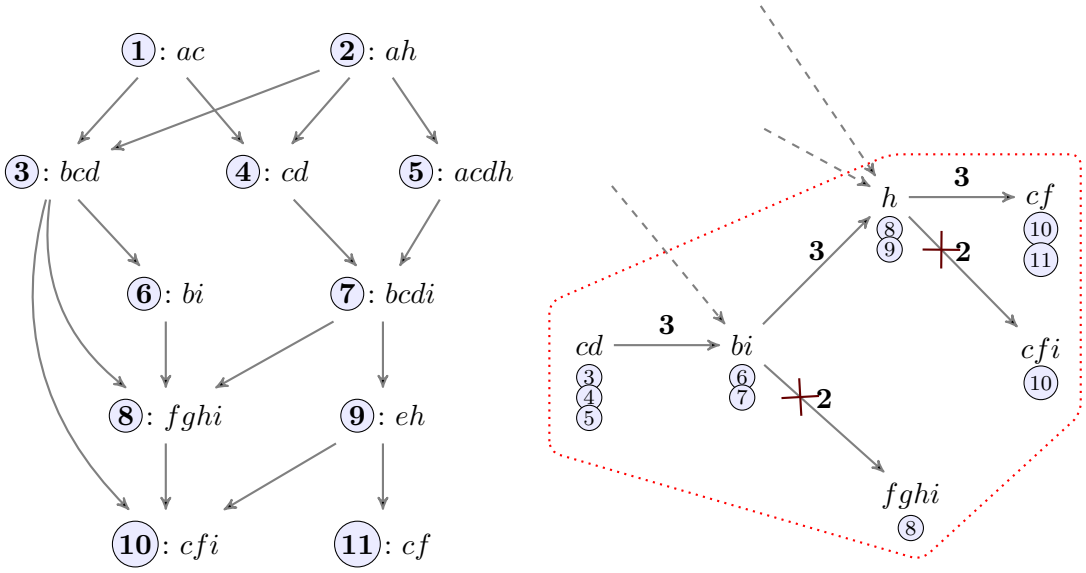
1 $V_{dest} := \{\text{derniers sommets de } occur_G(P)\}$
2 **si** $\exists$ⓥ $\in V_{dest}$ tel que ⓥ n'a pas d'enfant (ⓥ est une feuille) **alors stop**
3 $Li := \{i \in \mathcal{I} \mid \forall$ⓥ $\in V_{dest}$, ⓥ a au moins un fils ⓤ $\mid i \in \lambda_G($ⓤ$)\}$
`// Li est la liste des items qui sont dans au moins un fils de chaque dernier sommet de P`
4 $DB|_P := \{\forall$ⓥ$\rightarrowtail$ⓤ$\in E_G \mid$ ⓥ $\in V_{dest}$, transaction $T = \{$ⓥ$\rightarrowtail$ⓤ, $i_1 i_2 \dots i_N\} \mid i_k \in Li \cap \lambda_G(u)\}$
5 **pour chaque** itemset I fermé fréquent de $DB|_P$ de fréquence ω **faire**
6 $P' := P_{|P|} \overset{\omega}{\rightarrowtail} I$
7 **si** $P' \in LC2W$ **alors**
`// P' est une graine. Il ne faut pas oublier l'extension dans le sens inverse.`
8 Enlever P' de $LC2W$
9 `EtendreChemin` (G^{-1}, $LC2W^{-1}$, P'^{-1}, G_{sol})
`// On ajoute l'extension au graphe solution` G_{sol}
10 Ajouter P' à G_{sol}
11 `EtendreChemin` (G, $LC2W$, $P_1 \rightarrowtail \dots \rightarrowtail P_{|P|} \overset{\omega}{\rightarrowtail} I$, G_{sol})

Exemple Considérons le premier a-DAG de la figure 3.1 sur lequel nous appliquons l'algorithme avec $minsup = 3$. Nous prenons la graine $cd \overset{3}{\rightarrowtail} bi$, dont les occurrences sont ③$\rightarrowtail$⑥, ④$\rightarrowtail$⑦ et ⑤$\rightarrowtail$⑦, et nous essayons de l'étendre vers le bas (ses derniers sommets sont donc ⑥ et ⑦), comme montré dans la partie droite de la figure 3.5 page suivante. Les items candidats pour l'extension (ceux appartenant à au moins un fils de chaque dernier sommet) sont f, g, h et i mais pas e puisque le sommet ⑥ n'a aucun fils dont l'itemset associé contient e. La base de données projetée $DB|_P$ est la suivante :

⑥$\rightarrowtail$⑧	f, g, h, i
⑦$\rightarrowtail$⑧	f, g, h, i
⑦$\rightarrowtail$⑨	h

La ligne 5 de l'algorithme 2 nous donne l'itemset fermé h supporté par les arcs ⑥$\rightarrowtail$⑧, ⑦$\rightarrowtail$⑧ et ⑦$\rightarrowtail$⑨ (mais pas l'itemset $fghi$ car son support est 2, c'est-à-dire inférieur au $minsup$).

Nous pouvons maintenant ajouter l'itemset h au chemin comme le montrent les figures 3.4 et 3.5b, sur lesquelles les sommets sont représentés en dessous de chaque itemset qu'ils



(a) Retranscription du a-DAG de la figure 3.1 page 52

(b) Extensions vers le bas à partir de la graine $cd \overset{3}{\rightarrowtail} bi$

Figure 3.5 – Déroulement de l'algorithme de recherche des chemins pondérés condensés fréquents

supportent. Les branches coupées indiquent que l'extension viole la contrainte de support. Comme l'extension $bi \overset{3}{\rightarrowtail} h$ couvre toutes les occurrences de $bi \rightarrowtail h$ dans le a-DAG, celle-ci est une graine. De même, l'extension suivante $h \overset{3}{\rightarrowtail} cf$ est aussi une graine. Nous la traitons donc comme telle (algorithme 2 lignes 8 et 9). Sur la figure 3.4, on voit ainsi les différentes extensions vers le haut de cette graine.

Une fois l'extension de la graine $cd \overset{3}{\rightarrowtail} bi$ vers le bas achevée, on effectue celle vers le haut (figure 3.4). Finalement, nous générons les deux chemins pondérés condensés contenant $cd \overset{3}{\rightarrowtail} bi$, à savoir $ah \overset{3}{\rightarrowtail} cd \overset{3}{\rightarrowtail} bi \overset{3}{\rightarrowtail} h \overset{3}{\rightarrowtail} cf$ et $a \overset{5}{\rightarrowtail} cd \overset{3}{\rightarrowtail} bi \overset{3}{\rightarrowtail} h \overset{3}{\rightarrowtail} cf$.

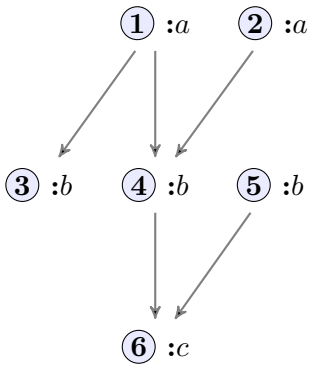
1.4 Une seconde stratégie fondée sur l'extension directe de l'ensemble complet des graines

Nous montrons ici que la stratégie précédente est correcte mais incomplète. Certaines graines, et donc certains chemins pondérés, ne sont pas générés. Nous proposons donc un autre algorithme regroupant les étapes d'extraction des graines et leurs extensions afin d'extraire toutes les solutions.

1.4.1 Extraction de l'ensemble complet des graines

Jusqu'ici, les différents théorèmes et lemmes démontrés prouvent la correction de la méthode de fouille. En revanche, un sous-ensemble de solutions n'est pas trouvé par l'algorithme proposé précédemment. En effet, rechercher les motifs clos dans $L2W$ ne revient pas à chercher toutes les graines $seed(G)$, comme le montre l'exemple de la figure 3.6. Dans ce a-DAG,

les maximaux de $L2W$ sont $a \overset{3}{\rightarrowtail} b$ et $b \overset{2}{\rightarrowtail} c$. Comme aucune de ces deux graines n'est extensible (corollaire 1.1), le condensé $a \overset{2}{\rightarrowtail} b \overset{1}{\rightarrowtail} c$ n'est pas trouvé.



Condensés représentant le a-DAG :

- $a \overset{3}{\rightarrowtail} b$
- $b \overset{2}{\rightarrowtail} c$
- $a \overset{2}{\rightarrowtail} b \overset{1}{\rightarrowtail} c$

Graines des condensés :

- $a \overset{3}{\rightarrowtail} b$ supporté par ①↣③, ①↣④, et ②↣④
- $b \overset{2}{\rightarrowtail} c$ supporté par ④↣⑥ et ⑤↣⑥
- $a \overset{2}{\rightarrowtail} b$ supporté par ①↣④ et ②↣④
- $b \overset{1}{\rightarrowtail} c$ supporté par ④↣⑥

FIGURE 3.6 – Exemple de a-DAG où les graines $LC2W$ ne suffisent pas à générer le condensé $a \overset{2}{\rightarrowtail} b \overset{1}{\rightarrowtail} c$.

En effet, certains fragments de condensés ne sont pas maximaux *par rapport à* $L2W$; ils ne le sont que par rapport à un sous-ensemble d'occurrences. En fait, cette maximalité n'est assurée que pour un des ensembles Q définis dans le théorème 1.2. En effet, le théorème 1.3 nous assure que tout fragment maximal dans $L2W$ l'est aussi dans un certain ensemble Q. L'inverse n'est malheureusement pas vrai, comme le montre le contre-exemple précédent. Pour un ensemble $Q(F)$ (où F est une graine), il existe des sous-ensembles $Q^i \subset Q(F)$ eux-même inclus dans des ensembles $Q^j \subset Q(F)$. Ce sont les fragments maximaux de Q^i contenus par ceux de Q^j que l'algorithme proposé ne peut pas trouver. Pour reprendre l'exemple précédent, les graines du condensé $a \overset{2}{\rightarrowtail} b \overset{1}{\rightarrowtail} c$ sont $a \overset{2}{\rightarrowtail} b$ et $b \overset{1}{\rightarrowtail} c$, mais leurs occurrences sont clairement contenues dans les occurrences d'autres graines.

Afin de trouver les graines manquantes, il faut vérifier s'il existe, pour les occurrences d'une graine de $LC2W$ donnée, un plus petit ensemble d'occurrences tel que celui-ci est extensible. Toujours dans l'exemple de la figure 3.6, la graine $a \overset{3}{\rightarrowtail} b$ est –entre autres– supportée par les occurrences ①↣④ et ②↣④, lesquelles forment un chemin pondéré similaire $a \overset{2}{\rightarrowtail} b$. Les occurrences de ce chemin sont inclues dans celles de $a \overset{3}{\rightarrowtail} b$, mais à la différence de ce dernier, $a \overset{2}{\rightarrowtail} b$ est extensible. Précisons que ce plus petit ensemble d'occurrences doit rester maximal par rapport au condensé dont il est la graine. Ainsi, on ne prendra pas le chemin pondéré formé à partir de la seule occurrence ①↣④, ni celui formé à partir de ②↣④, car leurs ensembles V_{dest} respectifs sont les mêmes que celui de $a \overset{2}{\rightarrowtail} b$ ($V_{dest} = \{④\}$), et mènent donc à la même extension.

Remarquons que cette nouvelle graine peut être trouvée en faisant des projections successives de G, de la même façon que lors de l'étape d'extension. La seule différence est que l'ensemble des items à exprimer dans cette base de données projetée ne se limite pas à Li. Au contraire, on recherche justement les ensembles d'itemsets fermés qui ne peuvent pas étendre la totalité des sommets de V_{dest} d'une graine dans $LC2W$. Ainsi, pour chaque graine F de

$LC2W$, on appellera la fonction récursive `TrouverEnsembleCompletDesGraines` (algorithme 3). Notons que, pour déterminer si une projection forme bien une nouvelle graine, nous devons d'abord vérifier si une graine plus grande n'existe pas déjà (condition de maximalité sur un ensemble Q) en supportant le même ensemble d'occurrences (ligne 5).

Algorithme 3 : `TrouverEnsembleCompletDesGraines`.

Entrées : G : a-DAG, F : graine appartenant à $LC2W$
Sorties : $LC2W$

1 $V_{dest} := \{$derniers sommets de $occur_G(F)\}$
2 $DB|_F := \{\forall ⓥ \rightarrowtail ⓤ \in E_G \mid ⓥ \in V_{dest},$
transaction $T = \{ⓥ \rightarrowtail ⓤ, \lambda_G(ⓤ)\}$
3 **pour chaque** itemset I fermé fréquent de $DB|_P$ de fréquence ω **faire**
4 $F' := F_{|F|} \overset{\omega}{\rightarrowtail} I$;
5 **si** $\nexists F'' \in LC2W \mid F' \sqsubseteq F'' \wedge occur_G(F') = occur_G(F'')$ **alors**
// F' est une nouvelle graine
6 Ajouter F' à $LC2W$;
7 `TrouverEnsembleCompletDesGraines` (G,F');

1.4.2 Extension à partir des graines

Remarquons qu'afin de trouver l'ensemble complet des graines, l'algorithme 3 effectue plusieurs projections. Ainsi, cet algorithme effectue déjà une opération nécessaire à l'extension des graines. Nous pouvons donc commencer à construire le graphe solution G_{sol} avant même d'avoir trouvé toutes les graines. L'algorithme 4 détaille la méthode à suivre. Les instructions déjà utilisées et expliquées dans l'algorithme 3 sont grisées.

Algorithme Pour construire le graphe solution, nous pouvons d'ores et déjà ajouter chaque graine à G_{sol} dès que celle-ci est trouvée (ligne 6), comme dans l'étape d'extension de l'algorithme 2 `EtendreChemin`. Tout comme dans cette étape d'extension, l'ajout d'une nouvelle graine F' se fera par concaténation dans G_{sol} si et seulement si tous les sommets de V_{dest} ont pu être étendus (opération déjà expliquée et illustrée par les figures 3.4 et 3.5). Dans le cas contraire, l'origine de cette graine sera nouvellement créée dans G_{sol} avec les sommets correspondants (qui seront donc un sous-ensemble de V_{dest}). De plus, on veillera à bien étendre dans le sens inverse afin de retrouver le bon sous-ensemble d'occurrences de chemins pondérés (ligne 9).

La figure 3.7 donne le résultat de l'exécution de cet algorithme sur la a-DAG de la figure 3.6. À partir de la graine $a \overset{3}{\rightarrowtail} b$, on effectue une extension vers le bas en cherchant les itemsets clos dans la base de données projetée. On trouve l'unique itemset b. Comme cet itemset ne permet d'étendre la graine $a \overset{3}{\rightarrowtail} b$ qu'à partir du sommet ④ seulement (et non tous les sommets ③ et ④ de V_{dest}), cette extension est stockée dans G_{sol} avec un nouveau sommet (b, {④}), et est représentée par la flèche rouge en pointillés. Cette nouvelle graine

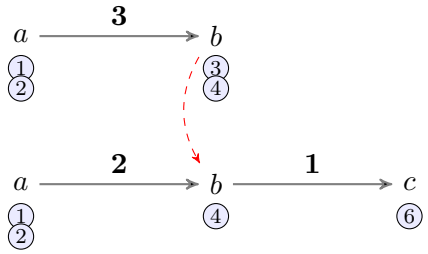


FIGURE 3.7 – Extensions à partir de la graine $a \overset{3}{\rightarrowtail} b \in LC2W$ du a-DAG de la figure 3.6, d'après l'algorithme 4.

$b \overset{1}{\rightarrowtail} c$ d'occurrence ④→⑥ est ensuite étendue vers le haut.

Algorithme 4 : `TrouverEnsembleCompletDesGrainesEtEtendre`.

Entrées : G : a-DAG, F : graine appartenant à $LC2W$
Sorties : $LC2W$, G_{sol}

$V_{dest} := \{$derniers sommets de $occur_G(F)\}$
$DB|_F := \{\forall$ⓥ→ⓤ$\in E_G \mid$ ⓥ $\in V_{dest}$,
transaction $T = \{$ⓥ→ⓤ$, \lambda_G($ⓤ$)\}\}$
pour chaque itemset I fermé fréquent de $DB|_P$ de fréquence ω **faire**
  $F' := F_{|F|} \overset{\omega}{\rightarrowtail} I$;
  si $\nexists F'' \in LC2W \mid F' \sqsubseteq F'' \wedge occur_G(F') = occur_G(F'')$ **alors**
    `// F' est une nouvelle graine`
    Ajouter F' à $LC2W$ et à G_{sol};
    `TrouverEnsembleCompletDesGrainesEtEtendre` (G,F');
  si au moins un des sommets de V_{dest} n'a pas été étendu **alors**
    `TrouverEnsembleCompletDesGrainesEtEtendre` (G^{-1},F'^{-1});

1.5 Performances

1.5.1 Jeux de données artificiels

Afin d'en tester les performances, nous avons fait subir des batteries de tests à ce dernier algorithme. Dans un premier temps, nous avons créé artificiellement trois jeux de données. Leur nom indique leurs caractéristiques : le jeu de données « V20K, E60K » contient ainsi 20 000 sommets et 60 000 arêtes. De même, nous avons créé les deux autres jeux de données « V40K, E120K » et « V200K, E600K » afin d'observer l'impact de la taille des DAGs sur les performances. Nous avons généré des attributs pour leur sommets. Parmi un ensemble de 15 attributs, nous avons pour chaque sommet tiré au sort [6] le nombre des attributs, puis les attributs en eux-mêmes [7]. Chacun des trois jeux de données comporte ainsi deux versions :

6. répartition gaussienne
7. répartition uniforme

une version dont la taille de l'ensemble des attributs varie entre 1 et 5 items ($1 \leq |\lambda_G| \leq 5$), et une version où cette taille varie entre 5 et 10 items ($5 \leq |\lambda_G| \leq 10$). Les temps d'exécution et le nombre de chemins pondérés fréquents sont reportés sur la figure 3.8. Nous avons effectué les tests sur une machine disposant de 16 giga octets de mémoire vive, et un processeur Intel® Core™ i5-2400 cadencé à 3,10 GHz. Les points manquants sur le graphe indiquent que le test pour le support minimum correspondant ne disposait pas assez de mémoire.

Dans un deuxième temps, nous avons voulu nous assurer que le jeu de données artificiel ressemble plus aux jeux de données tirés d'application spatio-temporelle. En l'occurrence, ceux-ci comportent une nette répartition par niveau sur les a-DAG ; chaque niveau correspond à une tranche temporelle, et il n'existe d'arc qu'entre deux tranches consécutives. Nous avons donc réparti en 10 niveaux les a-DAG générés précédemment. Les résultats sont reportés sur la figure 3.9.

1.5.2 Graphes de citation

Nous avons aussi examiné un jeu de données réel, dans lequel les sommets représentent des brevets déposés aux États-Unis entre 1975 et 1999 [8], et les arêtes représentent une citation d'un brevet vers un autre brevet. Chaque sommet comporte entre 5 et 7 attributs : le pays de la personne ou de l'entreprise déposant le brevet, l'état si ce pays est les États-Unis, l'année de dépôt, la classe du brevet, sa catégorie, et sa sous-catégorie. Il y a au total 506 items différents. Nous avons décomposé le jeu de données original en trois autres jeux plus petits. Le premier comporte 414 487 citations, le deuxième 196 097, et le troisième 75 687. Chaque test a été effectué 3 fois, le temps affiché sur la figure 3.10 est le temps moyen.

1.5.3 Discussions

Sans surprise, plus la taille du a-DAG augmente (en nombre de sommets tout comme en nombre d'arêtes), plus l'exécution est longue et les solutions sont nombreuses. On peut remarquer toutefois que la taille de l'ensemble des itemsets semble influencer plus fortement le processus. Seules les recherches sur les a-DAG comportant moins de 5 items passent à l'échelle sans contrainte de support. La répartition par niveaux des graphes rend plus difficile l'extraction des motifs fréquents. En effet, un graphe réparti par niveaux est assuré de comporter plus de longs chemins qu'un graphe sans répartition. Enfin, on peut aussi remarquer le biais induit dans la génération des graphes artificiels : à seuils de fréquence relative égaux, on trouve le même nombre de solutions, quelle que soit la taille du graphe. Ce constat soulève le problème de la génération d'un jeu de données synthétique « réaliste ».

Nous effectuerons une évaluation qualitative sur un jeu de données portant sur l'érosion en Nouvelle-Calédonie ; cette étude de cas sera détaillée dans le chapitre 4.

8. Jeu de données « cit-Patents », Stanford Large Network Dataset Collection

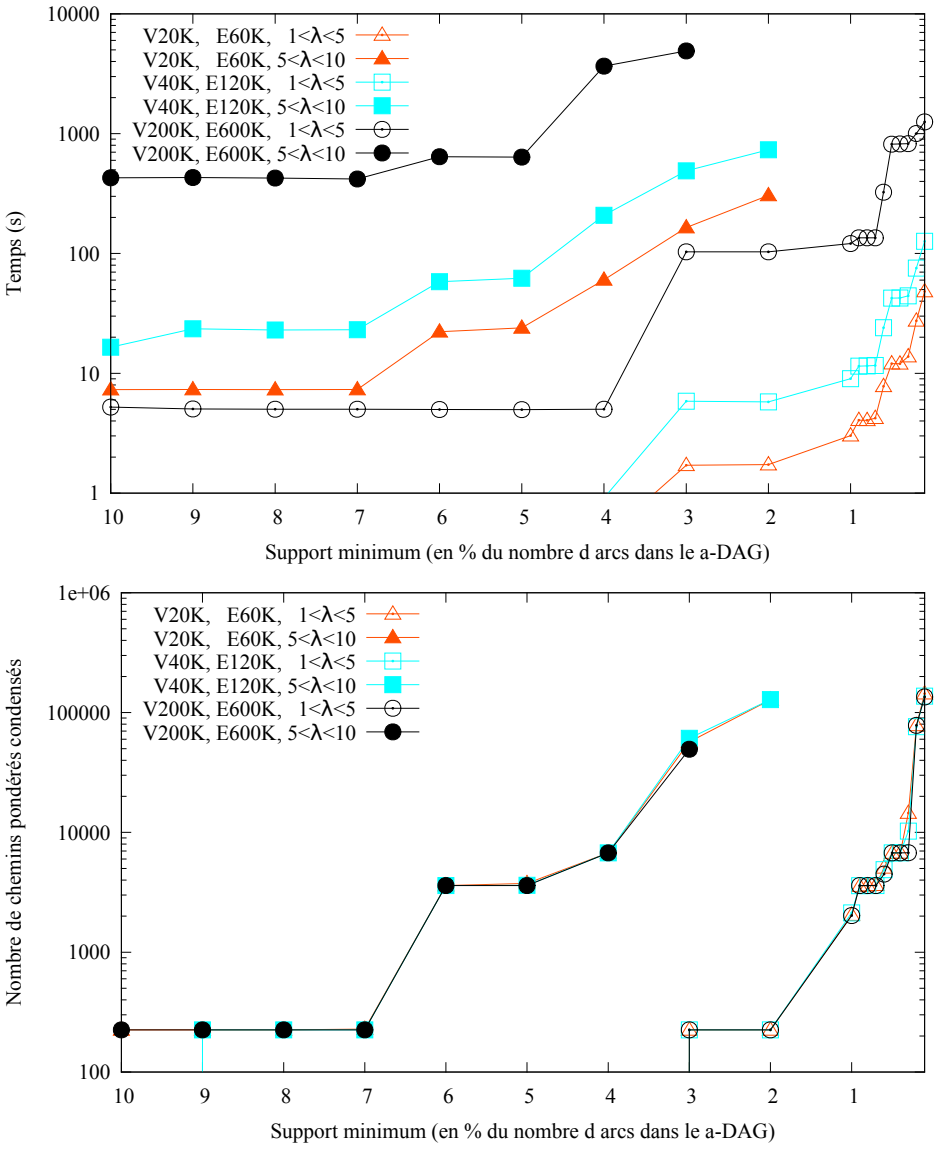


FIGURE 3.8 – Temps d'exécution et nombre de solutions trouvées, pour six jeux de données, en fonction de différentes valeurs de fréquence minimum (les seuils de fréquence sont donnés en pourcentage du nombre total d'arcs dans le a-DAG fouillé).

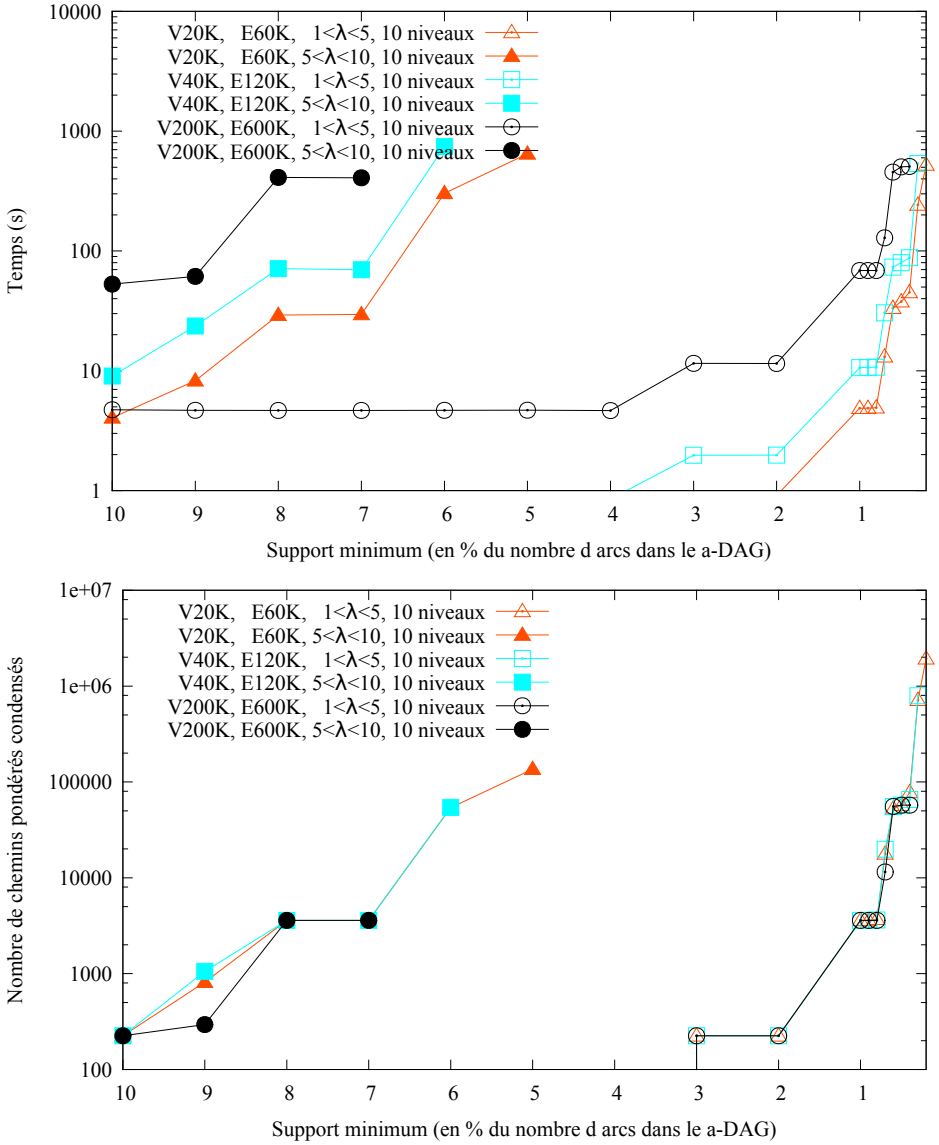


Figure 3.9 – Performances pour les jeux de données artificiels répartis en 10 niveaux.

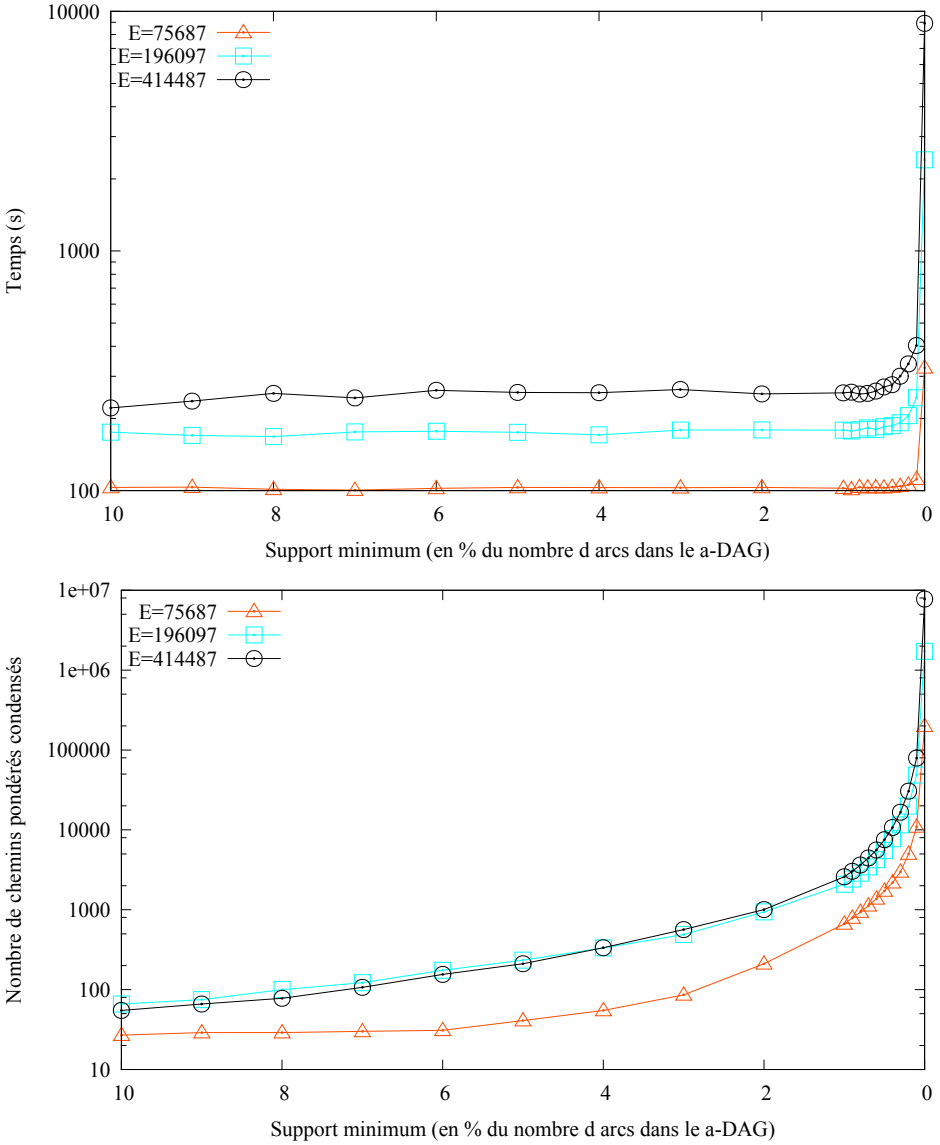


FIGURE 3.10 – Performances pour la fouille d'un graphe de citation de brevets.

2. Utilisation de modèles mathématiques pendant la fouille

Pour aider à la découverte de connaissances à partir de données, de nombreuses techniques de fouille de motifs ont été proposées. Un des verrous ralentissant leur diffusion est le nombre de motifs calculés qui s'avèrent soit triviaux, soit inintéressants par rapport à une connaissance existante. L'intégration de la connaissance du domaine dans la fouille de données sous contraintes est limitée dans les travaux existants. Certains motifs pertinents sont manquants car les méthodes échouent en partie dans l'évaluation de leur intérêt subjectif. Pourtant, dans la littérature nous disposons souvent de modèles mathématiques définis pas les experts et basés sur leur connaissance du domaine. Nous proposons donc d'exploiter de tels modèles pour en dériver des contraintes pouvant être utilisées durant la phase de fouille de données, afin d'améliorer à la fois la pertinence des motifs et l'efficacité des calculs. L'approche se veut générique, même si elle sera ici appliquée sur la découverte d'itemsets et illustrée par la problématique de l'étude de l'érosion des sols.

2.1 Spectre des modèles utilisés et leur intérêt pour la fouille

Les experts de champs scientifiques divers (par exemple, les géologues, physiciens ou épidémiologistes) expriment souvent leurs connaissances sous la forme de modèles. Par exemple, des experts de l'érosion des sols (Lane et Nearing, 1989 ; Morgan, 2001 ; Atherton, 2005) développent des modèles mathématiques (fonctions de plusieurs variables) pour estimer le risque d'érosion selon un ensemble de paramètres environnementaux (par exemple, la végétation, la géologie, les précipitations, la pente). De façon similaire, les épidémiologistes (Bailey, 1975 ; Burattini *et al.*, 2008 ; de Castro Medeiros *et al.*, 2011) développent des modèles pour estimer le nombre de personnes infectées par la Dengue, en fonction du nombre d'habitants, du cycle de vie des moustiques et des saisons. De tels modèles renferment une connaissance experte pour un contexte donné. En outre, il y a de plus en plus de variables (parmi celles utilisées dans ces modèles) pour lesquelles les valeurs sont facilement collectées aujourd'hui (grâce à la télédétection par exemple). Utiliser simultanément les modèles experts et les données disponibles apparaît maintenant comme un défi.

L'originalité de cette contribution est d'exploiter les modèles existant dans la littérature pour construire de nouvelles contraintes à insérer dans le processus de découverte de motifs. La pertinence des motifs peut ainsi être améliorée, et l'extraction devient plus rapide et efficace. De plus, les experts ne sont pas sollicités à chaque extraction pour exprimer la (partie de) connaissance utile à spécifier. Nous nous concentrons sur les itemsets et nous considérons les modèles définis par les experts sous la forme de fonctions mathématiques de plusieurs attributs/variables. Nous présentons plusieurs exemples de modèles linéaires, polynomiaux, et même non linéaires qui peuvent être utilisés pour améliorer la pertinence des motifs. Nous mettons en évidence les propriétés théoriques de ces modèles par rapport

au domaine de motif des itemsets ; ces propriétés peuvent être utilisées pour élaguer l'espace de recherche et ainsi améliorer l'efficacité calculatoire.

L'avantage de ces modèles est qu'ils sont précis, synthétiques et exprimés dans un langage universellement reconnu. Par exemple, en mécanique, on peut exprimer la période d'un objet de masse m pendu à un ressort de rigidité k sous la forme d'une fonction $f : \mathbb{R}^+ \times \mathbb{R}^+ \rightarrow \mathbb{R}, f(m,k) = 2\pi\sqrt{\frac{m}{k}}$. Supposons maintenant que l'on veuille analyser un ensemble de données portant sur des ressorts. On dispose de différents relevés de leurs fréquence d'oscillation, masse et rigidité. Il serait intéressant de voir, par exemple, quels sont les cas associés à une faible fréquence. Il faudrait alors utiliser un seuil minimum f_{min} afin de ne garder que les cas présentant une grande période d'oscillation (faible fréquence). Si l'application de f sur m et k ne correspond pas aux résultats attendus, les motifs mettront en avant des données contredisant le modèle physique. De plus, si l'on dispose de données sur leur environnement (indiquant par exemple si les expériences ont été effectuées à proximité de champs magnétiques, sur Terre ou dans l'espace, avec différents matériaux), ces informations complémentaires pourront alors être éventuellement présentes elles aussi dans les motifs.

Dans le cas de l'érosion, on peut distinguer deux types de modèles : les modèles empiriques et les modèles physiques. Les modèles empiriques sont construits exclusivement grâce à des expériences. Le modèle polynomial *Universal Soil Loss Equation* (USLE) de Wischmeier et Smith (1978) ainsi que le modèle linéaire proposé par Atherton (2005) en sont des exemples typiques : il s'agit, à partir de données limitées à une zone d'étude, d'assigner les bons coefficients aux différents paramètres choisis afin d'obtenir un résultat conforme aux observations. Les paramètres n'expriment pas d'autre connaissance que le fait qu'ils sont réputés influencer le résultat. Les modèles physiques sont les modèles quantitatifs basés sur des propriétés physiques, et éventuellement calibrés par des expérimentations. Par exemple, les modèles *Water Erosion Prediction Project* (WEPP) de Lane et Nearing (1989) et *Revised Morgan-Morgan Finley* (RMMF) de Morgan (2001) prennent tous deux en compte plusieurs modèles physiques (non linéaires et non polynomiaux). RMMF scinde le processus d'érosion en deux étapes : le détachement de particules dû à la pluie (voir figure 3.11) et celui dû au ruissellement. Chacune des étapes est basée sur un sous-modèle physique. Leurs résultats respectifs sont additionnés pour obtenir une évaluation des pertes de sol annuelles.

Paramètres	**Domaine de valeurs**
Indice de détachement du sol (in g/J) x_K	dépend du type de sol
Précipitations annuelles (en mm) x_R	[0 ; 12 000]
Proportion de pluie stoppée par la végétation x_A	[0 ; 1]
Pourcentage de couverture de la canopée x_{CC}	[0 ; 1]
Intensité des précipitations (en mm/h) x_I	{10 ; 25 ; 30} selon le climat de la zone étudiée
Hauteur de la végétation (en m) x_{PH}	[0 ; 130]

$$f(x_K, x_R, x_A, x_{CC}, x_I, x_{PH}) = x_k * [x_R * x_A * (1 - x_{CC}) * (11.9 + 8.7 \log x_I) + (15.8 + x_{PH}{}^{0.5}) - 5.87] * 10^{-3}$$

Figure 3.11 – Modèle de détachement de particules causé par la pluie, dans RMMF

Dans cette thèse, l'exemple moteur concerne les modèles experts pour étudier l'érosion des sols. Dans ce contexte, notre travail se focalise essentiellement sur les motifs tendant à être corrélés à une forte érosion, tout en considérant l'influence des autres paramètres environnementaux qui ne sont pas considérés par les modèles (par exemple, les paramètres relatifs aux activités humaines). Il s'avère que les contraintes basées sur de tels modèles permettent d'évaluer ou d'enrichir la connaissance des experts, et peuvent aussi mettre en avant des relations contradictoires.

2.2 Définitions formelles

2.2.1 Modèles experts

Soit $D_{model} = \{x_1, x_2, ..., x_n\}$ l'ensemble des **variables/attributs** d'un modèle expert. On note $dom(x_j)$ le **domaine** de x_j, c'est-à-dire l'ensemble des valeurs possibles de l'attribut x_j.

Un **modèle mathématique expert** est une fonction f définie comme suit :

$$\begin{array}{rccc} f: & dom(x_1) \times dom(x_2) \times ... \times dom(x_n) & \rightarrow & \mathbb{R} \\ & x = (x_1, x_2, \ldots, x_n) & \mapsto & f(x) \end{array}$$

Ce modèle représente la connaissance de l'un ou de plusieurs experts à propos d'un phénomène donné ; de tels modèles existent dans la littérature du domaine étudié.

Ces modèles sont des fonctions numériques. Ainsi, quand les experts veulent intégrer une donnée catégorielle (c'est-à-dire nominale, comme par exemple la variable « type de sol »), cette donnée doit être transformée en un nombre. La correspondance est faite par les experts, d'après leur connaissance du domaine. Cette connaissance va de pair avec le modèle utilisé. Par exemple, le modèle RMMF prend en entrée le type de sol avec comme valeurs nominales « sable » ou « limon » par exemple. Dans son article, l'expert associe à chaque type de sol une valeur numérique appelée « indice de détachement du sol » (*soil detachment index*). Cette valeur est déterminée d'après les expériences passées déjà effectuées ; en l'occurrence, « sable » vaut 1,2 et « limon » vaut 0,8. De la même façon, le modèle proposé par ATHERTON (2005) intègre des données de couverture des sols (par exemple, « forêt dense », ou « plantation de cannes à sucre »). Les valeurs numériques associées à chaque type de couverture du sol, données dans la contribution, sont proportionnelles à leur impact supposé ou connu des experts sur l'érosion du sol ; en l'occurrence, « forêt dense » vaut 1 et « plantation de cannes à sucre » vaut 4.

2.2.2 Itemsets

Établissons maintenant la connexion entre la définition donnée par AGRAWAL et SRIKANT (1994) d'un itemset I et les modèles précédents.

Soit $D_{DB} = \{d_1, d_2, ..., d_{n'}\}$ les **dimensions/attributs** d'une base de données binaire DB. Par exemple, $D_{DB} = \{\textit{précipitations}, \textit{végétation}, \textit{type de sol}, ...\}$. D_{DB} doit couvrir au moins un des attributs du modèle expert étudié, c'est-à-dire $D_{DB} \cap D_{model} \neq \emptyset$. Le domaine de $d_{j'}$ dans DB, noté $dom(d_{j'})$, est l'ensemble de ses valeurs catégorielles. Par exemple,

le domaine de l'attribut « précipitations annuelles » x_R du modèle RMMF (voir figure 3.11 page 74) est $dom(x_R) = [0\,;\,12\,000]$. Dans la base de données, on pourra pas exemple la discrétiser de la manière suivante : $dom(d_R) = \{$« $x_R \in [0\,;\,2\,000]$ », « $x_R \in [2\,001\,;\,3\,200]$ », « $x_R \in [3\,201\,;\,12\,000]$ »$\}$, par exemple. Nous considérons maintenant que chaque valeur d'origine est transformée en valeur catégorielle, dans laquelle attribut et valeur sont tous deux représentés (par exemple, la valeur « sol ultramafique » de l'attribut x_K sera notée « x_K = sol ultramafique »). Plus formellement, les valeurs d'un $dom(d_{j'})$ peuvent être vue comme des paires (attribut, valeur).

Soit $\mathcal{I} = \bigcup_{d_j \in D_{DB}} dom(d_j)$ l'ensemble de toutes les valeurs contenues dans la base de données DB. Une valeur $i \in \mathcal{I}$ est appelée **item**. Le domaine de motif est $\mathcal{L} = \{X \in 2^{\mathcal{I}} \mid \nexists i, i' \in X$ tels que $i, i' \in dom(d_j)$ avec $d_j \in D_{DB}\}$. En d'autres termes, un motif est la combinaison de valeurs (catégorielles) provenant toutes de différents attributs. En prenant n' le nombre d'attributs, un ensemble d'items $X = \{i_1, i_2, ..., i_k\} \in \mathcal{L}$, avec $k \leq n'$, est un **itemset**. Les attributs d'un itemset X **participant à** D_{model}, notés $Atts(X, D_{model})$, sont les attributs des items de X qui appartiennent aussi à D_{model}. Par exemple, $X =$ {« $x_K = sol\ ultramafique$ », « $x_R = [2001, 3200]$ », « $mine$ », « $chemin$ »} est un itemset. Ses attributs participant au modèle RMMF sont $Atts(X, D_{\text{RMMF}}) = \{x_K, x_R\}$.

2.2.3 Recherche d'itemsets sous contraintes

Le problème de l'extraction d'itemsets sous contrainte(s) est de trouver l'ensemble des itemsets satisfaisant un prédicat de sélection q dans les données (où q peut être une conjonction de contraintes primitives). Cet ensemble, parfois appelé « théorie de DB par rapport à $\mathcal{L}$ et q », est noté $\mathcal{T}h(\mathcal{L}, DB, q)$ (Mannila *et al.*, 1997). Ici, le prédicat q est une conjonction de contraintes basées sur des mesures objectives et subjectives. La première contrainte que l'on utilisera sera la contrainte de fréquence minimale : un motif est sélectionné si et seulement s'il apparaît dans la base de données plus souvent qu'un seuil $minsup$ défini par l'utilisateur. La deuxième contrainte est aussi une contrainte de seuil minimum, mais par rapport à un modèle expert, et est notée $q_{f\geq}$: un motif est sélectionné si et seulement si sa valeur selon un modèle expert f est plus grande qu'un seuil $minf$ défini par l'utilisateur.

2.3 Des motifs aux modèles

Une approche simple pour profiter d'une connaissance d'un domaine particulier est d'en dériver des contraintes primitives, et de les utiliser durant la fouille de données. Nous proposons d'utiliser des contraintes dérivées de modèles experts au lieu de contraintes *ad-hoc* définies par des experts. Les premières sont bien plus expressives que les règles basiques « si ..., alors ... », puisqu'un seul modèle expert peut générer un très grand nombre de ces règles.

Plusieurs types de contraintes pourraient être dérivés selon les données, selon les modèles experts, mais aussi selon le problème étudié. Dans cette partie, nous nous concentrons sur une contrainte qui est relative à celle de la fréquence minimale, même si leurs propriétés théoriques respectives diffèrent à bien des égards.

On peut définir une contrainte qui filtre les motifs dont la valeur par f est supérieure ou égale à un seuil arbitraire. En d'autres mots, cette contrainte ne garde que les motifs pour

lesquels les valeurs prédites par le modèle expert sont supérieures ou égales à ce seuil arbitraire. Selon ce que f exprime, cette contrainte aura différentes significations. Par exemple, si f estime la perte de sol (en $kg.m^{-2}$ par an) comme dans le modèle RMMF, cette contrainte filtrera les motifs correspondant à une perte de sol potentielle supérieure à une quantité donnée. En l'absence de « vérité terrain [9] » (sur les pertes de sol par exemple), cette contrainte permettra de mettre en avant si de telles pertes tendent à être fréquentes dans la zone étudiée, et dans quelles situations (grâces aux valeurs des autres paramètres environnementaux présents dans le motif). De plus, cela permettra aussi de voir avec quels autres facteurs (ceux non couverts par le modèle) ces pertes de sol sont fréquemment corrélées dans les données. En présence de « vérité terrain », cette contrainte permettra de comparer les prédictions des modèles experts avec la « vérité » fournie par les données. Les motifs en accord avec le modèle sont intéressants car ils seront doublement validés : d'une part par la vérité terrain, d'autre part par la connaissance du domaine (c'est-à-dire par le modèle expert). Ils renforcent ainsi notre confiance envers la solidité du modèle le plus faible (soit celui fourni par les données, soit celui exprimé par les experts). De plus, les items supplémentaires des motifs trouvés peuvent compléter les explications du modèle expert. Les motifs en contradiction avec la prédiction du modèle expert sont aussi intéressants, car ils permettent d'identifier des corrélations absentes du modèle expert (et ainsi améliorer ce modèle).

Soit $X \in \mathcal{L}$ un itemset, f un modèle expert, et $minf \in \mathbb{R}$ un seuil arbitraire. Notre contrainte de seuil dérivée du modèle expert f est définie comme suit :

$$q_{f\geq}(X) \equiv f(X) \geq minf$$

Dans la suite, nous prendrons l'exemple suivant comme modèle de référence :

$$\begin{aligned} f : \quad [1\,;\,16] \times [0\,;\,4\pi] \times [1\,;\,100] \subset \mathbb{R}^3 \;&\rightarrow\; [0,5] \subset \mathbb{R} \\ f(x_1, x_2, x_3) \;&=\; \sqrt{x_1} - \frac{1}{2}\cos(x_2) \times \log_{10}(x_3) \end{aligned}$$

Ce modèle, même s'il ne reflète aucun modèle expert en particulier, montre que les modèles à plusieurs variables et à valeur réelle que nous étudions ne sont pas nécessairement linéaires, et n'ont pas nécessairement un comportement monotone.

2.3.1 Valeur d'un itemset X par un modèle expert f

La contrainte décrite à l'instant implique de pouvoir calculer une valeur prédite par un modèle expert pour un itemset donné. Prenons l'exemple de modèle expert que l'on vient de définir. Si un motif X est {« $x_1 \in [3\,;\,5]$ », « $x_2 = 3$ », « $x_3 = A$ »}, quelle est sa valeur par f ? En d'autres termes, quelle est la prédiction du modèle expert f pour les valeurs $x_1 \in [3\,;\,5]$, $x_2 = 3$, et $x_3 = A$?

Pour un itemset {« $x_1 = 1$ », « $x_2 = 3$ », « x_3=A », « mine »}, nous devons seulement calculer $f(1, 3, 10)$ si l'on suppose que « $x_3 = A$ » est associé à la valeur 10 par les experts. Ce cas-ci est simple car tous les attributs du modèle apparaissent dans le motif. De plus, ces attributs sont associés à une seule valeur (et non un intervalle de valeurs). Notons que

9. La vérité terrain désigne les valeurs réelles que l'on peut relever sur le terrain

l'on ne peut pas prendre en compte l'item « mine » dans f, puisque cette information n'est pas intégrée dans le modèle expert. Néanmoins, cet item reste intéressant car il donne une information additionnelle sur la connaissance capturée par le motif. Plus formellement, si $Atts(X, D_{model}) = D_{model}$ et $\forall i \in X$, l'item i représente une simple valeur, alors $f(X = \{i_1, i_2, ..., i_n, ..., i_k\}) = f(i_1, i_2, ..., i_n)$. Cependant, dans des cas plus généraux, nous devons résoudre deux problèmes.

Tout d'abord, certains attributs du modèle peuvent ne pas être disponibles dans les données ($D_{model} \not\subseteq D_{DB}$). De la même façon, certains attributs du modèle peuvent ne pas être exprimés dans le motif. Considérons l'itemset X'= {« x_1=1 », « x_3=A », « mine »}. Celui-ci ne contient pas tous les attributs du modèle : x_2 n'y est pas exprimé. On peut trouver les bornes supérieures et inférieures pour $f(X')$ en prenant les valeurs de x_2 pour lesquelles f est maximale/minimale. Dans notre exemple, si $x_2 = \pi$ ou 3π, alors $f(1, x_2, 10) =$ 1,5. Cette valeur de f est la plus grande valeur possible avec les valeurs de x_1 et x_3. On peut facilement en déduire que $0{,}5 \leq f(X') \leq 1{,}5$, même si x_2 n'est pas représenté dans X'. Plus formellement, prenons un itemset $X = \{i_1, i_2, ..., i_n, ..., i_k\}$ et un modèle expert $f(x_1, ..., x_j, ...x_n)$. On a $\forall x_j \in D_{model}, x_j \notin Atts(X, D_{model})$:

$$\min_{\forall i_j \in dom(x_j)} f(i_1, ..., i_j, ...i_n) \leq f(X) \leq \max_{\forall i_j \in dom(x_j)} f(i_1, ..., i_j, ...i_n)$$

Deuxièmement, les domaines des valeurs du modèle et ceux des itemsets peuvent être différents. Le modèle mathématique est basé sur des valeurs numériques, alors que les itemsets sont basés sur des valeurs catégorielles (en faisant si nécessaire une discrétisation). On a souvent des itemsets représentant un mélange d'intervalles, de valeurs numériques et de valeurs catégorielles. Par exemple, considérons l'itemset $X'' =$ {« $x_1 = 4$ », « $x_2 \in [0\,;\, 2\pi[$ », « $x_3 = A$ »}. Un intervalle de valeurs est associés à l'attribut x_2. Cet item « $x_2 \in [0, 2\pi[$ » provient d'une étape de pré-traitement, durant laquelle la domaine de x_2 a été discrétisé en plusieurs intervalles (disjoints). Comme précédemment, il est possible de trouver $f(X'')$ en étudiant les bornes supérieures et inférieures pour lesquelles $x_2 \in [0, 2\pi[$. Si on étudie la fonction *cos* sur $[0\,;\, 2\pi[$, alors on sait que $f(X'')$ est maximal quand $x_2 = \pi$ (dans ce cas $f(4, \pi, 10) = 2{,}5$), et minimal quand $x_2 = 0$ (dans ce cas, $f(4, 0, 10) = 1{,}5$). Ainsi, on peut en déduire que $1{,}5 \leq f(X'') \leq 2{,}5$. La formule précédente peut être généralisée à n'importe quel item $i_j \in X$ représentant un intervalle $[inf_j, sup_j]$ d'un attribut x_j d'un modèle :

$$\min_{\forall i_j \in [inf_j, sup_j]} f(i_1, ..., i_j, ...i_n) \leq f(X) \leq \max_{\forall i_j \in [inf_j, sup_j]} f(i_1, ..., i_j, ...i_n)$$

En conséquence, la valeur d'un itemset X par un modèle expert f peut être un intervalle de valeurs. La définition de notre contrainte de seuil peut donc être étendue de la façon suivante :

$$\begin{aligned} \text{Pour } f(X) = [inf_X, sup_X], \quad q_{f\geq}(X) &\equiv f(X) \geq minf \\ &\equiv inf_X \geq minf \end{aligned}$$

Il est important maintenant d'étudier les propriétés théoriques de cette contrainte pour en estimer la complexité et en améliorer l'efficacité calculatoire. Par exemple, la fréquence est

une fonction monotone décroissante. Ainsi, si un itemset n'est pas fréquent, alors tous ses sur-ensembles ne le sont pas non plus. Cette propriété d'« anti-monotonie » a été abondamment utilisée dans les algorithmes de recherche de motifs fréquents afin de rendre possible leur passage à l'échelle. Nous détaillons donc maintenant les propriétés des modèles étudiés qui peuvent aider à élaguer l'espace de recherche.

2.3.2 Propriétés théoriques des modèles par rapport aux itemsets

Propriétés sur les relations entre un itemset et ses sur-ensembles Soient $X, Y \in \mathcal{L}$ deux itemsets tels que $X \subset Y$. Si X et Y expriment les mêmes attributs que ceux participant au modèle f, alors ils ont les mêmes items pour ces attributs, et donc $f(Y) = f(X)$. En d'autres mots, Y ne diffère de X seulement parce qu'il contient des items additionnels non considérés par le modèle, ce qui n'impacte donc pas le calcul de $f(Y)$. Dans ce cas, si $f(X) < minf$, alors $f(Y) < minf$. Par exemple, l'itemset $X'' = \{\text{« } x_1 = 4 \text{ »}, \text{« } x_2 \in [0\,;\, 2\pi[\text{ »}, \text{« } x_3 = A \text{ »}\}$ a la même valeur par f que $Y_1'' = \{\text{« } x_1 = 4 \text{ »}, \text{« } x_2 \in [0\,;\, 2\pi[\text{ »}, \text{« } x_3 = A \text{ »}, \text{« } mine \text{ »}\}$ et que $Y_2'' = \{\text{« } x_1 = 4 \text{ »}, \text{« } x_2 \in [0\,;\, 2\pi[\text{ »}, \text{« } x_3 = A \text{ »}, \text{« } mine \text{ »}, \text{« } chemin \text{ »}\}$. En effet, $f(4, 0, 10) \leq f(X'') \leq f(4, \pi, 10)$, de même que $f(Y_1'')$ et $f(Y_2'')$, puisque les attributs x_1, x_2 et x_3 de f ont les mêmes valeurs. En conséquence, si $f(X'') < minf$, alors $f(Y_1'')$ et $f(Y_2'')$ sont aussi inférieurs à $minf$.

Propriété 2.1

Soit $X \in \mathcal{L}, \forall Y \in \mathcal{L} \mid X \subset Y$ et $Atts(X, D_{model}) = Atts(Y, D_{model})$.
Si $q_{f\geq}(X)$ est faux, alors $q_{f\geq}(Y)$ est faux.

La situation est plus compliquée lorsque des attributs de f sont exprimés dans Y et pas dans X (dernière possibilité pour l'hypothèse $X \subset Y$). Par exemple, prenons l'itemset $X = \{\text{« } x_2 \in [0\,;\, 2\pi[\text{ »}, \text{« } x_3 = A \text{ »}\}$ et un de ses sur-ensembles $Y_1 = \{\text{« } x_1 = 16 \text{ »}, \text{« } x_2 \in [0\,;\, 2\pi[\text{ »}, \text{« } x_3 = A \text{ »}\}$. L'attribut x_1 n'est effectivement par exprimé dans X, mais l'est dans Y_1. On sait que $0{,}5 = f(1{,}0, 10) \leq f(X) \leq f(16, \pi, 10) = 4{,}5$ puisque $dom(x_1) = [1\,;\, 16]$. Si le seuil minimum pour f est 2, on peut avoir $f(X) < minf$ même si $f(Y_1) = [3{,}5\,;\, 4{,}5] > minf$. D'un autre côté, si $minf = 5$, alors nous pouvons être sûrs que tous les sur-ensembles de X satisfont $f(X) < 5$. Pour n'importe quelle valeur de x_1 appartenant à $[1\,;\, 16]$, les valeurs minimum et maximum de f sont 0,5 et 4,5.

Propriété 2.2

Soit $X \in \mathcal{L}$ tel que $inf_X \leq f(X) \leq sup_X$.
Si $q_{f\geq}(X)$ est faux et $sup_X < minf$, alors $\forall Y \in \mathcal{L} \mid X \subset Y$, $q_{f\geq}(Y)$ est faux.

Propriété sur les itemsets partageant les mêmes attributs Les propriétés précédentes décrivent les liens entre un itemset et ses sur-ensembles. Ces liens permettent de borner les valeurs f pour les sur-ensembles d'un itemset donné. Analyser la fonction f permet de mettre en avant d'autres propriétés entre les itemsets par rapport à un modèle donné. Cependant, ceci peut s'avérer complexe du fait de la nature des fonctions étudiés (éventuellement non linéaires sur plusieurs attributs). Il est difficile d'étudier dans sa globalité la

monotonicité d'une fonction sur plusieurs attributs. Notre solution consiste à analyser la croissance de la fonction par rapport à un seul attribut à la fois (les autres étant considérés comme constants). Cette solution revient à étudier les dérivées partielles de f sur chacun des attributs. Pour un attribut donné, l'objectif est d'identifier les intervalles de f dans lesquelles la fonction est monotone. Ainsi, pour chaque intervalle, il est possible de dériver des propriétés permettant d'élaguer l'espace de recherche.

Considérons les itemsets X={« x_1=4 », « $x_2 \in [\pi/2\,;\,\pi/2[$ », « x_3=A »} et Y={« x_1=4 », « $\boldsymbol{x_2 \in [0\,;\,\pi/2[}$ », « x_3 =A »}. Notons au passage que $Atts(X, D_{model}) = Atts(Y, D_{model})$. L'analyse de la fonction f par rapport à x_2 montre qu'elle est strictement croissante sur $[0\,;\,\pi]$ (c'est-à-dire, $\frac{\partial f}{\partial x_2} > 0$ sur $[0\,;\,\pi]$). Puisque X est supérieur à Y par rapport à X_2 ($Y \prec_{x_2} X$), on a $f(Y) <$$f(X)$. En effet, $f(X) = [2\,;\,2{,}5]$ et $f(Y) = [2{,}5\,;\,2]$. En conséquence, si $f(X) < minf$, alors $f(Y) < minf$ (même si $X \not\subset Y$).

De la même façon, considérons $Y'' = \{$« $\boldsymbol{x_1 = 1}$ », « $\boldsymbol{x_2 \in [0\,;\,\pi/2[}$ », « $x_3 = A$ »$\}$. On sait que $f(Y'') < f(X)$, car $\frac{\partial f}{\partial x_1} > 0$ sur $dom(x_1)$ et $Y'' \prec_{x_1} X$.

Plus formellement, une **relation d'ordre total**, notée $\prec_{x_j}$, peut être définie pour chacun des attributs $x_j \in D_{DB} \cap D_{model}$. Si $i, i' \in dom(x_j)$ représentent des intervalles, c'est-à-dire si $i =$ « $x_j \in [a, b]$ » et $i' =$ « $x_j \in [c, d]$ », alors $i \prec_{x_j} i'$ si et seulement si $b < c$. Par exemple, « $x_R \in [0\,;\,2\,000]$ » $\prec_{x_R}$ « $x_R \in [2\,001\,;\,3\,200]$ » car $2\,000 < 2\,001$. Si $i, i' \in dom(x_j)$ sont des valeurs nominales associées à des valeurs numériques num_i et $num_{i'}$ par des experts, alors $i \prec_{x_j} i'$ si et seulement si $num_i < num_{i'}$. Par exemple, « $x_K = sol\ volcanique$ » $\prec_{x_j}$ « $x_K = sol\ ultramafique$ » car les experts associent *sol volcanique* à 8 et *sol ultramafique* à 10.

Sur ces définitions, les propriétés précédentes peuvent être formalisées comme suit :

Propriété 2.3

Sous les hypothèses et avec les notations suivantes :

— Soient deux itemsets $X, Y \in \mathcal{L} \mid Atts(X, D_{model}) = Atts(Y, D_{model})$.

— $X.x_j$ désigne la valeur de l'attribut x_j dans X.

— Pour chaque attribut x_j de X et de Y appartenant au modèle, on a :

$$\frac{\partial f}{\partial x_j} > 0 \text{ sur } [a\,;\,b] \ \wedge\ X.x_j, Y.x_j \in [a\,;\,b] \ \wedge\ Y \prec_{x_j} X,$$
$$\text{ou } \frac{\partial f}{\partial x_j} < 0 \text{ sur } [a\,;\,b] \ \wedge\ X.x_j, Y.x_j \in [a\,;\,b] \ \wedge\ X \prec_{x_j} Y.$$

Si $q_{f\geq}(X)$ est faux, alors $q_{f\geq}(Y)$ est faux.

Remarquons que l'impact de cette propriété dépend de la discrétisation. On ne pourrait pas faire de telles déductions avec les itemsets $X = \{$« $x_1 = 4$ », « $\boldsymbol{x_2 \in [\pi, 2\pi[}$ », « $x_3 = A$ »$\}$ et $Y = \{$« $x_1 = 4$ », « $\boldsymbol{x_2 \in [0, \pi[}$ », « $x_3 = A$ »$\}$, car la fonction f croît avec x_2 sur $[0\,;\,\pi]$ et décroît sur $[\pi\,;\,2\pi]$. Soit x_j un attribut tel que son domaine est discrétisé en intervalles pour lesquels f est monotone. Intuitivement, plus le nombre d'items associés à chaque intervalle est grand, plus cette propriété permet d'élaguer des motifs.

À partir de ces propriétés, nous allons dans la section suivante intégrer les contraintes du

modèles expert dans l'algorithme d'extraction d'itemsets.

2.4 Insertion des modèles experts dans la fouille de motifs

La contrainte proposée est relativement simple à intégrer dans les algorithmes de fouille de données, puisqu'elle possède des propriétés similaires à celles des contraintes utilisées classiquement pour extraire des itemsets (par exemple, la contrainte de fréquence minimale). Seules quelques modifications devront être faites puisque la vérification de la contrainte ne nécessite aucun accès à la base de données ou une quelconque autre ressource. Seule la génération des motifs candidats sera impactée. L'intérêt d'utiliser cette contrainte de seuil basée sur les modèles expert durant l'extraction (et non en tant que post-traitement) est de rapidement supprimer les motifs inintéressants, ce qui améliore la performance et donc le passage à l'échelle.

Dans nos expériences, nous avons intégré une telle contrainte à l'algorithme `Close-By-One` (`CBO`) de KUZNETSOV et OBIEDKOV (2002). Initialement développé pour l'analyse de concepts formels, cet algorithme est utilisé dans notre contexte pour extraire les itemsets fermés fréquents. Le principe de l'algorithme est d'effectuer une recherche en profondeur dans le treillis de Galois liant itemsets et transactions, afin de calculer les motifs clos. A chaque étape, l'algorithme étend un motif en lui ajoutant chaque item supporté par toutes les transactions du motif en cours. Un test de canonicité est aussi effectué pour éviter la duplication des solutions, et le parcours inutile de certaines branches de recherche. L'algorithme (en particulier son test de canonicité) repose sur le fait qu'il existe un ordre total entre les items (par exemple, l'ordre lexicographique). Dans notre cas, les attributs du modèle sont énumérés en premier (par ordre lexicographique), suivis des autres attributs de la base de données. Pour chaque attribut, les items sont ordonnés par leur valeur.

Algorithmes Les algorithmes 5 et 6 décrivent cette approche. Le paramètre (X, T) de l'algorithme 6 représente le motif fermé X courant (à étendre), avec l'ensemble des transactions T dans lequel il apparaît. Le paramètre A est l'itemset qui a été utilisé pour générer X, et le paramètre i est le dernier item qui a été ajouté à X ($X = A \supseteq \{i\}$). Le paramètre B est l'ensemble des attributs qui sont utilisables pour l'extension. La ligne 1 effectue le test de canonicité pour éviter de générer le même motif deux fois. La ligne 5 enregistre le motif courant (avec sa fréquence $|T|$) parmi les solutions. Les lignes 8 et 10 énumèrent chaque item pouvant être utilisé pour étendre X. La ligne 9 calcule les attributs pouvant être utilisés pour les prochaines extensions. Les lignes 10 à 13 calculent la fermeture des extensions de X, leurs transactions, et exécutent les prochaines itérations. La fonction ϕ de la ligne 12 représente l'opérateur de fermeture. Pour rappel (voir l'état de l'art, chapitre 2, section 2.1.3 page 42), l'opérateur de fermeture est la composition de deux fonctions : $\phi = items \circ transactions$ [10]. Appliquer *transactions* à un ensemble d'items donne les transactions dans lesquelles cet ensemble apparaît (ligne 11). Appliquer *items* à un ensemble de transactions donne les items qui sont communs à toutes ces transactions (ligne 12).

10. Ces deux fonctions ont été renommées pour éviter les confusions ; *items* correspond à la fonction $f : 2^{\mathcal{T}} \to 2^{\mathcal{I}}$ de l'état de l'art, *transactions* correspond à la fonction $g : 2^{\mathcal{I}} \to 2^{\mathcal{T}}$.

Les seules différences entre cet algorithme et l'original se trouvent aux lignes 2, 3, 4 et 7. La ligne 2 vérifie la contrainte de support. La ligne 3 exprime la propriété 2.2 de notre contrainte : si la borne supérieure de $f(X)$ (ou simplement sa valeur si l'on n'a pas affaire à un intervalle) est inférieure à $minf$, alors tous les sur-ensembles de X n'ont pas besoin d'être explorés car ils ne satisferont pas la contrainte du modèle. Les lignes 4 et 7 expriment la propriété 2.1 : si l'extension de X viole la contrainte du modèle, alors tous ses sur-ensembles partageant les mêmes attributs participant au modèle n'auront pas besoin d'être explorés.

Algorithme 5 : `CBO` avec Contrainte de Modèle

Entrées : DB : base de données transactionnelle d'itemsets, D_{DB} : , $minsup$: seuil de fréquence minimum

Sorties : $Closed$: L'ensemble des itemsets fermés fréquents dont les valeurs par le modèle f sont supérieures à $minf$

```
1 Closed ← ∅
2 pour chaque d_k ∈ D_DB faire
3     B ← {d_l ∈ D_DB | d_k < d_l}
4     pour chaque i ∈ dom(d_k) faire
5         CBO_Recur ( φ({i}), transactions({i}), {i}, i, B, Closed )
6 retourner Closed
```

Algorithme 6 : `CBO_Recur`

Entrées : ((X,T), A, i, B, Closed)

```
1  si {h | h ∈ X\A et h ≺ i} = ∅ alors
2      si |T| ≥ minsup alors
3          si sup_X( f(X) ) < minf alors stop;
4          si f(X) ≥ minf alors
5              Closed ← Closed ∪ {(X, |T|)}
6              B_tmp ← B
7          sinon B_tmp ← B\{d ∈ D_DB | d ∉ D_model} ;
8          pour chaque d_k ∈ B_tmp faire
9              B ← B\{d_l ∈ D_DB | d_l ≤ d_k}
10             pour chaque j ∈ {h | h ∈ dom(d_k) et i ≺ h} faire
11                 U ← T ∩ transactions({j})
12                 Y ← items(U); // Y = φ(X ∪ {j})
13                 CBO_Recur ( (Y,U), X, j, B , Closed )
```

Notre approche est totalement générique. La plupart des algorithmes de fouille de données (tels que `A-Priori`, `FP-Growth`, `Eclat`) auraient pu être utilisés au lieu de `Close-By-One`. Cependant, en fonction de la stratégie algorithmique, l'exploitation de certaines des proprié-

tés servant à couper l'espace de recherche peut ne pas être simple. Par exemple, il est difficile d'exploiter la propriété 2.3 à cause de la stratégie de génération des motifs candidats basée sur la fermeture, alors que cela est plus facile pour les algorithmes cités car les itemsets sont étendus item par item.

Dans l'algorithme 7 (`A-Priori`), nous avons utilisé les propriétés 2.1 (ligne 5), 2.3 (ligne 8) et 2.2 (ligne 9). Le principal inconvénient pour exploiter la propriété 2.3 est qu'il faut maintenir un ordre entre les items d'un même domaine de variable, et que cet ordre varie selon certains intervalles de valeur. Ici, on émet l'hypothèse que la fonction f du modèle expert est monotone sur chaque variable. En outre, l'algorithme original étant un algorithme en largeur (de type « générer puis tester »), ces contraintes n'élagueront l'espace de recherche que pour une taille d'itemset donnée. Le bénéfice acquis ne sera pas forcément reporté sur la génération suivante (c'est-à-dire, la génération des itemsets de taille supérieure), car l'ensemble DB' (ligne 5) peut regrouper les items précédemment éliminés.

Algorithme 7 : `A-Priori` avec Contrainte de Modèle

1 $L_1 \leftarrow \bigcup_{d_j \in D_{DB}} dom(d_j)$
2 $k \leftarrow 1$
3 **tant que** $L_k \neq \emptyset$ **faire**
4 **pour chaque** itemset $X \in L_k$ **faire**
5 **si** $sup(X, D_{DB}) \geq minsup \wedge q_{f\geq}(X)$ est vraie **alors**
6 $F_k \leftarrow F_k \cup \{X\}$
7 **sinon si** $q_{f\geq}(X)$ est fausse **alors**
8 $L_k \leftarrow L_k \setminus \{Y \in L_k \mid Atts(Y, D_{model}) = Atts(X, D_{model}) \wedge \forall x_j \in Atts(Y, D_{model}) \cap Atts(X, D_{model}), X.x_j > Y.x_j\}$
9 $L_{k+1} \leftarrow \{X \in \mathcal{L} \mid |X| = k+1 \text{ et } \forall Y \subset X, Y \in \bigcup_{j \leq k} F_j \wedge sup(f(Y)) \geq minf\} \setminus X \in \mathcal{L} \mid |X| = k+1 \text{ et } \forall Y \subset X, Atts(Y, D_{model}) = Atts(X, D_{model}), q_{f\geq}(Y)$ est fausse
10 $k \leftarrow k+1$
11 **retourner** $\bigcup_k F_k$

Évaluation des performances L'utilisation de ces algorithmes nécessite à la fois des données et un modèle. Ce modèle devant faire partie de la littérature du domaine sur lequel portent les données, il est aussi nécessaire d'avoir des données réelles. Utiliser des données synthétiques n'a pas de sens, car aucun modèle ne peut exister sur un domaine « synthétique ». Pour cette raison, nous étudierons les performances de l'algorithme proposé dans l'étude de cas du chapitre suivant, portant sur l'érosion en Nouvelle-Calédonie.

Chapitre 4

Étude de cas : l'érosion en Nouvelle-Calédonie

Sommaire

1. Présentation générale

1.1 Zones d'étude

La Nouvelle-Calédonie est un territoire français insulaire d'environ 18 000 km^2 situé au large des côtes australiennes et néo-zélandaises. Ce territoire regroupe plusieurs îles, dont une principale appelée « Grande Terre » sur laquelle vit la très grande majorité de la population. Située en zone tropicale (juste au nord du tropique du Capricorne), la Nouvelle-Calédonie subit un climat tropical tempéré.

Ce climat, ainsi que les activités anthropiques [1], entraînent certaines conséquences facilitées par la nature géologique du territoire. Ces conséquences sont bien connues des géologues ; parmi elles, l'érosion altère à court et moyen termes l'occupation des sols. Les modifications sont suffisamment intenses pour pouvoir changer les paysages, et par voie de conséquence les habitats naturels, l'équilibre du vivant, ainsi que l'organisation économique, urbaine et sociétale de la population. En effet, l'érosion peut entraîner des glissements de terrains pouvant s'avérer dangereux notamment à proximité de zones habitées, déformer les paysages qui sont des ressources économiques car touristiques, modifier la trajectoire des eaux de pluie et ainsi nécessiter la construction d'infrastructures d'adaptation coûteuses. De nombreux mécanismes générant le phénomène d'érosion sont présents en Nouvelle-Calédonie, tels que l'érosion particulaire, les chutes et les mouvements de masse. L'eau étant connue comme le facteur principal d'érosion sur le territoire, les recherches se sont concentrées sur l'érosion hydrique. Deux grands types de phénomènes la caractérisent : l'érosion en nappe et l'érosion concentrée.

L'érosion en nappe (ou érosion diffuse) est provoquée par le type du sol (latéritique) qui tend à se désagréger et à laisser une croûte en surface sous l'action de la pluie (c'est l'effet « *splash* »). Ce phénomène, nommé « battance », s'effectue particulièrement lorsque les surfaces sont pauvres ou dénudées de végétation. La croûte de battance est propice au ruissellement en nappe et freine l'infiltration de l'eau. Par conséquent, des débris organiques et de fines particules minérales sont entraînés par l'eau.

L'érosion concentrée (ou érosion linéaire) correspond quant à elle au caractère linéaire des flux de matière dus au ruissellement. Les ravines sont, par exemple, le résultat de ce type d'érosion (Rouet *et al.*, 2009). La maîtrise des phénomènes érosifs représente ainsi un enjeu crucial, tout particulièrement en Nouvelle-Calédonie.

La biodiversité y est en effet remarquable et le taux d'endémisme exceptionnel pour les écosystèmes terrestres et marins. Les enjeux du développement durable dans de tels milieux sont donc considérables, d'autant qu'ils doivent s'envisager dans un contexte de changement climatique global et de pressions anthropiques croissantes (exploitations minières, agriculture, pêche, etc.). Ainsi, le territoire néo-calédonien connaît un développement économique exceptionnel avec la construction de deux nouvelles usines métallurgiques (de 2005 à 2009

1. d'origine humaine

pour la mine dite « de Goro » située dans le sud de la Grande Terre, voir figure 4.1). Ces usines, d'envergure mondiale, sont destinées au traitement du nickel. La Nouvelle-Calédonie possède un quart des ressources mondiales de nickel et assure 12% de la production mondiale de ce minerai. Les mines devant alimenter les nouvelles usines sont entrées en phase d'exploitation, et ce pour des décennies. Le pari d'un développement durable suppose de gérer harmonieusement et parallèlement, sur des espaces étroitement imbriqués, à la fois le classement du récif et la réussite des projets miniers et métallurgiques indispensables au rééquilibrage inscrit dans les Accords de Nouméa. L'inquiétude quant à l'impact environnemental des mines est d'autant plus forte que l'usine de Vale Inco [2] a subi deux fuites d'acides en Avril 2009 et Mai 2014, dont les effets sur l'environnement ne peuvent être bénins. Tout autre impact environnemental affecterait gravement à la fois le succès du projet industriel, mais aussi l'état naturel des régions exploitées et leurs alentours. Cette zone contient plusieurs réserves naturelles (par exemple, des zones humides protégées par la Convention de Ramsar [3]). Le lagon situé à proximité est classé au patrimoine mondial de l'UNESCO depuis 2008. Cette zone correspond aussi aux sites d'exploitation et à l'usine de traitement de Vale Inco. La période d'étude court sur la phase de démarrage de l'usine, ce qui présente l'intérêt de pouvoir étudier l'évolution de l'érosion avant et après l'usine.

La zone choisie pour cette étude de cas se situe à la pointe Sud de la Grande Terre. L'endroit présente en effet des objets d'étude très variés (présence de routes, de forêts, de lacs, de reliefs, de mines, de zones habitées, de divers types de végétation, de réserves naturelles).

1.2 Méthodes existantes d'évaluation de l'érosion

Bien que les mécanismes globaux (ruissellement, détachements de particules, mouvements de masse) impliqués dans les phénomènes érosifs en général soient connus par les experts, leurs modèles quantifiant cette érosion sont souvent centrés sur des cas particuliers, par exemple, sur des zones d'études agricoles (Renard *et al.*, 1997), et leur généralisation à d'autres cas n'est pas forcément adaptée ; il faut alors en développer de nouveaux. Pour cela, les experts font intervenir des coefficients arbitraires (Atherton, 2005 ; Morgan, 2001) issus de leurs observations, lesquelles ne peuvent techniquement pas couvrir exhaustivement tous les cas d'érosion dénombrables sur la zone d'étude pouvant s'étendre sur plusieurs milliers de kilomètres carrés (comme dans le cas de la Nouvelle-Calédonie).

Types de modèles Deux grandes classes de modèles peuvent ainsi être distinguées : les modèles empiriques et les modèles physiques.

Les modèles empiriques sont construits à partir de connaissance experte et d'expérimentations. Le modèle *Universal Soil Loss Equation* (USLE) de Wischmeier et Smith (1978) et le modèle proposé par Atherton (2005) en sont des exemples typiques. USLE est l'une

2. Société exploitant, entre autres, la mine de Goro
3. Convention relative aux zones humides d'importance internationale, particulièrement comme habitats des oiseaux d'eau, signée en 1971
4. Gaba (2013) sous license CC BY-SA 3.0

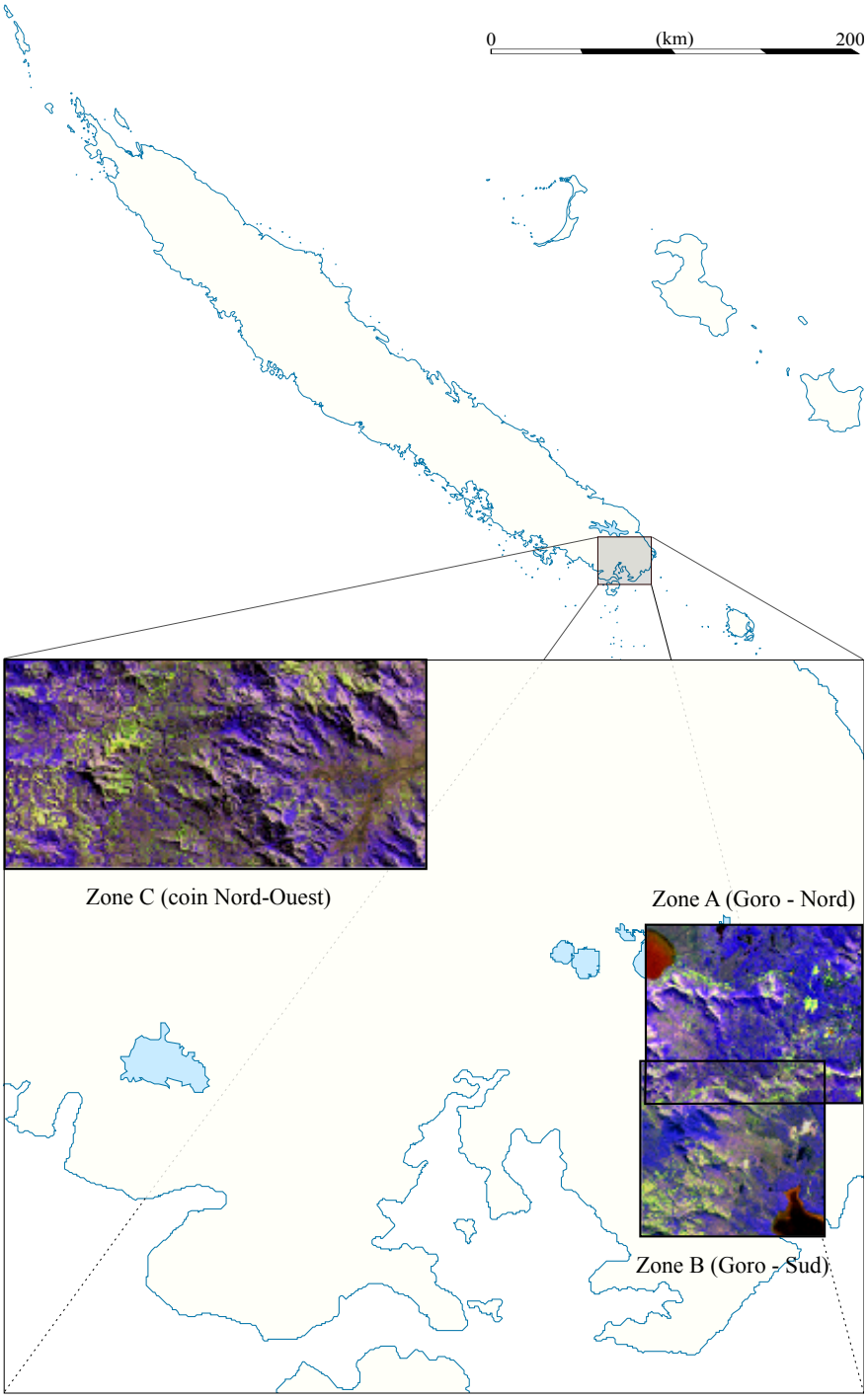


FIGURE 4.1 – Zone d'étude : Sud de la Nouvelle-Calédonie [4]

des méthodes les plus répandues pour estimer l'érosion hydrique. Cette équation, développée pour les milieux cultivés aux USA, prend en compte l'érosion pluviale, la topographie, la couverture végétale et la protection du sol . Ce modèle empirique de l'érosion du sol a été revu (*Revised Universal Soil Loss Equation* (RUSLE) de Renard *et al.* (1997)) pour intégrer de nouveaux paramètres. Le modèle d'Atherton a été développé pour modéliser l'érosion des sols aux îles Fidji, dont les conditions climatiques et géologiques sont proches de celles de la Nouvelle-Calédonie. Ce modèle s'appuie sur deux indices : le *Relative Erosion Prediction* (REP) et le *Watershed Development Index* (WDI). Le REP est une mesure relative de prédiction de l'érosion basée sur la pente, l'occupation du sol, les précipitations absolues ou saisonnières et l'érodibilité des sols. Le WDI représente le degré d'impact des infrastructures (densité des routes, degré de déforestation, et nombre de cours d'eau traversés par les routes par km^2) sur le bassin versant. Le risque total est obtenu en additionnant les deux mesures. En fonction de la valeur obtenue, le risque d'érosion pour le sol en question sera classé faible, moyen ou fort (la classification a été définie a priori par l'expert).

Les modèles physiques sont des modèles quantitatifs fondés sur des propriétés physiques et calibrés à partir des données expérimentales. Par exemple, les modèles *Water Erosion Prediction Project* (WEPP) de Lane et Nearing (1989) et *Revised Morgan-Morgan Finley* (RMMF) de Morgan (2001) sont basés chacun sur plusieurs modèles physiques. Le ruissellement et les pertes de sol peuvent aussi être évalués à partir d'un modèle nommé WEPP (Bhuyan et al. 2002). Ce modèle, développé par Flanagan *et al.* (2007), se base sur de multiples paramètres dont les notions de base de l'infiltration, la surface de ruissellement, la croissance des plantes, les résidus de décomposition, l'hydraulique, le labourage, la consolidation du sol ainsi que les mécanismes d'érosion (Nearing *et al.*, 1989). Le modèle RMMF déjà présenté au chapitre 3 figure 3.11 page 74 divise le processus d'érosion en deux phases : détachement par gouttes de pluie et détachement par ruissellement. Chaque phase est liée à un sous-modèle physique. Les résultats des deux sous-modèles sont ensuite additionnés pour estimer la perte en sol annuelle. Comme précédemment, des classes de valeurs sont ensuite associées à différents niveaux de risque par les experts.

1.3 Description des données

Les scenarii de fouille de données découlent évidemment des données dont on dispose. Dans cette thèse, il est acquis que nous disposons de données spatiales et temporelles issues d'images satellite. Dans cette étude de cas, nous disposons aussi de données complémentaires telles qu'un Modèle Numérique de Terrain (MNT), une cartographie de la lithologie, ainsi qu'une cartographie de l'occupation des sols pour l'année 2008. Les images satellites ont été prises en 1999, 2001, 2002, 2005, 2006, 2008 et 2009.

Les images satellites proviennent du satellite SPOT5 et contiennent les valeurs radiométriques Rouge, Vert, NIR (*Near Infrared*), et MIR (*Medium Infrared*). De ces valeurs peuvent être calculés plusieurs indices, tels que le NDVI (*Normalized Difference Vegetation Index*), le Redness, ou le Brightness pour les plus connus. Le NDVI, calculé en fonction du Rouge et du NIR ($\text{NDVI}=\frac{\text{NIR}-\text{Rouge}}{\text{NIR}+\text{Rouge}}$) est couramment utilisé en télédétection pour identifier les zones végétalisées : plus grande sera la valeur, plus la zone sera supposée contenir une végétation dense et verte. Le Redness donne un indice de rougeur ($\text{Redness}=\frac{\text{Rouge}^2}{\text{Vert}^3}$), utile pour repérer

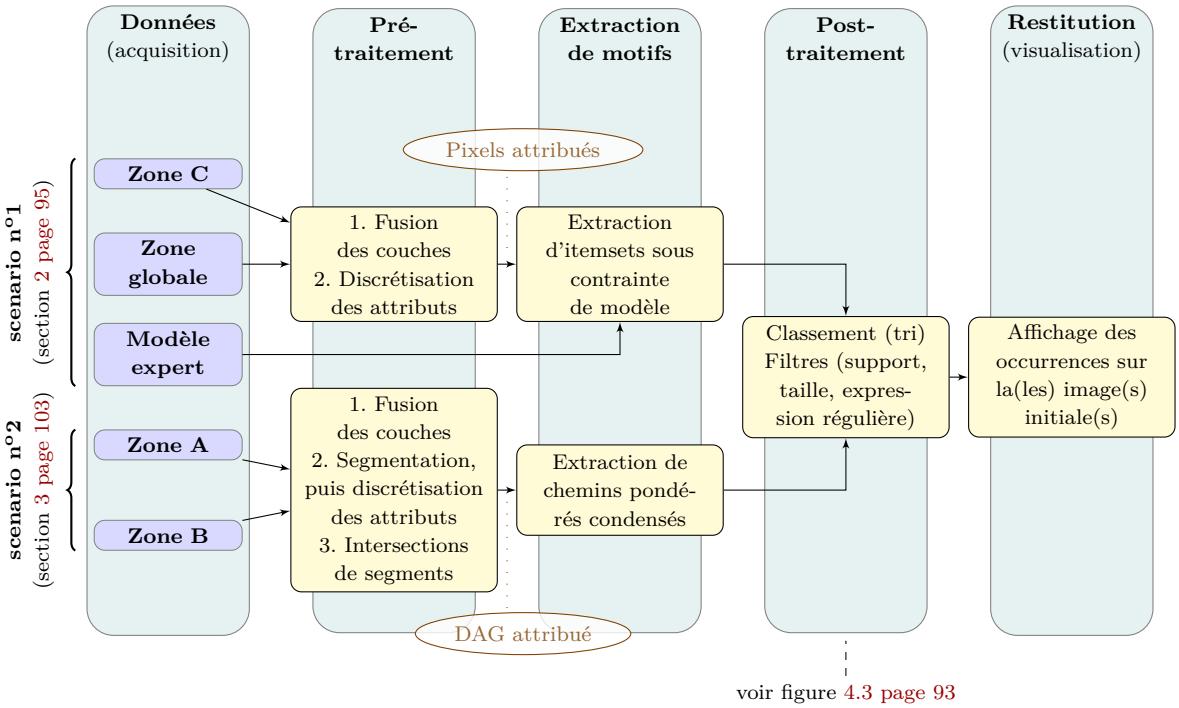


FIGURE 4.2 – Description des deux scenarii de fouille de données.

les zones érodées en Nouvelle-Calédonie qui laissent apparaître les hématites du sol (composées d'oxyde de fer donnant au sol sa couleur rouge). Pour cette raison, il est souvent utilisé pour détecter des sols nus. Quant au Brightness qui est la moyenne du Rouge et du Vert, il pourra indiquer les zones nuageuses pour ses valeurs les plus hautes.

Ces images SPOT sont d'une résolution de 10 mètres. Un pixel représente donc un carré de 10m de côté. La zone d'étude globale correspond à une image de 8 890 656 pixels, soit environ 889 km^2.

1.4 Processus de fouille de données mis en place

Nous avons mis au point deux scenarii de fouille de données, décrits dans la figure 4.2. Le premier scénario porte sur les données spatiales (zone globale d'étude et zone C). Le second porte sur les données spatio-temporelles acquises pour les zones A et B. Ces deux dernières zones couvrent un espace où l'activité anthropique y est intense (mines, usines, routes). Ces zones seront plus amplement décrites dans la section 3.

Dans le premier scénario, nous utilisons une image satellite dont nous analysons les pixels. Nous utilisons aussi les données additionnelles issues du MNT, du type des sols et de leur occupation. Chaque pixel est ainsi composé de plusieurs couches (c'est-à-dire, comporte plusieurs types de données), à partir desquelles nous générons autant d'attributs. L'ensemble de ces pixels peut ainsi être vu comme une base de données transactionnelle, où chaque transaction représente un pixel. Ainsi, nous recherchons les pixels fréquents présentant des risques forts ou moyennement forts d'érosion, selon un modèle expert.

Dans le deuxième scénario, nous disposons des mêmes données pour plusieurs dates.

Contrairement au premier scénario, nous ne nous intéressons pas individuellement à chaque pixel des images mais à des objets d'études regroupant plusieurs pixels. Pour cela, nous passons donc par une première étape de segmentation. Le calcul des attributs est modifié en conséquence, tel qu'expliqué plus loin. Puis, à partir des différents segments obtenus pour chaque date, nous créons un DAG, dont les sommets représentent les segments attribués. Ainsi, nous recherchons les chemins pondérés fréquents d'attributs dans ce a-DAG.

1.5 Prototypes de visualisation

Pour les deux scenarii, nous avons implémenté un prototype pour pouvoir remettre les différents motifs dans leur contexte, c'est-à-dire faire la correspondance entre un motif et les segments utilisés dans ses occurrences. La figure 4.3 donne une capture d'écran de ce logiciel pour la visualisation de chemins pondérés [5]. Afin de visualiser les occurrences des chemins pondérés à travers le temps, les occurrences des chemins sont coloriées en fonction de leur position temporelle. Ainsi, pour un motif $I_1 \overset{\text{poids}}{\longmapsto} I_2 \overset{\text{poids}}{\longmapsto} I_3 \overset{\text{poids}}{\longmapsto} I_4 \overset{\text{poids}}{\longmapsto} I_5$, tous les sommets des occurrences du motif qui contiennent l'itemset I_1 seront coloriés en rouge (que ces sommets appartiennent à l'image t_1, t_2, ... ou t_{n-1}). De la même façon, les sommets contenant I_2 seront coloriés en bleu (que ces sommets appartiennent à l'image t_2, t_3, ... ou t_{n-2}). Dans le cas où le motif s'avère être de taille 3, la couleur pour les derniers sommets des occurrences sera le vert ; si le motif est de taille 4, la couleur sera l'orange, et finalement si le motif est de taille 5, la couleur sera le mauve. Un chemin pondéré peut être supporté par des occurrences dont les segments correspondants peuvent appartenir à des temps différents. L'utilisation du code de couleurs permet donc de toujours identifier le début d'un chemin par la couleur rouge. Ainsi, une image à temps donné peut comporter des zones rouges ou bleues pour l'affichage d'un même motif : cela signifie que certaines occurrences du chemin commencent à ce temps-ci, et d'autres au temps précédent.

En outre, le logiciel permet de réaliser les étapes de pré-traitement des données à partir de données brutes (c'est-à-dire, avant la fouille de donnée et avant la visualisation, voir plus loin figure 4.11). Le visualiseur permet quelques opérations de post-traitement sur les motifs : tri en fonction du support, tri en fonction de la taille, support maximum, recherche de motif par expression régulière. Par exemple, on peut vouloir se focaliser sur les motifs porteurs de caractéristiques d'érosion. Pour cela, on peut filtrer les motifs avec l'expression régulière « *Redness4* » (puisque un fort Redness traduit la présence d'un sol érodé).

5. La visualisation d'itemsets suit le même principe, mais plus simplement (correspondance entre itemsets et pixels), et ne sera donc pas détaillée ici.

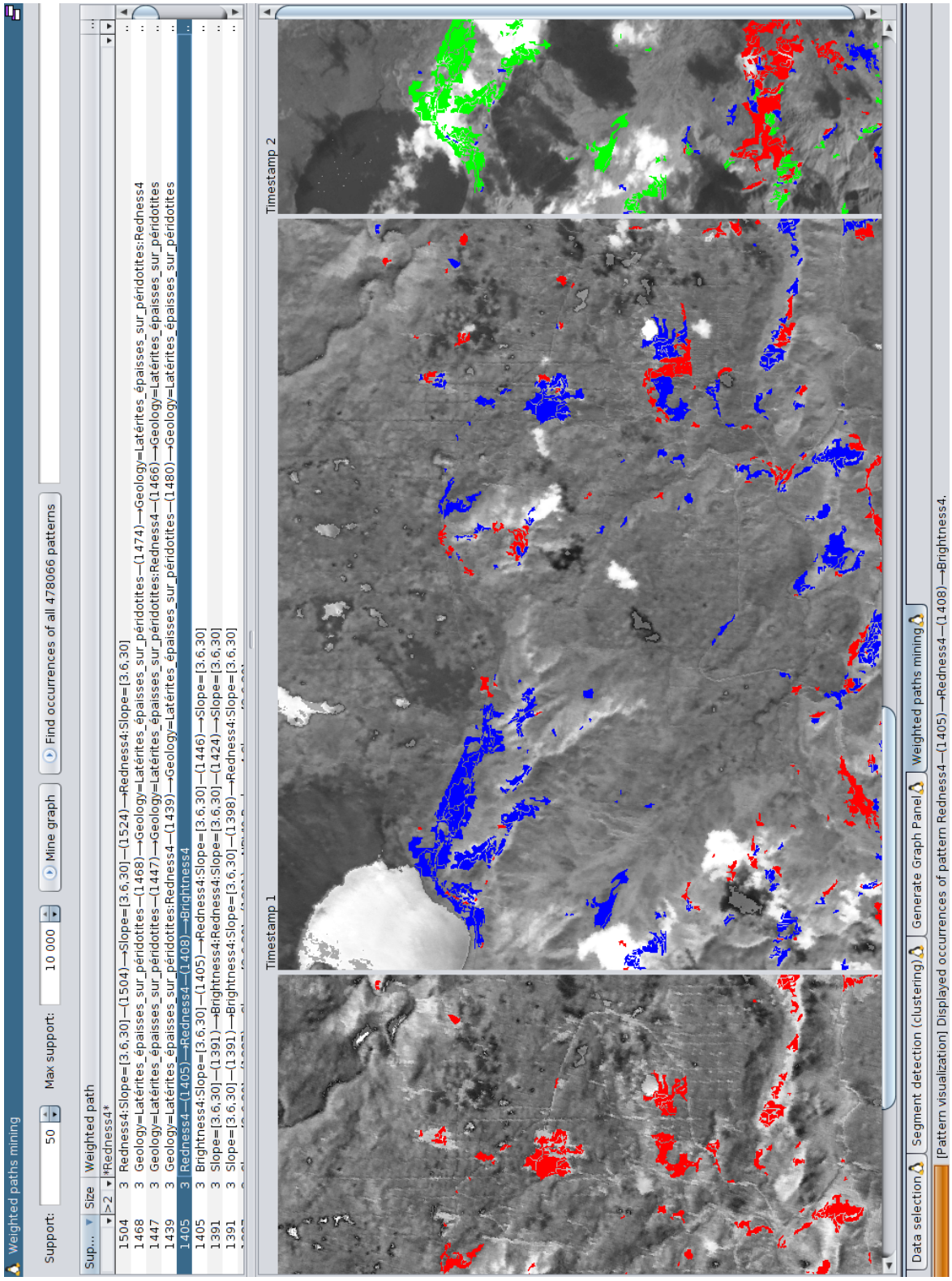


FIGURE 4.3 – Capture d'écran du prototype de visualisation

2. Utilisation de modèles experts pour la fouille de pixels

2.1 Données utilisées

Zone d'étude globale et zone C Nous avons étudié une zone située dans le coin Nord-Ouest de la zone d'étude globale. Cette vaste zone est la zone C dessinée sur la figure 4.1 page 89. Pour l'année 2008, elle ne présente aucun nuage susceptible d'engendrer du bruit. De plus, nous disposons pour cette année d'une vérité terrain concernant l'occupation des sols. Nous savons alors exactement où se trouvent les différents types de sol (forêts, points d'eau, savane, etc.). La zone C couvre 1 142 910 pixels, soit environ 114,3 km^2.

2.2 Modèle d'Atherton

Initialement développé pour estimer le risque d'érosion aux îles Fidji, il a récemment été adapté au contexte calédonien par Paul-Hus (2011). La principale modification a été de prendre en compte l'impact des mines (caractéristique propre à la Nouvelle-Calédonie) à la place de l'impact de l'industrie du bois qui est très présente aux Fidji. Ce modèle calcule 3 indices :

— le REP, qui calcule l'aléa en fonction de paramètres naturels (pente, pluie, type de sol, occupation du sol) ; l'indice additionne les différents risques assignés à ces paramètres (voir figure 4.4)

— le WDI, qui calcule l'aléa en fonction de paramètres anthropiques (routes, mines, ponts, etc.) ; tout comme l'indice précédent, celui-ci est une simple somme

— et finalement le *Composite Threat Index* (CTI), qui est la simple somme du REP et du WDI.

Sur une image à un temps donné, il peut être intéressant de vouloir détecter les zones présentant un fort risque d'érosion d'après ce modèle. Pour cela, on peut représenter un ensemble de pixels et leurs attributs respectifs sous la forme d'itemsets. Le risque d'érosion est calculé d'après le modèle Atherton décrit précédemment. En insérant une contrainte experte $q_{f\geq}$ avec un f_{min} dénotant une forte érosion, on se concentre sur les itemsets (et leurs occurrences : les pixels) présentant une forte érosion. Ce sera également l'occasion d'examiner les autres variables d'environnement (NDVI, Vert, Rouge, NIR), porteuses d'informations additionnelles. Elles permettent de confirmer ou pas les informations du modèle.

2.3 Traitements

L'image a été transformée en base de données transactionnelle représentant les informations sur les pixels. Les attributs sont les propriétés radiométriques des pixels (NDVI, Vert, Rouge, NIR) discrétisées en 10 intervalles, et les variables nécessaires au modèle Atherton (2005) utilisé comme modèle expert pour ces expérimentations (Slope, Geology, LandCover).

Variable	**Valeur**	**Description**
Pente (Slope), en %		
[0 ; 3,5]	0,5	Très faible
[3,6 ; 30]	1	Faible
[31 ; 50]	2	Modérée
[51 ; 60]	3	Forte
Pluviométrie, en mm		Classe d'érosivité
$< 2\ 000$	1	Élevée
[2 001 ; 3 200]	2	Très élevée
$> 3\ 201$	3	Extrême
Type de sol (Classification lithologique Geology)		Classe d'érodibilité
Dunites	1	
Harzburgites	1	
Gabbros pegmatoïdes, amphibolites	1	Faible
Alluvions	1	
⋮		
Cuirasses disloquées et démantelées	2	Modérée
Serpentinites	3	Élevée
Décharges minières non contrôlées et coulées de matériaux	4	
Zones d'exploitation et déblais miniers	4	
Latérites épaisses sur péridotites	4	Sévère
Latérites indifférenciées sur péridotites	4	
Latérites minces sur péridotites	4	
Occupation du sol (LandCover)		
Eau	0	
Mangrove	0	
Tannes	1	
Maquis dense para-forestier	1	
Forêt sur substrat ultramafique	1	
Savane	2	
Forêt sur substrat volcano-sédimentaire	2	
Maquis ligno-herbacé	2	
Végétation arbustive sur substrat volcano-sédimentaire	2	
Végétation éparse	3	
Zone d'habitation	3	
Sol nu	4	

Figure 4.4 – Valeurs expertes pour le modèle Atherton appliqué au contexte de la Nouvelle-Calédonie (Paul-Hus, 2011)

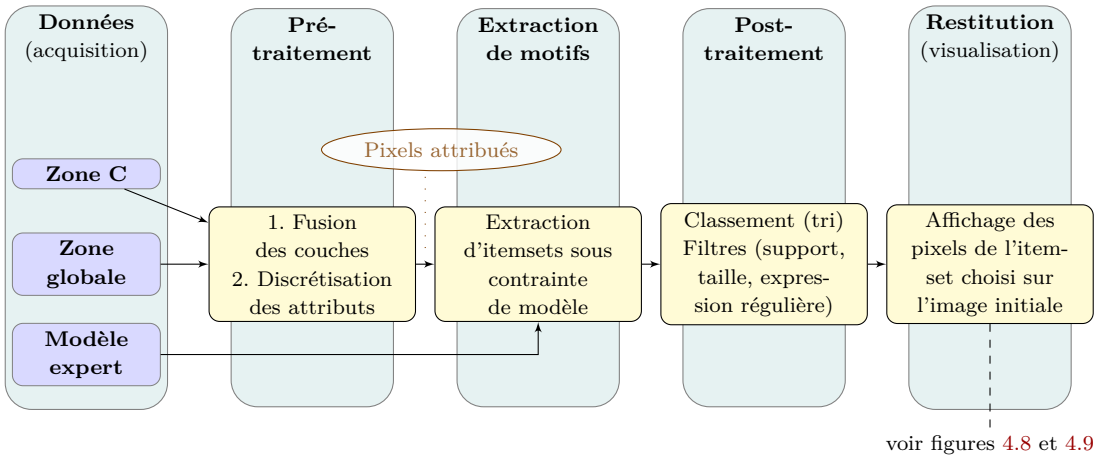


FIGURE 4.5 – Scénario 1.

Au final, nous obtenons une base de données transactionnelle composée de plus de 8 millions de transactions, où chaque transaction est constituée de 7 items. Ce jeu de données contient au total 74 items différents (regroupés selon les 7 attributs/variables cités précédemment). Nous avons exécuté l'algorithme sur une machine de 8 GB de RAM et un processeur Intel® Core™ i5-2400 cadencé à 3,20 GHz.

2.4 Résultats quantitatifs

La figure 4.6 donne les performances en terme de temps d'exécution ainsi que le nombre de solutions trouvées pour différents seuils de fréquence en abscisse, et pour différentes contraintes du modèle.

Nous avons répété l'expérience sur la zone C (coin Nord Ouest de la zone globale), ne faisant qu'un million de pixels. Ces résultats sont reportés sur la figure 4.7.

On aurait pu penser que les calculs nécessaires à l'application de la contrainte s'avéreraient trop coûteux pour améliorer significativement les performances de l'algorithme. Pourtant, nous pouvons constater que l'utilisation des contraintes du modèle réduit grandement (par plusieurs ordres de grandeur) le nombre de solutions, et accélère ainsi la fouille. En effet, même avec la faible contrainte $f \geq 3$, on élague beaucoup de motifs. De façon plus générale, sans notre contrainte basée sur le modèle ATHERTON, le nombre de solutions peut excéder 1 000 itemsets pour une fréquence minimale de 10%. Avec la contrainte basée sur ce modèle, le nombre de solutions ne dépasse jamais 10. De la même manière, le temps d'exécution peut atteindre 6 000 secondes sans notre contrainte, alors qu'il ne dépasse pas 2 000 secondes avec la contrainte $f \geq 15$ (en conservant la même contrainte de fréquence).

2.5 Résultats qualitatifs

Grâce à la réduction du nombre de solutions, nous avons pu plus facilement recueillir le seul itemset présentant une érosion particulièrement forte ($f \geq 15$) : {Geology = Serpentinites, LandCover = Sol nu sur substrat volcano-sédimentaire, Slope = [61 ; 100], Rouge =]16 ; 32], Vert =]14,2 ; 28,4], NIR = [0,0 ; 36,1], NDVI =]−0,071 ; 0,115], MIR = [0,0 ; 24,5]}.

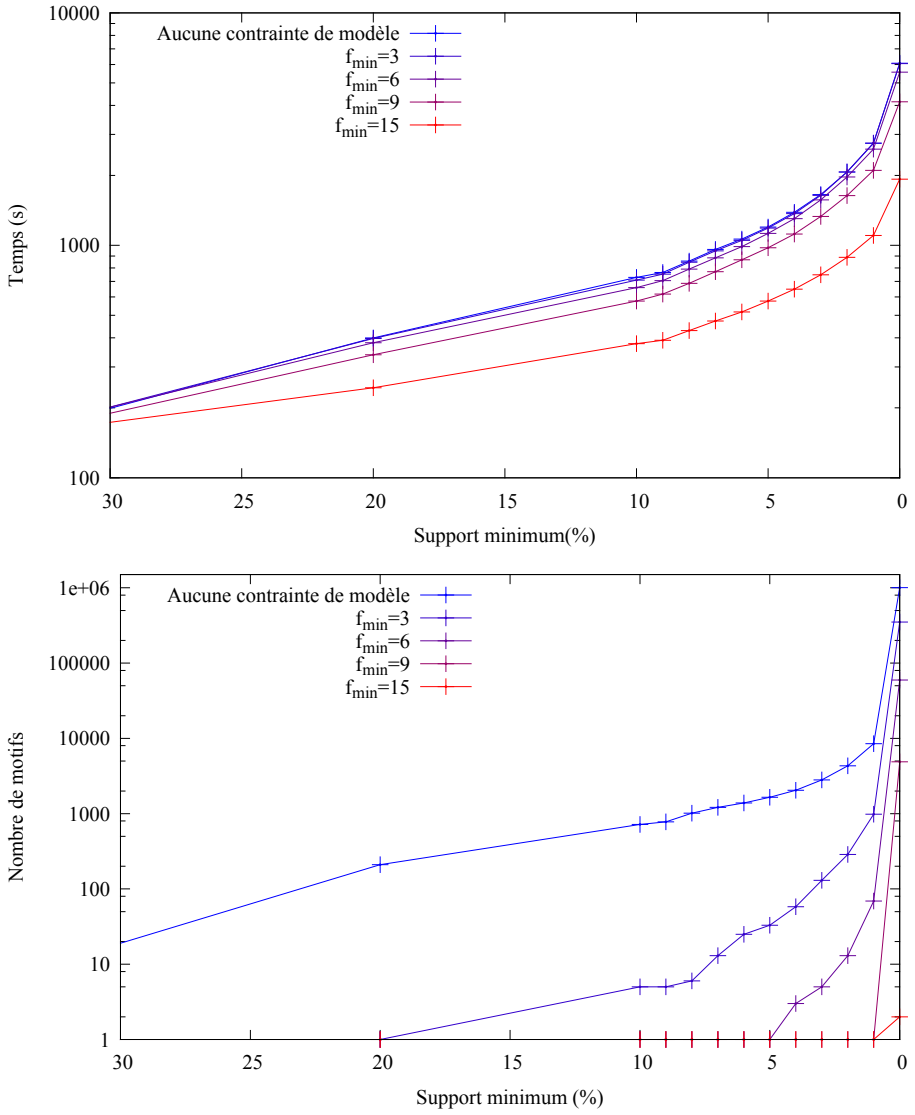


Figure 4.6 – Performances pour le jeu de données de 8 millions de pixels

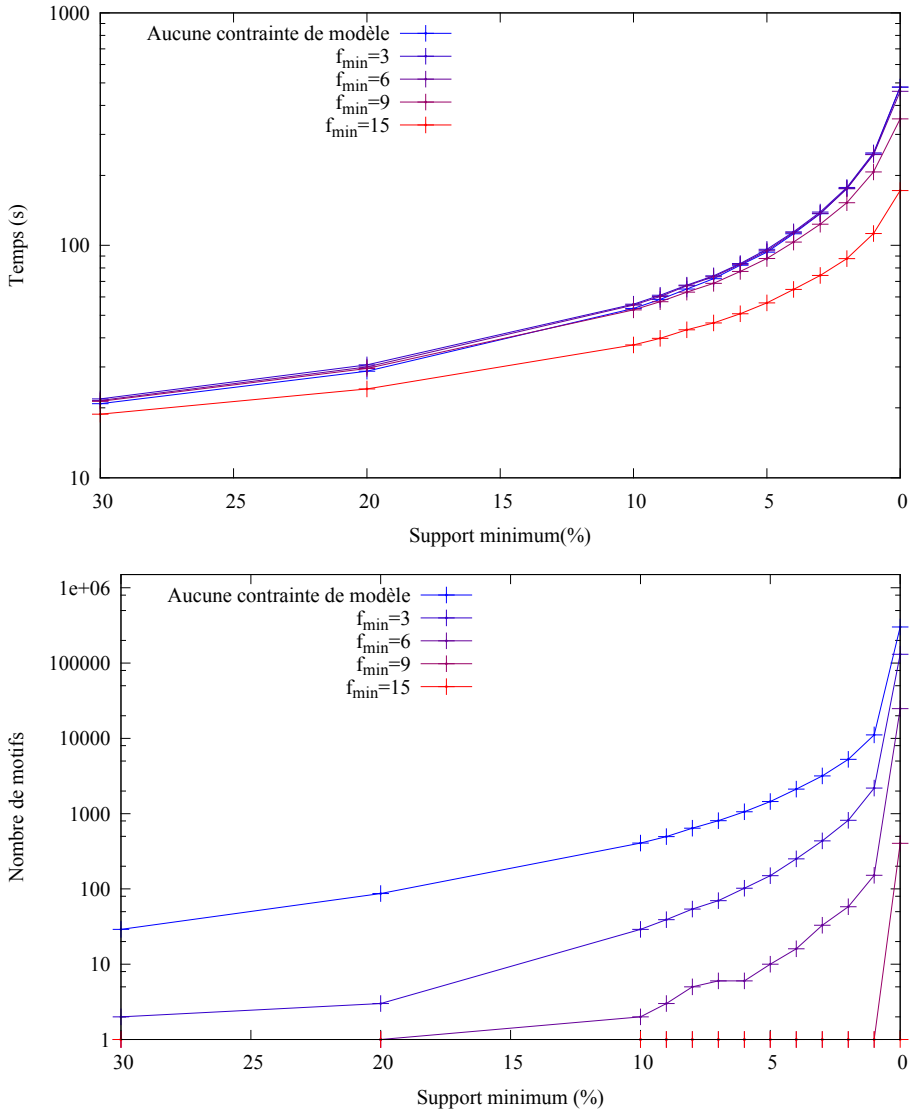


FIGURE 4.7 – Performances pour le jeu de données restreint à la zone C (1 million de pixels)

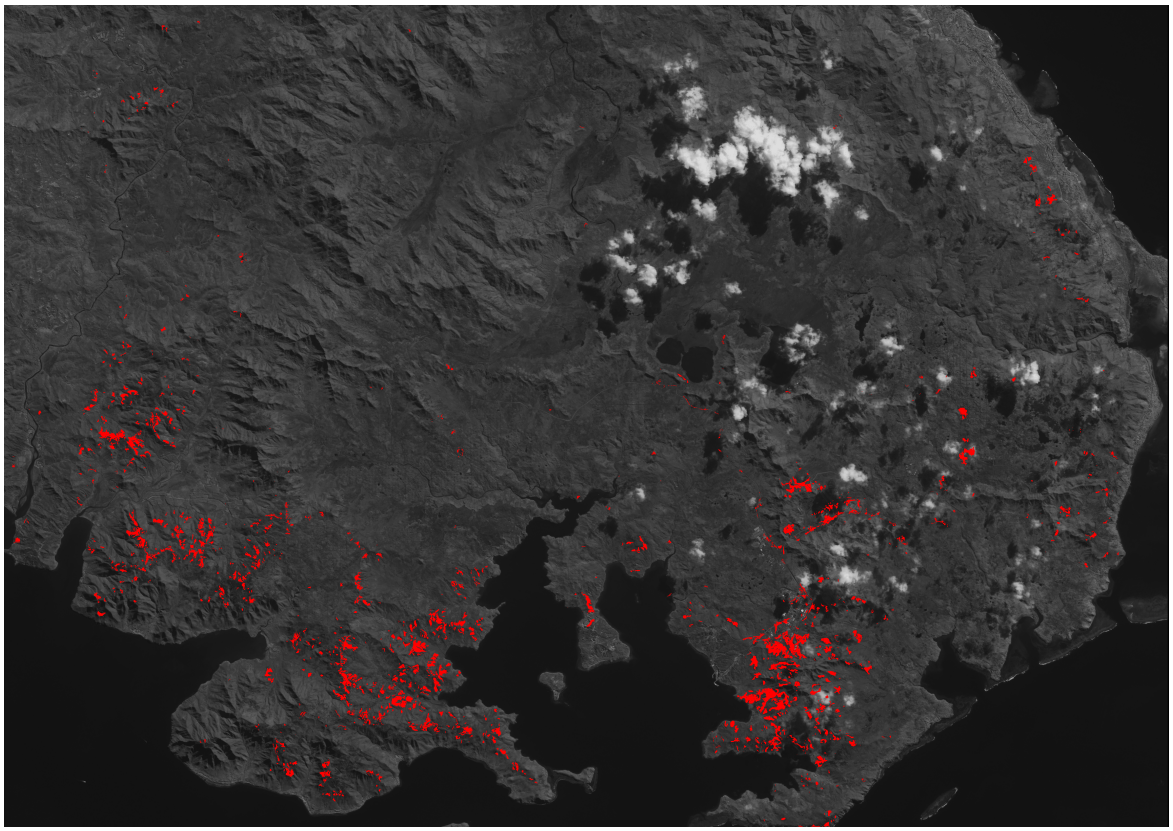
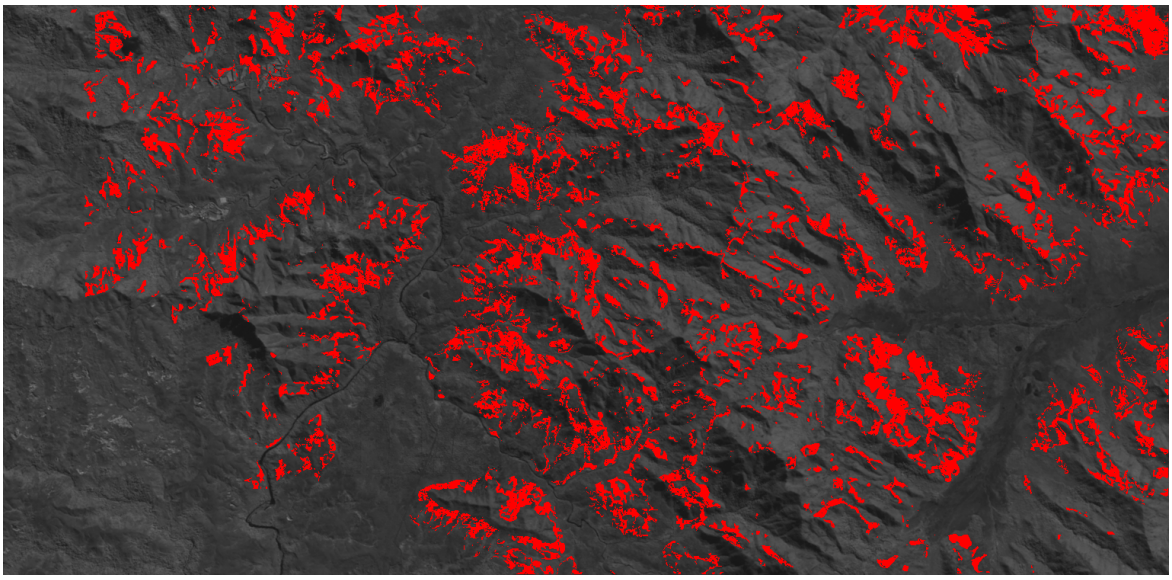


Figure 4.8 – Pixels de la zone d'étude globale marqués par une érosion forte ou moyennement forte ($f \geq 9$).

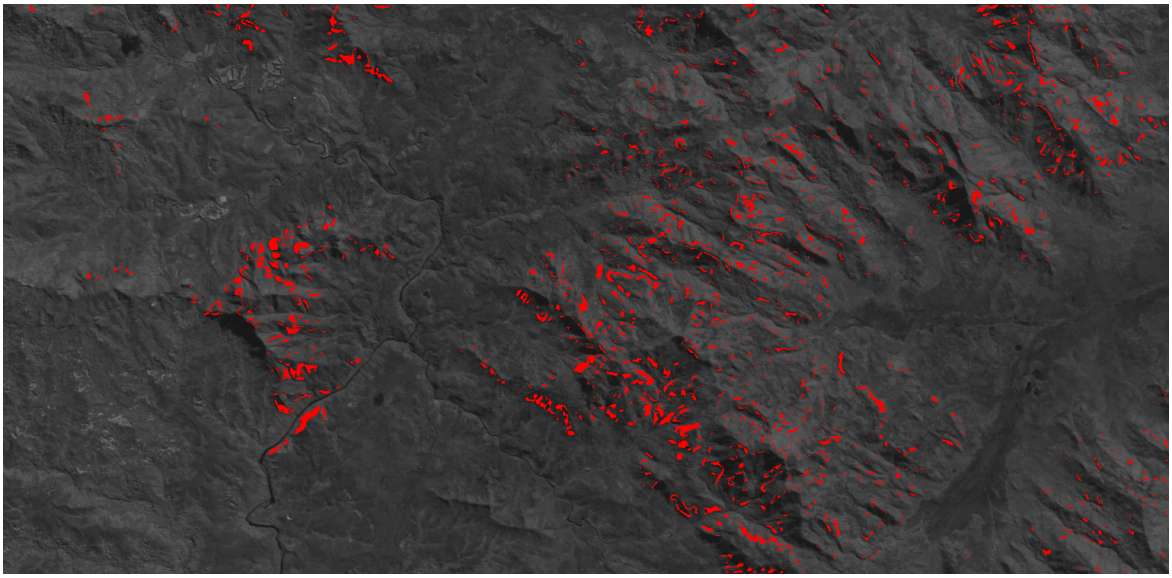
Les variables radiométriques (qui ne font pas partie du modèle) précisent que nous sommes en présence de faible indice de NDVI (dépendant directement du Rouge et du NIR), traduisant une absence de végétation, confirmant à son tour la pertinence du modèle.

Lorsque l'on baisse ce seuil à une érosion moyennement forte ($f \geq 9$, voir figure 4.8), on obtient le motif suivant à partir d'une fréquence *support* $\geq 1\%$: {Geology = Latérites épaisses sur péridotites, LandCover = Sol nu sur substrat ultramafique }. Les items additionnels que l'on obtient, avec une fréquence plus faible (*support* $\geq 0{,}5\%$) sont NDVI =]0,011 5 ; 0,301], Redness =]0,014 ; 0,021], Brightness =]80,232 ; 106,977] et NIR =]36,1 ; 72,2]. En clair, une forte brillance, un fort indice de rougeur et une faible végétation sont des indicateurs de cette érosion passablement forte.

Quant à la zone C, qui ne présente donc que très peu de pixels soumis à forte érosion, nous avons pris comme contrainte de modèle minimale $f \geq 6$, ne gardant que les pixels soumis à une érosion plus modérée. La figure 4.9a montre les pixels du motif le plus fréquent sur cette zone. Ces pixels sont tous situés sur des pentes faibles ou très faibles. Lorsque l'item « LandCover = Maquis ligno-herbacé » est retiré, on s'aperçoit que les itemsets les plus fréquents concernent les pixels où la pente est modérée (4.9b).



(a) Pixels supportant l'itemset {Geology = Latérites épaisses sur péridotites, LandCover = Maquis ligno-herbacé, NDVI =]0,491 ; 0,608]} (*support* = 8%).



(b) Pixels présentant une pente modérée {Geology = Latérites épaisses sur péridotites, Slope = [31 ; 50]} (*support* = 2%)..

Figure 4.9 – Pixels de la zone C marqués par une érosion modérée ($f \geq 6$).

3. Suivi d'objets d'intérêt sur une série temporelle d'images

Sur plusieurs pas de temps, il est intéressant de découvrir les évolutions de zones. L'étude portant sur l'érosion, nous nous attarderons sur les zones supposées érodées. La section suivante détaille avec plus de précisions la méthode et les techniques utilisées pour construire un jeu de données à partir des données brutes.

3.1 Détails des données

Zones A et B Nous disposons de plusieurs images satellite de résolution 10 mètres datant de 1999, 2001 (résolution 20m), 2002, 2005, 2006, 2008 et 2009. La difficulté d'obtenir des séries d'images idéales (à savoir, sans nuages, de même résolution, suffisamment espacées dans le temps) nous a contraints à nous dispenser de l'image datant de 2001 [6]. Nous avons focalisé notre étude sur deux zones, respectivement de taille 794 × 660 pixels ($\approx$ 52,5 km^2) appelée « zone A », et 669 × 626 pixels ($\approx$ 41,9 km^2) appelée « zone B ». Elles sont représentées figure 4.1. Toutes deux couvrent les variétés de régions observables dans le sud de la Nouvelle-Calédonie : points d'eau douce ou marine, activité anthropiques (mines, usines et pistes), relief, bassins versants, plaines, marécages, forêts. La zone A se trouve au Nord, légèrement à l'Est de la zone B. Ces deux zones englobent la mine dite « de Goro ». De plus, les précipitations y sont fortes et régulières.

3.2 Traitements

Ce second scénario est décrit dans la figure 4.10. Nous expliquerons en premier lieu les étapes de pré-traitement des données pour les transformer en un a-DAG.

3.3 Construction de la base de données sous forme d'un DAG attribué

3.3.1 Segmentation spatiale

Afin de détecter les objets d'études, il est nécessaire d'effectuer une étape de segmentation. La segmentation d'image est une technique qui regroupe un ensemble de pixels selon un critère d'homogénéité. En l'occurrence, il s'agit de détecter les (portions de) rivières, lacs, plaines, montagnes présentant les mêmes caractéristiques. On s'appuie pour cela sur les diverses variables que l'on possède pour chaque pixel, ce qui inclut les variables radiométriques issues des images satellites. On peut aussi utiliser les données d'un MNT donnant des indications de pente (Slope) ou une vérité terrain comme le type des sols (Geology) ou bien leur occupation (LandCover).

6. des images de résolutions différentes pourraient cependant être utilisées *via* des pré-traitements

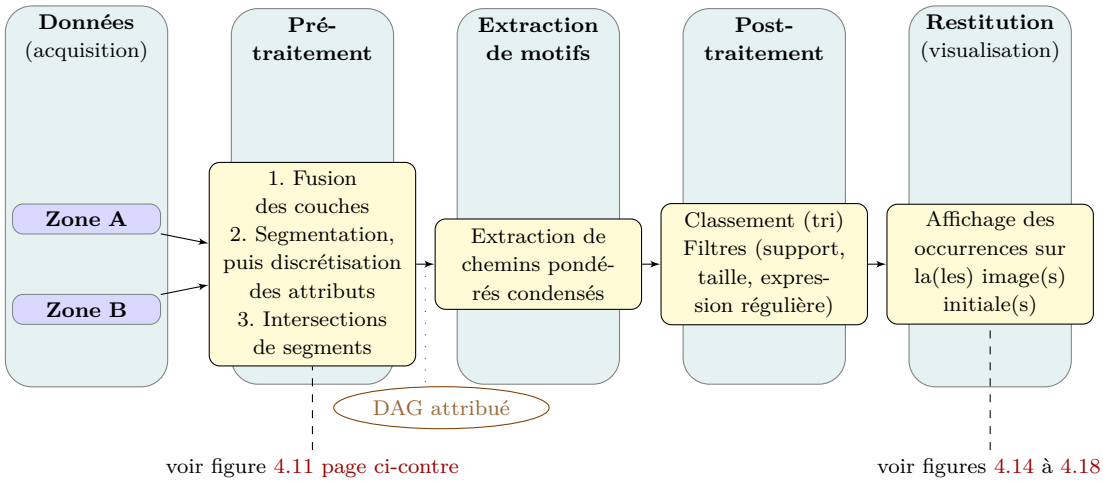


Figure 4.10 – Description des deux scenarii de fouille de données.

Lorsque les attributs en question sont déjà à valeurs catégorielles (Geology et LandCover), la segmentation est triviale : la classe d'un pixel pour cet attribut est la valeur de cet attribut. Dans le cas contraire, la segmentation peut s'avérer plus complexe :

— Soit on discrétise arbitrairement les valeurs de l'attribut (par exemple, en fonction d'un modèle expert, comme cela a été fait pour la pente Slope),

— Soit on applique une technique de segmentation non supervisée, telle que la méthode du *Watershed* (Beucher et Lantuéjoul, 1979) qui est très utilisée en segmentation d'images.

La méthode de segmentation *watershed* ou segmentation par ligne de partage des eaux considère une bande d'image (usuellement une image à niveaux de gris) comme un relief topographique et en simule l'inondation. La méthode utilise deux paramètres *Level* et *Threshold*. *Level* correspond au niveau d'eau qui va venir inonder le relief. *Threshold* est un seuil souvent fixé à 1% du *Level*, qui sera la valeur utilisée pour baisser le niveau d'eau à chaque itération de l'algorithme *watershed*. Pour nos séries d'image, nous avons utilisé une valeur de *Level* de 0,7.

Dans ce travail, nous avons appliqué une segmentation pour chaque couche (attribut), indépendamment l'une de l'autre. Un segment d'une couche peut donc recouvrir un segment d'une autre couche. Sur l'image totale, pour s'assurer que les pixels d'un même segment partagent bien les mêmes attributs, il faut donc redécouper les segments, c'est-à-dire calculer leurs intersections. Ces intersections nous donneront alors tous les segments d'une image, tels que montrés figures 4.13a et 4.13b page 106. Ce processus de segmentation est illustré par la figure 4.12.

Une fois que l'on dispose d'un ensemble de segments (c'est-à-dire nos objets d'étude) pour chaque image, nous construisons le a-DAG associé, comme décrit dans l'état de l'art (chapitre 2, figure 2.9 page 35).

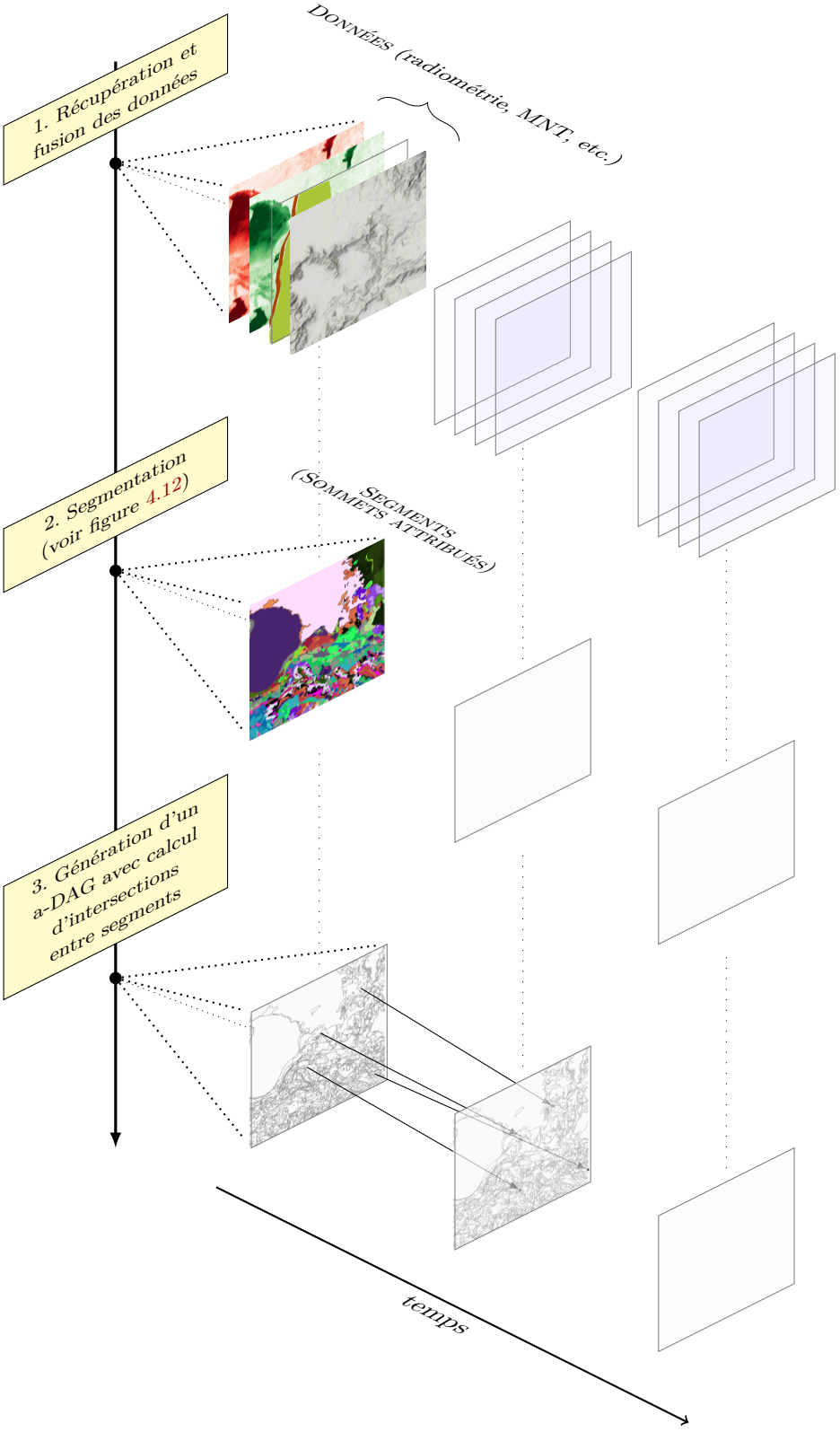


Figure 4.11 – Séquence de traitement des données d'origine vers un a-DAG.

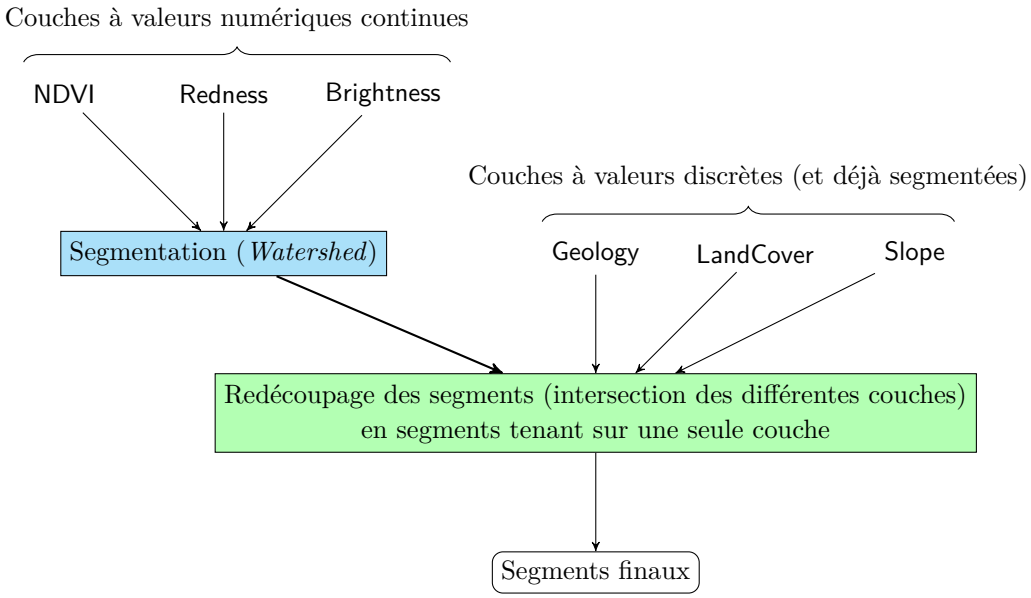
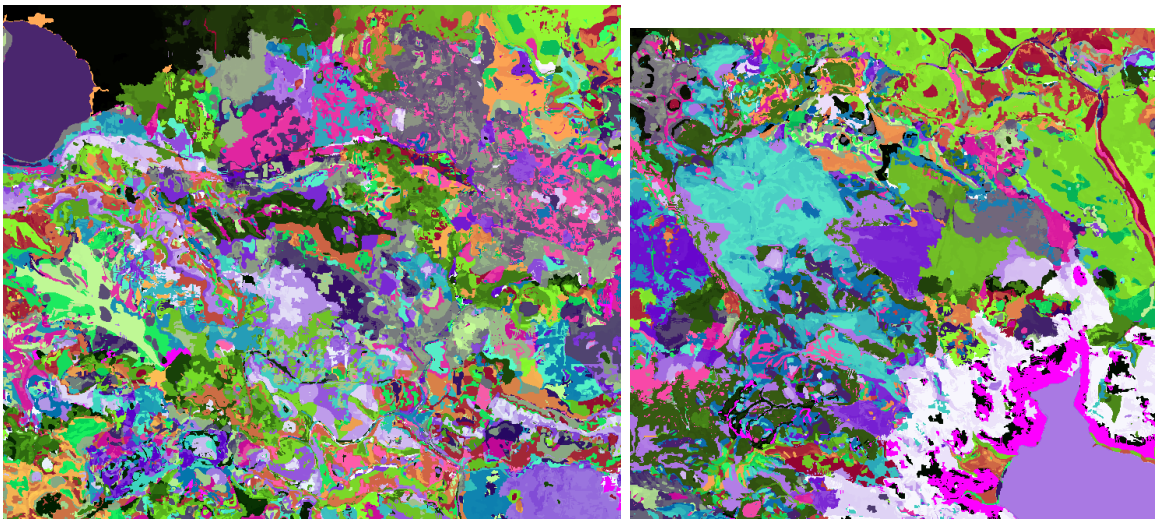


Figure 4.12 – Séquence de traitement des diverses couches de données pour la segmentation



(a) Segments de la zone A (b) Segments de la zone B

Figure 4.13 – Segments des zones d'étude A et B, situées au Sud de la Grande-Terre (voir carte de la figure 4.1). Les couleurs n'ont pas de signification, et ne servent qu'à différencier les segments.

3.3.2 Attribution des sommets

Pour chaque segment, nous faisons correspondre un ensemble d'attributs, issus des variables utilisées pour la segmentation. Lorsque ces attributs sont catégoriels (Geology, LandCover, Slope), la correspondance est directe, car il ne peut y avoir qu'une seule valeur de ces attributs par segment. Ce n'est pas le cas pour les variables à valeurs continues, c'est-à-dire les variables radiométriques NDVI, Redness, et Brightness : un segment regroupe un ensemble de pixels dont les valeurs bandes sont certes proches, mais pas nécessairement identiques. De plus, même si l'on décide, pour une bande/variable donnée, de prendre la moyenne de ces valeurs, on peut éventuellement se retrouver avec autant de valeurs différentes qu'il n'y a de segments dans le jeu de données. En outre, les valeurs de bandes peuvent être sensiblement différentes d'une image satellite à une autre. Il est donc nécessaire de normaliser ces ensembles de valeurs afin de les rendre comparables.

Pour régler ces problèmes, nous discrétisons ces valeurs : pour chaque image, nous scindons en 5 parties distinctes chaque bande radiométrique. Nous créons ainsi 5 intervalles de valeurs différents par bande radiométrique et par image. Pour chaque bande, l'intervalle contenant les valeurs les plus faibles d'une image donnée correspondra à l'intervalle contenant les valeurs les plus faibles d'une autre image. Les bornes de cet intervalle nous intéressent donc peu. Seul nous importe son rang parmi les autres intervalles. Finalement, nous obtenons les attributs NDVI0, NDVI1, ..., NDVI4, Redness0 .. Redness4, et Brightness0 .. Brightness4. L'attribut NDVI0 signifie « valeurs de NDVI très faible », NDVI4 signifie « valeurs de NDVI très fort », et ainsi de suite. Grâce à cette discrétisation, on peut éliminer des problèmes de calibration des images dus aux capteurs, à la luminosité et autres aléas lors des différentes prises de vue.

Finalement, pour une bande donnée, l'attribut devant être assigné à un segment est celui représentant l'intervalle qui contient la valeur moyenne (sur l'ensemble des pixels du segment) de cette bande. Par exemple, pour un segment contenant 3 pixels dont les valeurs de NDVI sont respectivement 1, 2 et 3,3 (moyenne=2,1) et pour une discrétisation NDVI0 = [−1 ; 1,2], NDVI1 = [1,3 ; 4], ..., NDVI4 = [9 ; 12], on choisira l'attribut NDVI1 [7].

3.3.3 Création des arêtes

Nous devons alors estimer si un segment S d'un temps t_i peut influencer la présence d'un autre segment S' au temps suivant t_{i+1}. Comme expliqué dans l'état de l'art, nous considérons qu'un segment peut en influencer un autre si et seulement si le premier se trouve dans le voisinage du deuxième. La distance spatiale entre deux segments sera jugé suffisamment courte s'ils s'intersectent à plus de $\mathcal{T}\%$ de leurs surfaces respectives. Par exemple, en prenant $\mathcal{T} = 10\%$, un segment S d'un temps t_i sera considéré voisin de S' d'un temps t_{i+1} si au moins 10% des pixels de S sont dans S'. Cette influence peut être généralisée par une relation spatiale quelconque. Par exemple, l'expert peut aussi estimer qu'il y a une influence si les objets de deux temps consécutifs se situent à une distance donnée.

7. exemple donné à titre indicatif ne représentant pas nécessairement le vrai échantillon de valeurs

3.4 Motifs fréquents de la zone de Goro - Nord (zone A)

Parmi les motifs trouvés, nous nous sommes focalisés sur ceux exprimant éventuellement un phénomène d'érosion ; ces motifs présentent donc une augmentation de l'indice Redness.

Dans la partie Nord de la mine de Goro, il est intéressant de voir que la majorité des zones fortement érodées (Redness4) sont situées en pente faible. La figure 4.14 montre les zones supportant le motif $\mathsf{Redness4}, \mathsf{Slope} = [3{,}6\,;\,30] \overset{246}{\longmapsto} \mathsf{Redness4}, \mathsf{Slope} = [3{,}6\,;\,30] \overset{244}{\longmapsto} \mathsf{Redness4}, \mathsf{Slope} = [3{,}6\,;\,30] \overset{252}{\longmapsto} \mathsf{Redness4}, \mathsf{Slope} = [3{,}6\,;\,30] \overset{259}{\longmapsto} \mathsf{Redness4}, \mathsf{Slope} = [3{,}6\,;\,30]$. Son support 244 représente en effet $\approx 69\%$ du support 356 du motif $\mathsf{Redness4} \overset{358}{\longmapsto} \mathsf{Redness4} \overset{356}{\longmapsto} \mathsf{Redness4} \overset{362}{\longmapsto} \mathsf{Redness4} \overset{373}{\longmapsto} \mathsf{Redness4}$. Néanmoins, ces occurrences se situent en partie au pied de massifs plus pentus. La présence d'un fort indice de Redness à ces endroits peut alors traduire le fait que ces régions forment des zones de dépôt pour la matière provenant de surfaces en amont.

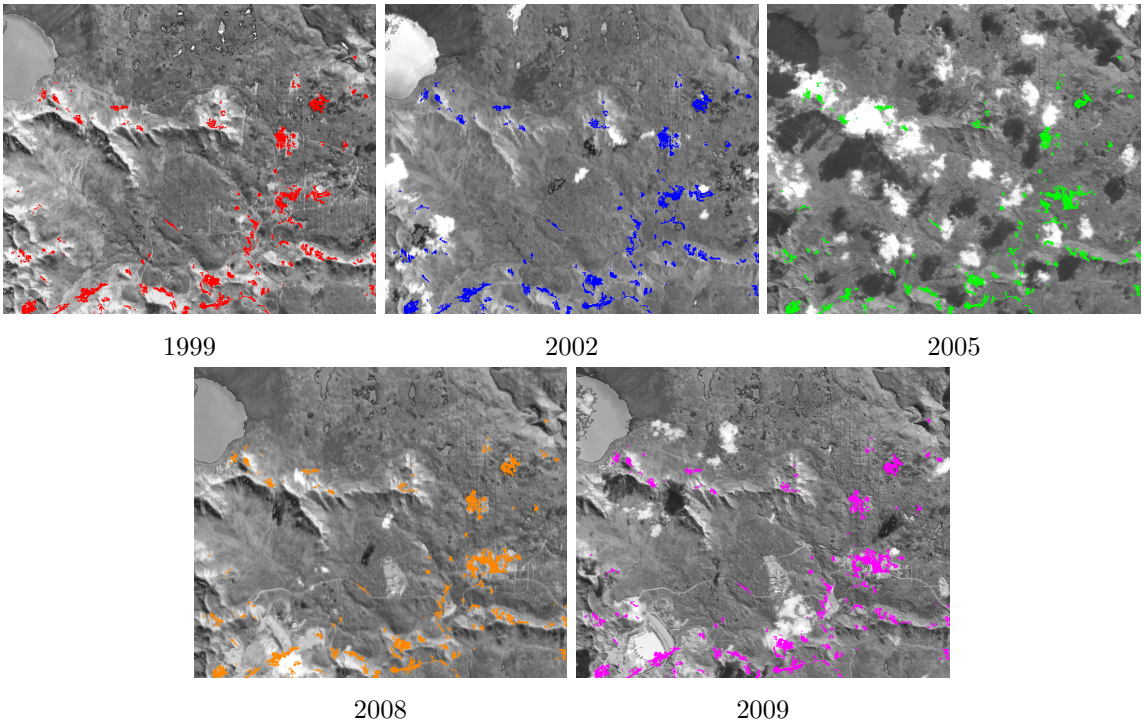


Figure 4.14 – Zone A, $\mathsf{Redness4}, \mathsf{Slope} = [3{,}6\,;\,30] \overset{246}{\longmapsto} \mathsf{Redness4}, \mathsf{Slope} = [3{,}6\,;\,30] \overset{244}{\longmapsto} \mathsf{Redness4}, \mathsf{Slope} = [3{,}6\,;\,30] \overset{252}{\longmapsto} \mathsf{Redness4}, \mathsf{Slope} = [3{,}6\,;\,30] \overset{259}{\longmapsto} \mathsf{Redness4}, \mathsf{Slope} = [3{,}6\,;\,30]$ Suivi de l'évolution des zones qui étaient érodées en 1999.

Parmi les régions pouvant indiquer une revégétalisation (c'est-à-dire, le processus inverse de l'érosion) illustrées sur la figure 4.15, nous pouvons nous apercevoir que seuls des endroits très ponctuels sont concernés (peu de segments, faible surface). De plus, une partie de ces régions sont des petits lacs (régions dans le contour β, image de 2009) ; l'augmentation de NDVI peut alors dénoter une simple prolifération d'algues à la surface. D'autres régions (contour γ) sont en fait des pistes qui sont de moins en moins utilisées. Quand des véhicules empruntent ces chemins, leurs roues soulèvent de la terre et brassent les particules fines ; ces

particules fines sont plus brillantes et plus rouges, donc avec un indice de NDVI plus faible que celui des particules plus grossières qui sont les seules à rester après des averses. Ainsi, au cours du temps, quand un chemin n'est plus utilisé, les particules à sa surface changent de nature au gré des averses, et il est normal d'observer une augmentation de NDVI. De plus, l'espace devient libre à une recolonisation de la végétation. On peut d'ailleurs observer une (très faible) pousse de végétation sur les régions du contour γ. Quant à la région du contour α, elle représente une zone forestière ; l'augmentation de NDVI observée reste à déterminer : il peut s'agir soit d'une densification grâce à de nouvelles pousses, soit grâce à augmentation de l'envergure des arbres. Dans ce dernier cas, il est étonnant que les autres forêts de l'image ne présentent pas elles aussi une telle augmentation d'envergure.

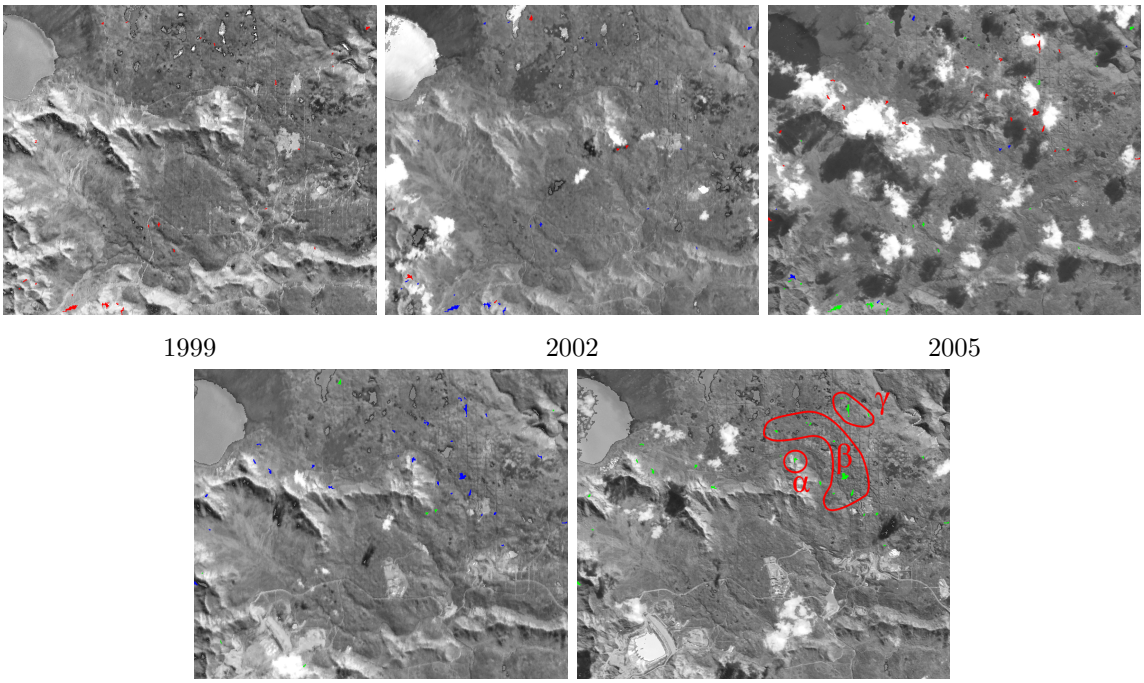


Figure 4.15 – Zone A, NDVI2 $\overset{57}{\rightarrowtail}$ NDVI3 $\overset{58}{\rightarrowtail}$ NDVI4 Suivi de l'évolution des zones subissant une éventuelle végétalisation.

3.5 Motifs fréquents de la zone de Goro - Sud (zone B)

Dans la partie sud de la mine de Goro, on peut déjà observer que les régions étant initialement (en 1999) érodées ont tendance à rester stables, c'est-à-dire toujours autant érodées (figure 4.16, motif Redness4 $\overset{258}{\rightarrowtail}$ Redness4 $\overset{246}{\rightarrowtail}$ Redness4 $\overset{217}{\rightarrowtail}$ Redness4 $\overset{219}{\rightarrowtail}$ Redness4). Cela n'a rien de surprenant, d'autant plus que les régions concernées se concentrent sur et autour de la mine (au Sud Ouest) et du bassin de rétention visible à partir de 2008 (au Nord Ouest). Cependant, il est à noter que certaines régions gagnent en surface dans le temps. Parmi ces régions, on peut distinguer les crêtes des reliefs entourant la mine (ellipse α, image de 2009) ainsi que les routes et chemins (ellipses β). Précisons que la surface d'un segment est une information qui l'on aurait pu ajouter sous forme d'attribut sur le sommet correspondant dans le a-DAG.

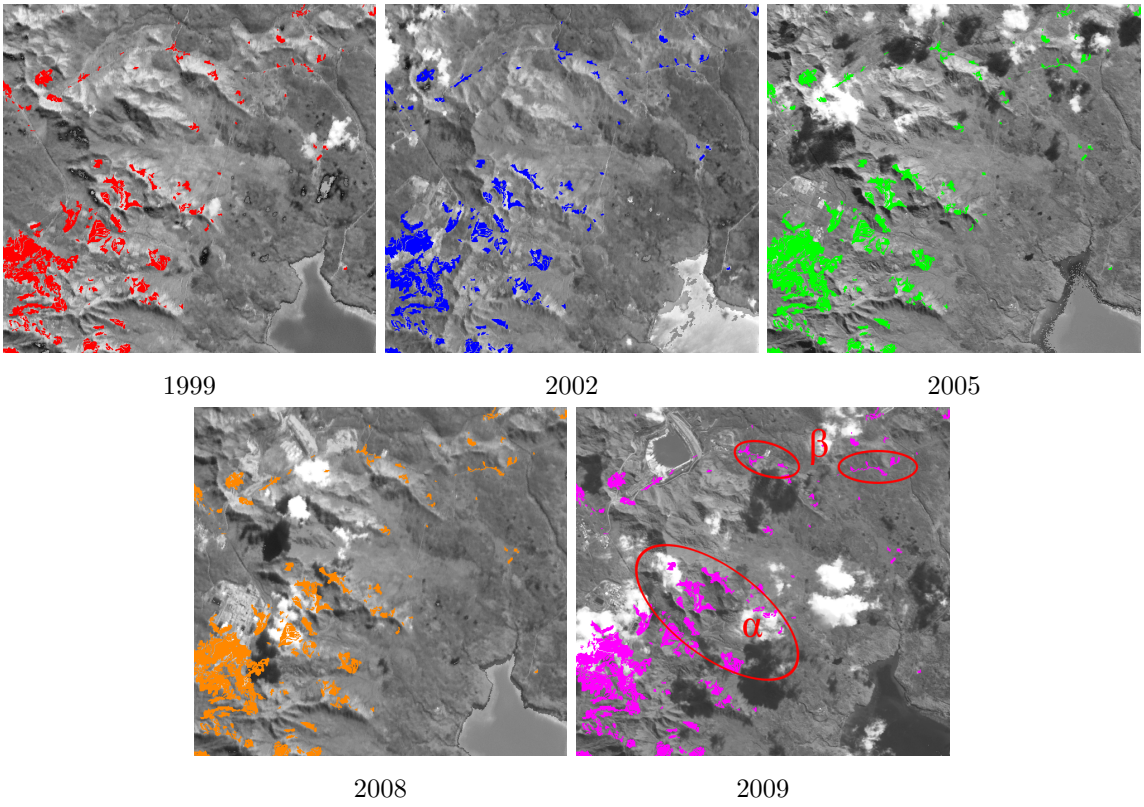


Figure 4.16 – Zone B, Redness4 $\overset{258}{\rightarrowtail}$ Redness4 $\overset{246}{\rightarrowtail}$ Redness4 $\overset{217}{\rightarrowtail}$ Redness4 $\overset{219}{\rightarrowtail}$ Redness4 . Suivi de l'évolution des zones qui étaient érodées en 1999.

Les augmentations de Redness les plus brusques sont illustrées par les figures 4.17 page ci-contre et 4.18 page 112. Pour le motif de la figure 4.17 Redness1 $\overset{17}{\rightarrowtail}$ NDVI0, Redness4 $\overset{15}{\rightarrowtail}$ Brightness4, NDVI0, Redness4 $\overset{16}{\rightarrowtail}$ Redness4, cette augmentation coïncide avec la création de la mine à l'Ouest de la zone, et dans une moindre mesure à la création du bassin de rétention dans le Nord.

Le motif NDVI3, Redness2 $\overset{\geq 15}{\rightarrowtail} * \overset{\geq 15}{\rightarrowtail} * \overset{\geq 15}{\rightarrowtail}$ Brightness4, NDVI0, Redness4 de la figure 4.18 montre de façon plus générale l'évolution de zones végétalisées (NDVI3) et moyennement

érodées (Redness2) vers des zones complètement dénudées, c'est-à-dire sans couvert végétal (NDVI0) et très érodées (Redness4). Comme précédemment, cette évolution radicale concerne les régions aux alentours de l'usine et du bassin de rétention.

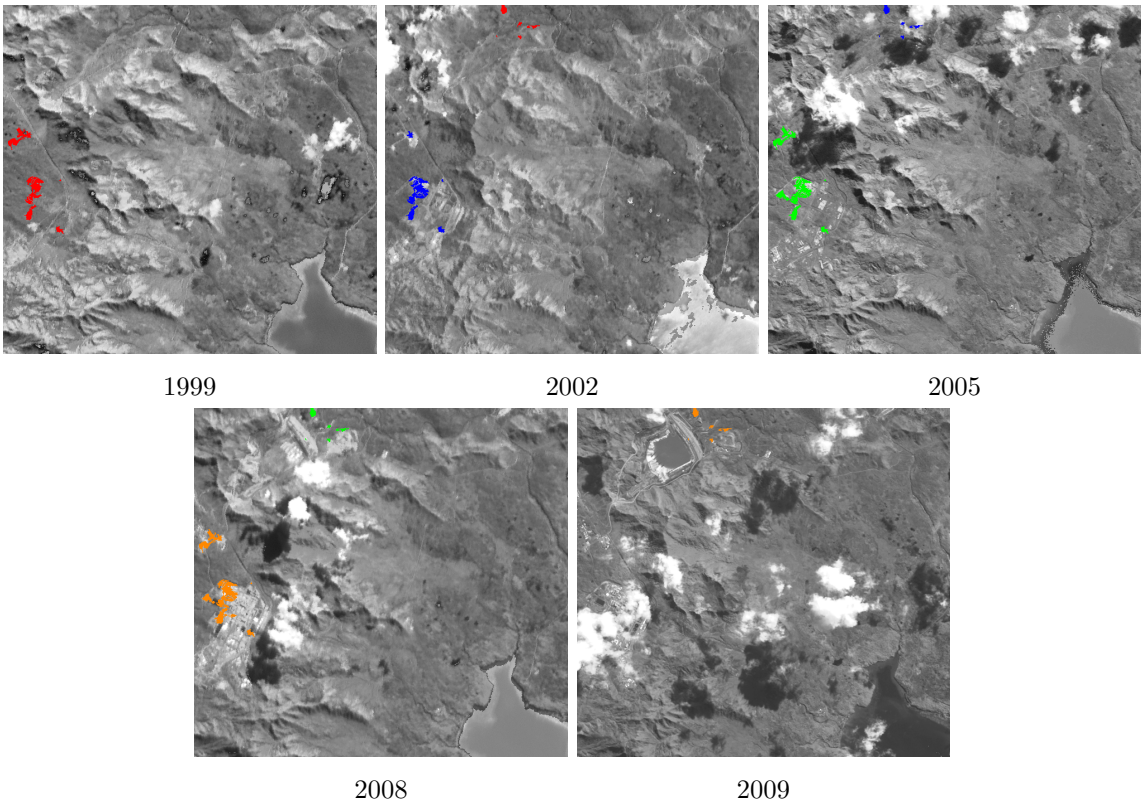


Figure 4.17 – Zone B, Redness1 $\overset{17}{\longmapsto}$ NDVI0, Redness4 $\overset{15}{\longmapsto}$ Brightness4, NDVI0, Redness4 $\overset{16}{\longmapsto}$ Redness4 . Ce motif montre une augmentation de Redness brutale entre 2 pas de temps puis une stabilisation de la valeur. Pendant toute la période où la valeur de Redness est stable, nous pouvons observer un agrandissement de la zone, notamment entre 2002 et 2005.

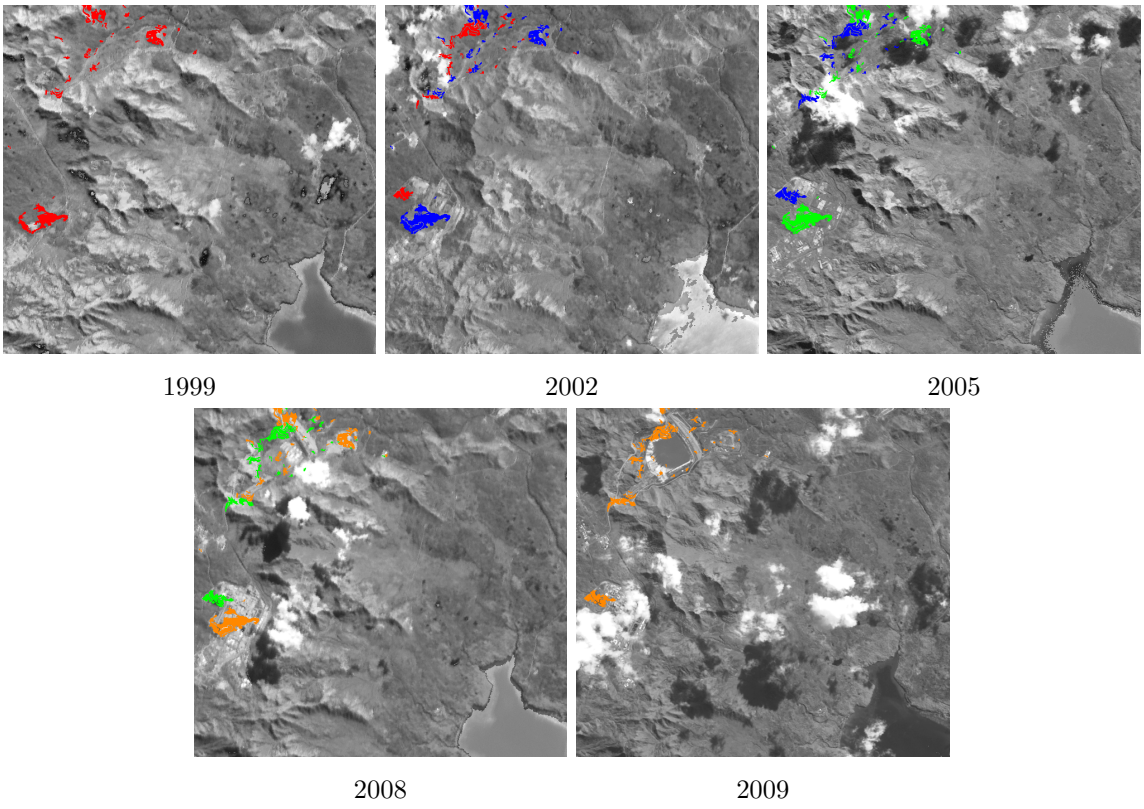


Figure 4.18 – Zone B, NDVI3, Redness2 $\overset{\geq 15}{\longmapsto} * \overset{\geq 15}{\longmapsto} * \overset{\geq 15}{\longmapsto}$ Brightness4, NDVI0, Redness4 . Régions végétalisées s'érodant très fortement au bout de 3 pas de temps.

Chapitre 5

Conclusion et perspectives

1. Contexte général

Nous avons vu que les phénomènes spatio-temporels peuvent se manifester :

— dans de nombreux contextes ; par exemple, on peut vouloir analyser des ensembles de trajectoires, des ensembles d'objets évoluant spatialement, ou des ensembles de caractéristiques d'objets évoluant temporellement.

— sous de nombreuses formes, en fonction des hypothèses prises et des phénomènes observés. Par exemple, on peut avoir :

  — des objets supposés indépendants par rapport au temps, ou au contraire s'influençant les uns les autres sous certaines conditions

  — des objets clairement identifiés (des véhicules, des quartiers), ou plus diffus (comme la Dengue)

  — des objets associés à plusieurs caractéristiques ou au contraire uniquement caractérisés par un identifiant (pour le cas des trajectoires de véhicules, par exemple)

  — des objets dont la structure et/ou la nature restent inchangées, ou au contraire pouvant changer de nature, fusionner, ou se diviser (pour le cas de zones érodées par exemple)

Chacune des études décrites dans l'état de l'art propose respectivement de s'adresser à des classes de problèmes couvrant une ou plusieurs formes de ces manifestations, sans pour autant toutes les prendre en compte à la fois. Par exemple, certaines méthodes de fouille de données pourront s'appliquer soit au cas où les objets sont fixes dans le temps (motifs spatio-temporels séquentiels, graphes de co-évolution), soit au cas où ils se déplacent (*co-locations, SOAP, SPCOZ, flow patterns*, etc.), mais pas les deux. De même, certaines méthodes prendront en compte un voisinage spatial, d'autres caractériseront les objets d'études avec de multiples attributs, et d'autres considéreront que les objets d'études peuvent apparaître et disparaître ; aucune n'englobe toutes ces hypothèses à la fois. Les phénomènes érosifs couvrent pourtant l'ensemble de ces conditions.

D'autre part, il arrive très souvent que l'application étudiée nécessite une connaissance du domaine détenue par un nombre limité d'utilisateurs. L'intégration de cette connaissance non-triviale dans l'analyse de motifs est nécessaire à la fois pour l'interprétation finale de résultats, mais aussi pour trier les motifs porteurs d'informations intéressantes. De plus, les données à étudier sont par définition trop nombreuses pour être analysables simplement par un humain, et nécessitent donc de recourir à des méthodes systématiques. Il devient tout aussi important de vouloir intégrer la connaissance directement dans les algorithmes de fouille de données pour profiter du filtrage des motifs afin de passer à l'échelle. De même, il est judicieux de vouloir éviter que certains motifs expriment la même information.

2. Contributions

Ce manuscrit présente ainsi deux contributions majeures : la recherche de chemins pondérés condensés dans un unique DAG attribué, l'utilisation de modèles experts pour la fouille de données sous contraintes. Plus précisément, nous donnons :

1. une proposition de modélisation générale de phénomènes spatio-temporels (le a-DAG) ;
2. un nouveau domaine de motifs (le chemin pondéré), permettant de prendre en compte les différentes façons dont une succession d'événements peut apparaître dans un a-DAG ;
3. une nouvelle représentation condensée pour les bases de données sous forme d'un graphe unique, dans un cadre formel ;
4. une exploitation des modèles du domaine déjà définis dans la littérature ;
5. une définition d'une nouvelle contrainte dérivée de ces modèles du domaine, avec plusieurs propriétés théoriques ;
6. l'intégration de cette contrainte dans les algorithmes d'extraction d'itemsets pour améliorer le passage à l'échelle.

Nous avons appliqué ces techniques à un ensemble de données spatio-temporelles portant sur l'érosion en Nouvelle-Calédonie. Nous avons établi pour cela deux scenarii de fouille de données.

Le premier scénario porte sur la recherche de propriétés de pixels exprimant une forte érosion d'après un modèle expert. Le deuxième scénario porte sur la recherche de chemins pondérés dans un a-DAG représentant les liens d'influence entre différents objets d'étude à travers le temps. Les résultats démontrent l'intérêt de l'approche ainsi que son efficacité, confirmée par les autres expérimentations réalisées sur des jeux de données réels (et jeux de données synthétiques pour la fouille de a-DAG). Les approches sont d'ailleurs génériques.

Afin d'évaluer l'apport d'information des motifs trouvés, nous avons développé une interface graphique permettant d'afficher les occurrences des motifs. Même si le développement de cet outil de visualisation n'a pas fait l'objet d'étude particulière, restituer dans le contexte de l'application l'information fournie par les motifs (c'est-à-dire, retranscrire les motifs sur les données initiales) est primordial : sans cela, l'utilisateur se heurte à une interprétation potentiellement difficile des motifs, et leur pertinence ne peut être facilement évaluée.

3. Perspectives

Nous avons évalué les performances de notre algorithme de recherche de chemins pondérés fréquents. Nous avons vu que le passage à l'échelle pourrait s'avérer compliqué au niveau du nombre d'arêtes ou du nombre de sommets, mais surtout par rapport à la taille des ensembles d'attributs. Il serait donc intéressant de trouver de nouvelles stratégies de recherche plus efficaces. D'autres expérimentations pourraient être menées, pour notamment évaluer l'impact du choix des graines à étendre en premier. Enfin, d'autres études de cas permettraient de conforter la validation par les experts de nos deux contributions majeures.

D'autres perspectives plus ambitieuses peuvent être envisagées ; nous les décrivons dans les paragraphes suivants.

3.1 Modélisation des phénomènes spatio-temporels

Les arcs du a-DAG proposé représentent un lien à la fois spatial et temporel. Le sens de cette relation spatio-temporelle dépend de la façon dont elle est calculée ; en l'occurrence, nous avons pris l'intersection des objets de temps consécutifs pour l'étude de cas sur l'érosion. Il pourrait être intéressant de dissocier les composantes spatiales et temporelles. Par exemple, un a-DAG ne permet pas de représenter les liens spatiaux des objets d'une date seule. Si l'on exprimait ces relations, nous étudierions des séquences de graphes de voisinage. Il ne s'agit cependant pas de graphes dynamiques : l'information temporelle exprimée par le a-DAG est une relation $m - n$, et non une relation $1 - 1$. Nous serions donc plutôt en présence d'un unique graphe comportant deux types d'arêtes : des arêtes simples pour le voisinage entre sommets d'une même date, et des arcs pour l'évolution temporelle d'un sommet vers un (des) autres(s). Des travaux sont en cours (dont entre autres ceux de Alatrista-Salas *et al.* (2012) ou de Desmier *et al.* (2013)), mais se limitent à des classes spécifiques de problèmes spatio-temporels (en l'occurrence, les relations temporelles sont de type $1 - 1$).

3.2 Nouveaux domaines de motifs pour la fouille d'un a-DAG

Bien qu'un a-DAG modélise les possibles fusions ou divisions des objets d'étude, aucune contrainte ne permet de filtrer ces types d'évolutions durant le processus de fouille. Afin de faire ressurgir un motif exprimant une fusion ou une division, il est ainsi nécessaire d'étendre le domaine de motifs des chemins pondérés à des sous-graphes. L'évaluation de la fréquence peut être ainsi rendue plus compliquée, et l'expression de non-redondance d'information (représentation condensée sans perte) devra être formalisée dans un cadre théorique plus large. De nouvelles mesures d'intérêt devront être développées pour s'adapter à ces nouveaux domaines de motifs, et à leur application à des études de cas spatio-temporelles. Avec de plus riches domaines de motifs, le passage à l'échelle s'avérera plus difficile, et représentera un défi encore plus grand.

3.3 Combinaison de contraintes de modèles et extension à de nouveaux domaines de motifs

L'intégration de connaissance experte, exprimée sous forme de modèles mathématiques, laisse entrevoir de nombreuses perspectives. Tout d'abord, on pourrait combiner plusieurs modèles, chacun étant éventuellement pondéré par les experts en fonction du contexte d'application. On pourrait alors comparer les résultats de chaque modèle pour :

- soit s'assurer que les modèles donnent des résultats similaires ;
- soit observer des différences de résultats entre modèles, auquel cas il serait possible de :
  - soit juger qu'un modèle donnant des résultats différents des autres est inadapté à l'étude de cas effectuée,
  - soit mettre en avant des motifs que seul le modèle discriminé est capable de capter.

Ces méthodes nous permettraient de répondre aux questions « Quelle est la connaissance commune à plusieurs modèles ? » et « Quelles sont les différences de connaissance exprimées par ces modèles ? ». Il serait donc intéressant de définir de nouvelles contraintes, comme des contraintes d'étonnement, ou de contradiction par rapport à une vérité terrain. En outre, il serait aussi intéressant d'appliquer ces nouvelles contraintes et celles développées dans cette thèse à d'autres domaines de motifs (par exemple, aux chemins pondérés d'un a-DAG).

3.4 Visualisation d'occurrences plus intuitives

La restitution des résultats de la fouille de données aux experts utilisateurs est essentielle dans le processus d'Extraction de Connaissances à partir des Données (ECD). La visualisation des motifs apporte donc une grande valeur ajoutée. Elle peut cependant s'avérer compliquée du fait de la complexité de certains domaines de motifs. Nous n'avons pas, par exemple, représenté dans les occurrences des arcs des chemins pondérés dans notre prototype de visualisation. Dans notre cas, cette information pouvait être dispensée car les arcs du a-DAG étaient construits entre segments spatialement très proches (pour rappel, en fonction d'un pourcentage d'intersection des surfaces). Dans d'autres cas, cette information pourrait s'avérer nécessaire, mais difficilement représentable avec suffisamment de clarté à cause du nombre d'arcs à dessiner (qui dépend directement du support). Une solution éventuelle serait une illustration en 3 dimensions, mais cette représentation pourrait rendre moins intuitive la restitution d'informations.

En outre, le fait que certains segments soient supportés par plusieurs sommets d'un même chemin pondéré pose un problème de superposition des couleurs. Il faudrait donc laisser à l'utilisateur le choix d'afficher seulement certaines occurrences. De façon générale, le post-traitement est coûteux, et nécessiterait un projet de recherche à part entière.

Bibliographie

Agrawal, Rakesh et Ramakrishnan Srikant (1994). « Fast Algorithms for Mining Association Rules in Large Databases ». In : *VLDB*. Sous la direction de Jorge B Bocca, Matthias Jarke et Carlo Zaniolo. Tome 1215. Morgan Kaufmann, pages 487–499 (cité pages 8, 12, 37, 39, 75).

Alatrista-Salas, Hugo (2013). « Extraction de relations spatio-temporelles à partir des données environnementales et de la santé ». Thèse de doctorat. Université Montpellier II (cité pages 4, 10, 18).

Alatrista-Salas, Hugo, Sandra Bringay, Frédéric Flouvat, Nazha Selmaoui-Folcher et Maguelonne Teisseire (2012). « The pattern next door : Towards spatio-sequential pattern discovery ». In : *Advances in Knowledge Discovery and Data Mining*. Springer, pages 157–168 (cité pages 9, 28, 117).

Anand, Sarabjot S., David A. Bell et John G. Hughes (1995). « The role of domain knowledge in data mining ». In : *Proceedings of the fourth international conference on Information and knowledge management - CIKM '95*. New York, New York, USA : ACM Press, pages 37–43 (cité page 13).

Antunes, Cláudia (2008). « An ontology-based framework for mining patterns in the presence of background knowledge ». In : *ICAI*, pages 1–6 (cité pages 13, 44).

Asai, Tatsuya, Hiroki Arimura, Takeaki Uno et Shin-Ichi Nakano (2003). « Discovering Frequent Substructures in Large Unordered Trees ». In : *Discovery Science*, pages 47–61 (cité page 30).

Atherton, James (2005). *Watershed Assessment for Healthy Reefs and Fisheries*. Rapport technique 679 (cité pages 13, 73–75, 88, 90, 95).

Bailey, N.T.J. (1975). *The mathematical theory of infectious diseases and its applications*. Charles Griffin & Company Ltd (cité pages 13, 29, 73).

Balcázar, José L, Albert Bifet et Antoni Lozano (2010). « Mining frequent closed rooted trees ». In : *Mach. Learn.* 78.1-2, pages 1–33 (cité page 30).

Beucher, Serge et Christian Lantuéjoul (1979). « Use of watersheds in contour detection ». In : (cité page 104).

Borgelt, Christian et M.R. Berthold (2002). « Mining molecular fragments : finding relevant substructures of molecules ». In : *ICDM*. IEEE Comput. Soc, pages 51–58 (cité pages 10, 29, 31, 32).

Boulicaut, Jean-François et Baptiste Jeudy (2010). « Constraint-based Data Mining ». In : *Data Mining and Knowledge Discovery Handbook*. Sous la direction d'Oded Maimon et Lior Rokach. Springer, pages 339–354 (cité pages 13, 37).

BRINGMANN, Björn et Siegfried NIJSSEN (2008). « What is frequent in a single graph ? » In : *PAKDD '08*, pages 858–863 (cité pages 40, 41, 53).

BRISSON, Laurent, Martine COLLARD et Nicolas PASQUIER (2005). « Improving the knowledge discovery process using ontologies ». In : *1st international workshop on Mining Complex Data in conjunction with ICDM (5th IEEE International Conference on Data Mining)* (cité pages 13, 44).

BURATTINI, M.N., M. CHEN, A. CHOW, F.A.B. COUTINHO, K.T. GOH, L.F. LOPEZ, S. MA et E. MASSAD (2008). « Modelling the control strategies against dengue in Singapore ». In : *Epidemiology and infection* 136.3, pages 309–19 (cité pages 13, 73).

CALDERS, Toon, Christophe RIGOTTI et Jean-François BOULICAUT (2004). « A Survey on Condensed Representations for Frequent Sets ». In : *Constraint-Based Mining and Inductive Databases*, pages 64–80 (cité page 54).

CAO, Huiping, Nikos MAMOULIS et David W CHEUNG (2005). « Mining Frequent Spatio-Temporal Sequential Patterns ». In : *ICDM*. IEEE Computer Society, pages 82–89 (cité pages 9, 21).

— (2007). « Discovery of Periodic Patterns in Spatiotemporal Sequences ». In : *IEEE Trans. Knowl. Data Eng.* 19.4, pages 453–467 (cité page 21).

CAO, Longbing (2010). « Domain-driven data mining : Challenges and prospects ». In : *Knowledge and Data Engineering, IEEE Transactions* 22.6, pages 755–769 (cité page 13).

CELIK, Mete, Shashi SHEKHAR, James P ROGERS et James A SHINE (2006). « Sustained Emerging Spatio-Temporal Co-occurrence Pattern Mining : A Summary of Results ». In : *ICTAI*, pages 106–115 (cité pages 9, 23).

— (2008). « Mixed-Drove Spatiotemporal Co-Occurrence Pattern Mining ». In : *IEEE Trans. Knowl. Data Eng.* 20.10, pages 1322–1335 (cité pages 9, 23).

CERF, Loïc, Jérémy BESSON, Céline ROBARDET et Jean-François BOULICAUT (2008). « Data-Peeler : Constraint-Based Closed Pattern Mining in n-ary Relations ». In : *SIAM* Proc. SIAM, pages 37–48 (cité pages 50, 55, 60).

CHEHREGHANI, Mostafa Haghir (2011). « Efficiently Mining Unordered Trees ». In : *ICDM'11*, pages 111–120 (cité page 30).

CHEN, Yen-liang, Hung-pin KAO et Ming-tat KO (2004). « Mining DAG Patterns from DAG Databases ». In : *Advances in Web-Age Information Management*, pages 579–588 (cité pages 34, 35, 39).

CRESSIE, Noel (1993). *Statistics for Spatial Data*. Rev Sub. Wiley-Interscience (cité page 28).

DE CASTRO MEDEIROS, Líliam César, César Augusto Rodrigues CASTILHO, Cynthia BRAGA, Wayner Vieira de SOUZA, Leda REGIS et Antonio Miguel Vieira MONTEIRO (2011). « Modeling the dynamic transmission of dengue fever : investigating disease persistence. » In : *PLoS neglected tropical diseases* 5.1, e942 (cité pages 13, 73).

DEHASPE, Luc, Hannu TOIVONEN et Ross D KING (1998). « Finding Frequent Substructures in Chemical Compounds ». In : *KDD*. Tome 98, page 1998 (cité page 31).

DESMIER, Elise (2014). « Co-evolution Pattern Mining in Dynamic Attributed Graphs ». Thèse de doctorat. INSA de Lyon (cité page 33).

Desmier, Elise, Marc Plantevit, Céline Robardet et Jean-François Boulicaut (2012). « Cohesive co-evolution patterns in dynamic attributed graphs ». In : *Discovery Science*. Springer, pages 110–124 (cité pages 33, 39).

— (2013). « Trend Mining in Dynamic Attributed Graphs ». In : *ECML/PKDD*. Springer, pages 654–669 (cité pages 32, 33, 117).

Domingos, Pedro (2007). « Toward knowledge-rich data mining ». In : *Data Mining and Knowledge Discovery* 15.1, pages 21–28 (cité page 13).

Du, Xiaoxi, Ruoming Jin, Liang Ding, Victor E Lee et John H Thornton Jr. (2009). « Migration motif : a spatial - temporal pattern mining approach for financial markets ». In : *KDD*, pages 1135–1144 (cité page 9).

Fayyad, Usama M, Gregory Piatetsky-Shapiro et Padhraic Smyth (1996). « From Data Mining to Knowledge Discovery in Databases. » In : *AI Magazine* 17.3, pages 37–54 (cité pages 3, 4).

Fisher, Peter, Patrick Laube, Marc Kreveld et Stephan Imfeld (2005). « Finding REMO - Detecting Relative Motion Patterns in Geospatial Lifelines ». In : *Developments in Spatial Data Handling*. Springer Berlin Heidelberg, pages 201–215 (cité page 22).

Flanagan, D C, J E Gilley et T G Franti (2007). « Water Erosion Prediction Project (WEPP) : Development History, Model Capabilities and Future Enhancements ». In : *ASABE* 50, pages 1603–1612 (cité page 90).

Flouvat, Frédéric, Jean-François N'guyen Van Soc, Elise Desmier et Nazha Selmaoui-Folcher (2014a). « Domain-driven co-location mining : Extraction, visualization and integration in a GIS ». In : *GeoInformatica* (cité page 23).

Flouvat, Frédéric, Jérémy Sanhes, Claude Pasquier, Nazha Selmaoui-Folcher et Jean-François Boulicaut (2014b). « Improving pattern discovery relevancy by deriving constraints from expert models ». In : *ECAI*, pages 327–332 (cité page 49).

— (2014c). « Les modèles experts : une source d'informations pour l'extraction de motifs ». In : *RFIA* (cité page 49).

Fukuzaki, Mutsumi, Mio Seki, Hisashi Kashima et Jun Sese (2010). « Finding itemset-sharing patterns in a large itemset-associated graph ». In : *PAKDD'10*, pages 147–159 (cité pages 10, 11, 49, 51).

Gaba, Eric (2013). *Carte administrative vierge de la collectivité territoriale de Nouvelle-Calédonie, France, destinée à la géolocalisation* (cité page 88).

Giannotti, Fosca et Dino Pedreschi, éditeurs (2008). *Mobility, Data Mining and Privacy - Geographic Knowledge Discovery*. Springer (cité page 18).

Giannotti, Fosca, Mirco Nanni, Fabio Pinelli et Dino Pedreschi (2007). « Trajectory pattern mining ». In : *KDD*. Sous la direction de Pavel Berkhin, Rich Caruana et Xindong Wu. ACM, pages 330–339 (cité page 21).

Goyal, Amit, Francesco Bonchi et Laks V S Lakshmanan (2011). « A Data-based approach to Social Influence Maximization ». In : *Proceedings of the VLDB Endowment* 5.Mc, pages 73–84 (cité page 29).

Gudmundsson, Joachim, Marc J van Kreveld et Bettina Speckmann (2004). « Efficient detection of motion patterns in spatio-temporal data sets ». In : *Proceedings of the 12th ACM International Workshop on Geographic Information Systems, ACM-GIS 2004, No-*

vember 12-13, 2004, Washington, DC, USA. Sous la direction de Dieter PFOSER, Isabel F CRUZ et Marc RONTHALER. ACM, pages 250–257 (cité page 22).

GÜNNEMANN, Stephan et Thomas SEIDL (2010). « Subgraph Mining on Directed and Weighted Graphs ». In : *PAKDD*, pages 133–146 (cité page 34).

HAI, Phan Nhat, Pascal PONCELET et Maguelonne TEISSEIRE (2012). « GeT_Move : An Efficient and Unifying Spatio-Temporal Pattern Mining Algorithm for Moving Objects ». In : *CoRR* abs/1204.0, page 17 (cité pages 9, 22).

HSU, Wynne, Mong Li LEE et Junmei WANG (2009a). « Mining Generalized Flow Patterns ». In : *Temporal and spatio-temporal Data Mining*. IGI Publishing, pages 189–208 (cité page 9).

— (2009b). « Mining Spatio-Temporal Trees ». In : *Temporal and spatio-temporal Data Mining*. IGI Publishing, pages 209–226 (cité page 9).

HUANG, Yan, Shashi SHEKHAR et Hui XIONG (2004). « Discovering Colocation Patterns from Spatial Data Sets : A General Approach ». In : *IEEE Trans. Knowl. Data Eng.* 16.12, pages 1472–1485 (cité pages 23, 33).

HUANG, Yan, Liqin ZHANG et Pusheng ZHANG (2008). « A Framework for Mining Sequential Patterns from Spatio-Temporal Event Data Sets ». In : *IEEE Trans. Knowl. Data Eng.* 20.4, pages 433–448 (cité pages 27, 33).

INOKUCHI, Akihiro, Takashi WASHIO et Hiroshi MOTODA (2000). « An Apriori-Based Algorithm for Mining Frequent Substructures from Graph Data ». In : *PKDD*. Sous la direction de Djamel A ZIGHED, Henryk Jan KOMOROWSKI et Jan M ZYTKOW. Tome 1910. Lecture Notes in Computer Science. Springer, pages 13–23 (cité page 10).

JAROSZEWICZ, Szymon et Dan A SIMOVICI (2004). « Interestingness of frequent itemsets using Bayesian networks as background knowledge ». In : *ACM SIGKDD*, pages 178–186 (cité pages 12, 43).

JAROSZEWICZ, Szymon, Tobias SCHEFFER et Dan A SIMOVICI (2009). « Scalable pattern mining with Bayesian networks as background knowledge ». In : *DMKD* 18.1, pages 56–100 (cité pages 13, 44).

KEMPE, David, Jon KLEINBERG et Éva TARDOS (2003). « Maximizing the spread of influence through a social network ». In : *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, pages 137–146 (cité page 29).

KIMURA, Masahiro, Kazumi SAITO, Ryohei NAKANO et Hiroshi MOTODA (2009). « Extracting influential nodes on a social network for information diffusion ». In : *Data Mining and Knowledge Discovery* 20.1, pages 70–97 (cité page 29).

KONTONASIOS, Kleanthis-Nikolaos, Jilles VREEKEN et Tijl DE BIE (2013). « Maximum entropy models for iteratively identifying subjectively interesting structure in real-valued data ». In : *Machine Learning and Knowledge Discovery in Databases*. Springer, pages 256–271 (cité page 44).

KURAMOCHI, Michihiro et George KARYPIS (2005). « Finding Frequent Patterns in a Large Sparse Graph* ». In : *Data Min. Knowl. Discov.* 11.3, pages 243–271 (cité page 40).

KUZNETSOV, Sergei O et Sergei A OBIEDKOV (2002). « Comparing performance of algorithms for generating concept lattices ». In : *J. Exp. Theor. Artif. Intell.* 14.2-3, pages 189–216 (cité pages 42, 81).

Lane, L.J. et M.A. Nearing (1989). « USDA - Water Erosion Prediction Project : Hillslope Profile Model Documentation ». In : (cité pages 73, 74, 90).

Mabit, Loïc, Nazha Selmaoui-Folcher et Frédéric Flouvat (2011). « Modélisation de la dynamique de phénomènes spatio-temporels par des séquences de motifs ». In : *EGC*. Sous la direction d'Ali Khenchaf et Pascal Poncelet. Tome RNTI-E-20. Revue des Nouvelles Technologies de l'Information. Hermann-Éditions, pages 455–466 (cité pages 9, 55).

Mamoulis, Nikos, Huiping Cao, George Kollios, Marios Hadjieleftheriou, Yufei Tao et David W Cheung (2004). « Mining, indexing, and querying historical spatiotemporal data ». In : *KDD*, page 236 (cité page 21).

Mannila, Heikki, Hannu Toivonen et A Inkeri Verkamo (1997). « Levelwise Search and Borders of Theories in Knowledge Discovery. » In : *Data Mining and Knowledge Discovery* 1.3, pages 241–258 (cité pages 37–39, 76).

Mathioudakis, Michael, Francesco Bonchi, Carlos Castillo, Aristides Gionis et Antti Ukkonen (2011). « Sparsification of influence networks ». In : *ACM SIGKDD*. ACM, pages 529–537 (cité page 38).

McGarry, Ken (2005). « A survey of interestingness measures for knowledge discovery ». In : *The Knowledge Engineering Review* 20.01, page 39 (cité pages 12, 40, 43).

Miyoshi, Yuuki, Tomonobu Ozaki et Takenao Ohkawa (2009). « Frequent Pattern Discovery from a Single Graph with Quantitative Itemsets ». In : *2009 IEEE International Conference on Data Mining Workshops*, pages 527–532 (cité pages 11, 32, 41).

Mohan, Pradeep, Shashi Shekhar, James A Shine et James P Rogers (2010). « Cascading Spatio-temporal Pattern Discovery : A Summary of Results ». In : *SDM*, pages 327–338 (cité pages 9, 10, 33).

Morgan, R.P.C (2001). « A simple approach to soil loss prediction : a revised Morgan-Morgan-Finney model ». In : *Catena* 44.4, pages 305–322 (cité pages 13, 73, 74, 88, 90).

Moser, Flavia, Recep Colak, Arash Rafiey et Martin Ester (2009). « Mining Cohesive Patterns from Graphs with Feature Vectors ». In : *SDM '09*, pages 593–604 (cité page 11).

Mougel, Pierre-Nicolas, Christophe Rigotti et Olivier Gandrillon (2012). « Finding Collections of k-Clique Percolated Components in Attributed Graphs ». In : *PAKDD*, pages 181–192 (cité page 39).

Nearing, M.A., G.R. Foster, L.J. Lane et S.C. Finkner (1989). « A process-based soil erosion model for USDA-Water Erosion Prediction Project Technology ». In : *ASAE* 32, pages 1587–1593 (cité page 90).

Ng, Raymond T., Laks V. S. Lakshmanan, Jiawei Han et Alex Pang (1998). « Exploratory mining and pruning optimizations of constrained associations rules ». In : *ACM SIGMOD Record* 27.2, pages 13–24 (cité pages 12, 39).

Padmanabhan, Balaji et Alexander Tuzhilin (1998). « A Belief-Driven Method for Discovering Unexpected Patterns ». In : *KDD*, pages 94–100 (cité pages 12, 43).

Parent, Christine, Stefano Spaccapietra, Chiara Renso, Gennady Andrienko, Natalia Andrienko, Vania Bogorny, Maria Luisa Damiani, Aris Gkoulalas-Divanis, Jose Macedo, Nikos Pelekis, Yannis Theodoridis et Zhixian Yan (2013). « Semantic

Trajectories Modeling and Analysis ». In : *ACM Comput. Surv.* 45.4, pages 1–32 (cité page 13).

PASQUIER, Claude, Jérémy SANHES, Frédéric FLOUVAT et Nazha SELMAOUI-FOLCHER (2013). « Extraction de motifs fréquents dans des arbres attribués. » In : *EGC*. Sous la direction de Christel VRAIN, André PÉNINOU et Florence SÈDES. Tome RNTI-E-24. Revue des Nouvelles Technologies de l'Information. Hermann-Éditions, pages 193–204 (cité pages 30, 32, 39).

PASQUIER, Claude, Frédéric FLOUVAT, Jérémy SANHES et Nazha SELMAOUI-FOLCHER (2014). « Extraction de motifs dans des graphes orientés attribués en présence d'automorphisme ». In : *EGC*. Tome E-26. Rennes, France. : Revue des Nouvelles Technologie de l'Information, pages 371–382 (cité pages 30, 39).

PASQUIER, Nicolas, Yves BASTIDE, Rafik TAOUIL et Lotfi LAKHAL (1999). « Discovering Frequent Closed Itemsets for Association Rules ». In : *ICDT*. Springer, pages 398–416 (cité pages 38, 42, 54).

PAUL-HUS, Catherine (2011). « Méthodes d'étude de l'érosion et gestion des sites dégradés en Nouvelle-Calédonie ». Mémoire de master. Université de Sherbrooke (cité pages 95, 96).

PEI, Jian, Jiawei HAN et Laks V. S. LAKSHMANAN (2001a). « Mining Frequent Itemsets with Convertible Constraints ». In : *Data Engineering*. Section 4, pages 433–442 (cité pages 12, 39).

PEI, Jian, Jiawei HAN, Behzad MORTAZAVI-ASL, Helen PINTO, Qiming CHEN, Umeshwar DAYAL et Mei-Chun HSU (2001b). « Prefixspan : Mining sequential patterns efficiently by prefix-projected pattern growth ». In : *2013 IEEE 29th International Conference on Data Engineering (ICDE)*. IEEE Computer Society, page 215 (cité pages 27, 56).

POEZEVARA, Guillaume, Bertrand CUISSART et Bruno CRÉMILLEUX (2011). « Extracting and summarizing the frequent emerging graph patterns from a dataset of graphs ». In : *Journal of Intelligent Information Systems* 37.3, pages 333–353 (cité pages 31, 32).

PRYOR, Graham (2012). « Why manage research data ». In : *Managing research data*, pages 1–16 (cité page 3).

QIAN, Feng, Qinming HE et Jiangfeng HE (2009). « Mining Spread Patterns of Spatio-temporal Co-occurrences over Zones ». In : *ICCSA (2)*. Sous la direction d'Osvaldo GERVASI, David TANIAR, Beniamino MURGANTE, Antonio LAGANÀ, Youngsong MUN et Marina L GAVRILOVA. Tome 5593. Lecture Notes in Computer Science. Springer, pages 677–692 (cité page 23).

RAEDT, Luc De et Albrecht ZIMMERMAN (2007). « Constraint-Based Pattern Set Mining ». In : *SDM*, pages 1–12 (cité page 12).

RAISSI, Chedy et Pascal PONCELET (2007). « Sampling for Sequential Pattern Mining : From Static Databases to Data Streams ». In : *ICDM*. IEEE, pages 631–636 (cité page 38).

RENARD, Kenneth G, George R FOSTER, Glenn A WEESIES, D K MCCOOL, D C YODER et OTHERS (1997). « Predicting soil erosion by water : a guide to conservation planning with the revised universal soil loss equation (RUSLE). » In : *Agriculture Handbook (Washington)* 703 (cité pages 88, 90).

Rouet, Isabelle, Dominique Gay, Michel Allenbach, Nazha Selmaoui, Anne-Gaelle Ausseil, Morgan Mangeas, Johnatan Maura, Pascal Dumas et Didier Lille (2009). « Tools for soil erosion mapping and hazard assessment : application to New Caledonia, SW Pacific ». In : *International Congress on Modelling and Simulation (MODSIM'09), Cairns, Australia*, pages 1986–1992 (cité page 87).

Sanhes, Jérémy, Frédéric Flouvat, Nazha Selmaoui-Folcher et Jean-François Boulicaut (2012). « Extraction d'arbres spatio-temporels d'itemsets pour le suivi environnemental. » In : *EGC*, pages 581–582 (cité page 30).

Sanhes, Jérémy, Frédéric Flouvat, Claude Pasquier, Nazha Selmaoui-Folcher et Jean-François Boulicaut (2013a). « Extraction de motifs condensés dans un unique graphe orienté acyclique attribué ». In : *EGC*. Tome RNTI-E-24. Toulouse, France : Hermann-Editions, pages 205–216 (cité page 49).

Sanhes, Jérémy, Frédéric Flouvat, Nazha Selmaoui-Folcher, Claude Pasquier et Jean-François Boulicaut (2013b). « Weighted Path as a Condensed Pattern in a Single Attributed DAG ». In : *IJCAI*, pages 1642–1648 (cité page 49).

Selmaoui-Folcher, Nazha et Frédéric Flouvat (2011). « How to Use "Classical" Tree Mining Algorithms to Find Complex Spatio-Temporal Patterns ? » In : *DEXA (2)*, pages 107–117 (cité page 30).

Selmaoui-Folcher, Nazha, Frédéric Flouvat, Dominique Gay et Isabelle Rouet (2011). « Spatial pattern mining for soil erosion characterization ». In : *IJAEIS* 2.2, pages 73–92 (cité page 23).

Selmaoui-folcher, Nazha, Frédéric Flouvat, Hugo Altrista-Salas et Sandra Bringay (2013). « Motifs spatio-temporels : enjeux et applications à l'environnement ». In : *Revue d'Intelligence Artificielle*, pages 1–30 (cité pages 18, 21).

Termier, Alexandre, Marie-Christine Rousset et Michele Sebag (2004). « DRYADE : A New Approach for Discovering Closed Frequent Trees in Heterogeneous Tree Databases ». In : *ICDM*, pages 543–546 (cité page 30).

Termier, Alexandre, Yoshinori Tamada, Kazuyuki Numata, Seiya Imoto, Takashi Washio, Tomoyuki Higushi et Tomoyuki Higuchi (2007). « DigDag, a first algorithm to mine closed frequent embedded sub-DAGs ». In : *MLG*, pages 1–5 (cité pages 32, 34, 39).

Tobler, Waldo (1970). « A computer movie simulating urban growth in the Detroit region ». In : *Economic Geography* 46.2, pages 234–240 (cité pages 11, 18, 34).

Tsoukatos, Ilias et Dimitrios Gunopulos (2001). « Efficient Mining of Spatiotemporal Patterns ». In : *SSTD*. Sous la direction de Christian S Jensen, Markus Schneider, Bernhard Seeger et Vassilis J Tsotras. Tome 2121. Lecture Notes in Computer Science. Springer, pages 425–442 (cité pages 9, 25).

Uno, Takeaki, Tatsuya Asai, Yuzo Uchida et Hiroki Arimura (2003). « LCM : An Efficient Algorithm for Enumerating Frequent Closed Item Sets. » In : *FIMI*. Sous la direction de Roberto J Bayardo Jr. et Mohammed Javeed Zaki. Tome 90. CEUR Workshop Proceedings. CEUR-WS.org (cité page 42).

Wang, Junmei, Wynne Hsu, Mong-Li Lee et Jason Tsong-Li Wang (2004). « FlowMiner : Finding Flow Patterns in Spatio-Temporal Databases ». In : *ICTAI*. IEEE Computer Society, pages 14–21 (cité pages 9, 26).

Wang, Junmei, Wynne Hsu et Mong-Li Lee (2005). « Mining Generalized Spatio-Temporal Patterns ». In : *DASFAA*. Sous la direction de Lizhu Zhou, Beng Chin Ooi et Xiaofeng Meng. Tome 3453. Lecture Notes in Computer Science. Springer, pages 649–661 (cité pages 9, 27).

Wang, Junmei, Wynne Hsu, Mong Li Lee et Chang Sheng (2006). « A Partition-Based Approach to Graph Mining ». In : *ICDE*. IEEE Computer Society, pages 74–74 (cité page 10).

Washio, Takashi et Hiroshi Motoda (2003). « State of the art of graph-based data mining ». In : *SIGKDD Explor. Newsl.* 5.1, pages 59–68 (cité page 10).

Werth, Tobias, Alexander Dreweke, Marc Wörlein, Ingrid Fischer et Michael Philippsen (2008). « DAGMA : Mining Directed Acyclic Graphs ». In : *IADIS European Conference on Data Mining* (cité pages 34, 36).

Wischmeier, W.H. et D.D. Smith (1978). « Predicting rainfall erosion losses - A guide to conservation planning ». In : (cité pages 74, 88).

Yan, Xifeng et Jiawei Han (2002). « gSpan : Graph-Bases Substructure Pattern Mining ». In : *ICDM* 3, pages 721–724 (cité page 31).

— (2003). « CloseGraph ». In : *ACM SIGKDD*. Tome 6. New York, New York, USA : ACM Press, page 286 (cité pages 43, 54).

Yan, Xifeng, Jiawei Han et Ramin Afshar (2003). « CloSpan : Mining Closed Sequential Patterns in Large Datasets ». In : *SDM '03*, pages 166–177 (cité pages 43, 54).

Yang, Hui, Srinivasan Parthasarathy et Sameep Mehta (2005). « A generalized framework for mining spatio-temporal patterns in scientific data ». In : *KDD*. Sous la direction de Robert Grossman, Roberto J Bayardo et Kristin P Bennett. ACM, pages 716–721 (cité page 24).

Yao, Xiaobai (2003). « Research Issues in Spatio-temporal Data Mining ». In : *White paper UCGIS* (cité pages 8, 9).

Yuan, May (2008). « Toward Knowledge Discovery about Geographic Dynamics in Spatio-temporal Databases ». In : *Geographic Data Mining and Knowledge Discovery, Taylor and Francis*. Sous la direction de J Han et Harvey J Miller, pages 347–365 (cité pages 8, 9).

Zaki, Mohammed J (2002). « Efficiently mining frequent trees in a forest ». In : *KDD'02*, pages 71–80 (cité pages 29, 30).

— (2004). « Efficiently Mining Frequent Embedded Unordered Trees ». In : *Fundam. Inf.* 66.1-2, pages 33–52 (cité page 30).

Abstract

Spatio-temporal events denote a large range of phenomena with different characteristics. For example, migration flows studies appear to be very different from disease spread studies. Indeed, interestingness of the first relies on tracking trajectories, whereas the second is about finding the factors of spread. Moreover, each class of a spatio-temporal problem can be tackled differently, depending on which parameters are considered: the studied spatial neighbourhood, the number of characteristics associated with the objects, or whether events are supposed correlated or independent. As a result, data mining techniques are often specific to a sub-class of spatio-temporal problem, that is to say, to a limited set of hypothesis.

In order to bring out new knowledge from data, it seems to be necessary to enlarge this set of hypothesis, that is to say, to widen the field of possibilities regarding correlations that may exist between events. For this, we propose a new model that allows to take into account more considerations than existing studies. For example, this representation allows to model the complex spatio-temporal dynamic of erosion phenomenon: an object can be split up in several other objects, or can merge with other objects into one. More precisely, we use a single directed graph, that becomes acyclic thanks to the temporal component of the problem, and that is attributed by several characteristics. Mining a single graph is a non-trivial operation, and is even more complex because of the plurality of the attributes. We focus here on searching paths of attributes, under frequency and non-redundancy constraints. Those constraints have been largely studied for transactional databases, but have been less studied in the case of a single graph (or even not studied at all).

Conjointly to those primitive constraints, it is often necessary to filter the set of found patterns that can be too numerous and/or not relevant for experts. To do so, we need to solicit experts on the domain of the studied data. However, it is difficult to translate a wide knowledge of a given domain into constraints. In addition, such translation could plausibly bring some human mistakes. From this observation, we propose to use existing expert knowledge that has been expressed in the form of mathematical models and published in the litterature of the domain. These models present the advantages of being both highly informative and synthetic; their use avoids –or greatly reduce– human intervention. We focus on the case where those models are mathematical functions of several variables giving a result in $\mathbb{R}$, that we can use as an expert measure to define a minimum threshold-based constraint. We highlight some of its theorical properties enabling search space pruning for frequent itemsets mining.

Finally, we apply the two mining methods to the study of erosion in New-Caledonia. The studied data is heterogeneous with numerical and categorical values coming from multiple sources (e.g. from satellite images, digital elevation model, land cover truth or geology). We elaborate two scenarii. In the first one, we mine a set of pixels, that can be seen as a transactional database. We seek properties on pixels expressing a high erosion risk according to an expert model. In the second scenario, we mine a single attributed acyclic graph: we exploit the previous results to seek temporal series of characteristics leading to a high or low erosion risk. A visualisation prototype allows to remap and highlight occurrences of these paths. The results bear out the interest of proposed approaches. In particular, they highlights areas that are known for their erosive dynamic.

Keywords: *Spatio-temporal data mining, Single graph, Attributed DAG, Condensed weighted path, Model constraint, Domain knowledge, Erosion*

Résumé

Les événements spatio-temporels regroupent une large diversité de phénomènes comportant des caractéristiques propres. Par exemple, l'étude de flux migratoires se révèle ainsi très différente de l'étude de propagation de maladies. En effet, le domaine d'intérêt de la première porte sur le suivi des trajectoires, tandis que celui de la deuxième porte sur les facteurs de la propagation. De plus, chaque classe d'un problème spatio-temporel peut être abordée différemment, que l'on considère ou non un voisinage spatial, une caractérisation des objets d'étude unique ou multiple, ou bien une (in)dépendance entre les événements. Ainsi, les techniques de fouilles de données développées sont souvent restées spécifiques à une sous-classe de problème spatio-temporel, c'est-à-dire sous un ensemble restreint d'hypothèses.

Or, pour réussir à dégager des connaissances nouvelles à partir de données, il est nécessaire d'élargir cet ensemble d'hypothèses, c'est-à-dire élargir le champs des possibles quant aux corrélations qu'il peut exister entre événements. Nous proposons donc une modélisation de ces phénomènes spatio-temporels permettant de prendre en compte plus de considérations que dans l'état de l'art. En outre, cette modélisation permet d'exprimer des événements qui existent dans les phénomènes d'érosion : un objet d'étude peut se diviser en plusieurs objets, ou fusionner avec d'autres objets pour n'en former qu'un seul. Plus précisément, nous modélisons les dynamiques spatio-temporelles sous la forme d'un unique graphe orienté, que la composante temporelle des problèmes rend acyclique, et dont les sommets sont attribués par plusieurs caractéristiques. La fouille de données dans un unique graphe est une opération non-triviale, que la pluralité des attributs rend encore plus complexe. Nous nous concentrons sur la recherche de chemins d'attributs dans un tel graphe, sous contraintes de fréquence minimale et de non-redondance d'information. Ces contraintes, bien que très utilisées dans la littérature pour les bases de données transactionnelles, ont peu ou pas été étudiées dans le cas d'un graphe unique.

Conjointement à ces contraintes primitives, il est très souvent nécessaire de filtrer l'ensemble des motifs solution qui peuvent s'avérer trop nombreux et peu pertinents. Pour cela, les experts du domaine des données étudiées doivent être sollicités, afin d'exprimer les filtres pertinents sous forme de contraintes. Il est cependant difficile et fastidieux de traduire de façon complète la connaissance d'un domaine vers des contraintes. De plus, cette traduction est susceptible d'apporter son lot d'erreurs humaines. En partant de ce constat, nous proposons d'utiliser les connaissances expertes existant dans la littérature du domaine sous la forme de modèles mathématiques. Ces modèles présentent l'avantage d'être à la fois riches et synthétiques, et leur utilisation permet de réduire l'intervention humaine. Nous nous concentrons sur le cas des fonctions mathématiques donnant un résultat dans $\mathbb{R}$, que nous pouvons alors utiliser comme mesure experte ; nous en dérivons ainsi des contraintes de seuil minimum. Nous mettons en évidence des propriétés sur cette contrainte permettant d'élaguer l'espace de recherche d'itemsets fréquents, et ainsi améliorer le passage à l'échelle de l'extraction.

Enfin, nous appliquons les méthodes développées à l'étude de l'érosion en Nouvelle-Calédonie. À partir de sources d'information multiples (images satellite, modèles numériques de terrain, occupation des sols, pente, géologie) et hétérogènes (à valeur numérique, ou catégorielles), nous fouillons tantôt un ensemble de pixels pour un temps donné, tantôt un unique graphe acyclique attribué. Dans le premier cas, nous recherchons des caractéristiques communes aux pixels exprimant une forte érosion selon un modèle expert. Dans le deuxième cas, nous utilisons les résultats précédents pour rechercher des successions temporelles de caractéristiques menant à des zones présentant une forte ou faible érosion. Un prototype de visualiseur permet de retrouver et de mettre en évidence les occurrences de ces chemins. Les résultats obtenus confortent l'intérêt des approches proposées. Ils mettent notamment en avant des zones connues pour leur dynamique en matière d'érosion.

Mots-clés : *Fouille de données spatio-temporelles, Graphe unique, DAG attribué, Chemin pondéré condensé, Contrainte de modèle, Connaissance experte, Érosion*