MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITÉ MOULOUD MAMMERI DE TIZI OUZOU

FACULTÉ DE GÉNIE ÉLECTRIQUE ET D'INFORMATIQUE

DÉPARTEMENT D'INFORMATIQUE

THÈSE DE DOCTORAT LMD

DISCIPLINE : INFORMATIQUE

Option « Intelligence Artificielle et Systèmes d'Information »

Présentée par

Zina AIT YAKOUB

Sujet

Logiques de Description & Analyse de Concepts Formels pour des représentations incomplètes.

Devant le jury d'examen composé de :

Mr	AHMED-OUAMER Rachid	Professeur	UMMTO	Président
Mr	DJOUADI Yassine	Professeur	USTHB	Rapporteur
Mme	KHELLAF Faiza	Professeur	USTHB	Examinatrice
Mr	RASSOUL Idir	M.C.A	UMMTO	Examinateur
Mme	AMIROUCHE Fatiha	M.C.A	UMMTO	Examinatrice

Soutenue le: 01/03/2018

Remerciements

J'exprime toute ma gratitude à mon encadreur Mr DJOUADI Yassine, Professeur à l'USTHB, je le remercie très sincèrement pour le temps et la patience qu'il m'a accordés, tout au long de ces années. Ses grandes qualités professionnelles et humaines m'ont aidée à aller au bout de ce travail, dans la confiance et la reconnaissance.

Je remercie Mr AHMED OUAMER Rachid, Professeur à l'UMMTO, pour l'honneur qu'il nous fait pour présider ce jury. Mes remerciements vont également à Mme KHELLAF Faiza, Professeur à l'USTHB, à Mr RASSOUL Idir, maître de conférences classe A à l'UMMTO, et à Mme AMIROUCHE Fatiha, maître de conférences classe A à l'UMMTO, pour avoir accepté de consacrer leurs précieux temps à l'évaluation de ce travail.

Je remercie aussi chaleureusement Mr PRADE Henri et Mr DUBOIS Didier, chercheurs au laboratoire IRIT, pour leur bon accueil lors de mon stage et pour m'avoir facilité de communiquer avec eux malgré leur programme très chargé. J'éprouve un profond respect pour leur travail et leur parcours, ainsi que pour leurs qualités humaines dont je garderai un long souvenir. Je les remercie aussi pour leur collaboration dans les différents articles que nous avons publiés en commun, ce qui a permis de mener à bien ce travail.

Je suis très reconnaissante envers Mr TABIA Karim, maître de conférences classe A à l'université d'Artois, pour son accueil au laboratoire CRIL.

Un merci particulier à la famille AHMED ALI (Faiza, Kamilia et Cherif) pour leur présence et leur aide dans chacun de mes stages de recherche en France.

Je remercie du fond du cœur, peut être jamais assez, ma famille et particulièrement mes parents, ma sœur et mes frères qui ont toujours cru en moi et qui m'ont beaucoup aidée à surmonter mes inquiétudes quand l'avenir restait pour moi quelque peu aléatoire.

Merci enfin à tous mes amis (es), en particulier (Rabia, Farida, Sabrina), qui ont grandement contribué par leur présence et leur soutien moral pendant toutes ces années.

Résumé

Mots clés : analyse de concepts formels, opérateurs possibilistes, logiques de description, représentations incomplètes.

La théorie de l'analyse de concepts formels (ACF) dépend classiquement de l'utilisation de l'opérateur de dérivation de Galois. Les similitudes formelles entre la théorie des possibilités et l'analyse de concepts formels ont conduit à l'utilisation d'opérateurs possibilistes en ACF, qui ont été ignorés auparavant. Dans cette thèse, une approche basée sur l'utilisation de la composition asymétrique des deux opérateurs possibilistes les plus habituels est proposée. Elle permet de compléter la base de Duquinne-Guigues, en dérivant les implications d'attributs avec les disjonctions des deux côtés des implications. L'approche est également généralisée à des contextes incomplets et incertains impliquant des informations positives et négatives. Nous appliquons ces résultats, dans le cadre des logiques de description (LD), pour compléter une description terminologique (TBox) avec des GCIs disjonctives.

Abstract

Key words : formal concepts analysis, possibilistic operators, description logics, incomplete knowledge

Formal concept analysis theory (FCA) classically relies on the use of the Galois powerset operator. Formal similarities between possibility theory and formal concept analysis have led to the use of possibilistic operators in FCA, which were ignored before. In this thesis, an approach based on the use of asymmetric composition of the two most usual possibilistic operators is proposed. It enables us to complement the stem base, by deriving attribute implications with disjunctions on both sides of the implications. Besides, the approach is also generalized to incomplete contexts involving explicit positive and negative information. We outline the potential application of these results to the completion of TBoxes in description logic.

Table des matières

Liste des tableaux

Liste des figures

CHAPITRE 1

Introduction générale

Sommaire

1.1 Contexte et problématique

L'analyse de concepts formels (ACF) est une théorie mathématique introduite par Rudolf Wille [112] dans les années 1980. L'ACF consiste à apprendre des paires de sous-ensembles ({objets}, {attributs}), appelées concepts formels, à partir d'une relation binaire, appelée contexte formel. Le processus de découverte de concepts formels est basé sur l'opérateur classique de dérivation de Galois. Ces concepts formels sont organisés dans une hiérarchie (c'est-à-dire un ordre partiel), appelée treillis de concepts.

En particulier, le treillis de concepts s'est avéré très utile pour apprendre des implications d'attributs [95] à savoir des formules de la forme
$\{a_1, ..., a_n\} \longmapsto \{b_1, ..., b_m\}$ où les deux ensembles d'attributs $\{a_1, ..., a_n\}$ et $\{b_1, ..., b_m\}$ sont implicitement interprétées dans une forme conjonctive. C'est-à-dire que l'interprétation de $\{a_1, ..., a_n\} \longmapsto \{b_1, ..., b_m\}$ est "si nous avons a_1 et ... et a_n alors nous avons b_1 et ... et b_m ".

Il existe différentes représentations d'ensembles d'implications d'attributs. Missaoui et al. [85, 86] génèrent des implications d'attributs mixtes, c'est-à-dire des implications d'attributs impliquant au moins un attribut positif et un attribut négatif. La base de Luxemburger [82] concerne des implications d'attributs partiels (avec exceptions). La base de Duquenne-Guigues [66, 73] représente un ensemble minimal d'implications d'attributs.

Il s'avère que toutes les bases d'implications d'attributs existantes dans la littérature sont limitées à une représentation (sémantique) conjonctive des implications d'attributs. Cette limitation est la conséquence directe de l'utilisation de l'opérateur de dérivation classique de Galois (C'est-à-dire l'opérateur de suffisance).

La Logique de Description (LD) [10, 45] est une famille de formalismes de logiques de premier ordre permettant la représentation des connaissances sous forme de concepts (classe, prédicat unaire), de rôles (propriété d'objet, prédicat binaire), et d'individus (instances de classe). En logique de description, la connaissance est divisée en deux parties : un ensemble de formules relatives aux informations terminologiques (TBox), et un ensemble de formules relatives aux informations sur les assertions (ABox). La logique de description est devenue très répandue pour la représentation et la modélisation des connaissances.

Il existe plusieurs travaux qui visent à tirer profit de l'utilisation conjointe de la théorie de l'ACF et de la logique de description. Pour rappel, les points intéressants de ces deux formalismes sont le pouvoir d'expressivité et la notion de relations. On notera que la notion de concept, comme ensemble d'objets partageant des propriétés communes, apparaît dans ces deux formalismes, même si la façon de les définir formellement est différente. Il existe plusieurs approches quand à l'utilisation conjointe de l'ACF et des LDs. Les travaux qui enrichissent l'ACF en empruntant des propriétés plus complexes semblables à celles des constructeurs de concept dans les LDs [62, 97, 98, 101, 120], et les travaux qui emploient les méthodes de l'ACF pour résoudre certains problèmes rencontrés dans la représentation de connaissances à l'aide de LDs [2, 8, 11, 13–16, 33, 34, 36, 102, 104].

Il s'avère aussi que toutes les approches proposées pour obtenir ou compléter une base terminologique (TBox) à partir d'une base assertionnelle (ABox), s'appuient sur l'utilisation de l'opérateur de dérivation classique de Galois (c'est-à-dire l'opérateur de suffisance). Ces travaux sont basés sur le treillis de Galois de tous les concepts formels obtenus en utilisant la composition des opérateurs de suffisance. Par conséquent, les axiomes induits tels que les GCIs sont limités à leur forme conjonctive.

En 1987, Gorgov et al. [68] ont observé que la modalité de nécessité habituelle dans la logique modale pouvait être complétée par un autre opérateur qu'ils ont appelé opérateur de suffisance. Douze ans plus tard, Düntsch et Orlowska [58, 59] ont repris cette idée et étudiaient le cadre algébrique de la logique modale de la suffisance. Ils ont fourni la contrepartie suffisante des algèbres dites booléennes avec des opérateurs, à l'origine dus à Jónsson and Tarski [75], puis étudiaient des algèbres booléennes avec des opérateurs de nécessité et de suffisance. Comme la logique modale sémantique s'appuie sur les relations d'accessibilité, et comme l'opérateur de suffisance correspond précisément à l'opérateur de dérivation de Galois classique dans l'ACF, il est clair que quatre opérateurs de dérivation peuvent être appliqués aux contextes formels dans l'ACF. Ce cadre algébrique formel a été appliqué par Düntsch and Gediga [56,57] à l'analyse qualitative des données à grande échelle (non seulement à l'ACF, mais aussi aux rough sets, voir aussi Yao et Chen [117] pour plus de détails).

En 2007, Dubois et al. [47] ont donné une lecture possibiliste à la théorie de l'analyse de concepts formels, mettant en évidence les analogies entre les quatre fonctions de la théorie des possibilités et les quatre opérateurs de dérivation initialement soulignés par Düntsch et Orlowska [58]. En particulier, les propriétés de la composition hybride asymétrique des opérateurs de possibilité et de nécessité ont été redécouvertes indépendamment [39].

Classiquement, l'ACF s'applique à une relation binaire représentée par un contexte formel. Un objet satisfait un attribut ou ne le satisfait pas. Au cours des dernières années, l'ACF a été appliquée dans de nombreuses disciplines comme la psychologie, la sociologie, l'anthropologie, la médecine, la biologie, la linguistique, etc. Dans de tels cas, l'ACF traite inévitablement des structures d'information relationnelle (contextes formels) issues de l'investigation humaine (jugement, observation, mesure, etc.). De pareilles informations sont inévitablement entachées d'incertitude et/ou d'imprécision. Des recherches récentes [43] [42] [37] [47] [51] ont été initiées afin de prendre en considération des contextes formels contenant de pareilles informations (incertaines et/ou imprécises). Dans [43] [42], les auteurs ont traité la notion de gradualité pour les concepts formels flous et l'impact de l'incertitude sur la notion de concept formel dans le cadre de la théorie des possibilités. Ils ont également introduit la notion de typicalité pour les objets et la notion d'importance pour les attributs.

Dans [94] et [25], les auteurs ont soulevé la question de distinguer entre le cas où l'on sait qu'un objet ne possède pas un attribut et le cas où on ne sait pas si l'objet possède l'attribut ou non, c'est une distinction que les contextes habituels ne peuvent pas faire. Ils ont proposé d'introduire une troisième valeur, notée " ?", dans un contexte formel, ce qui conduit à la notion de contexte incomplet.
D'autres aspects de degrés sont traités comme l'incertitude. Dans ce cas, les attributs peuvent rester non flous mais la certitude de la satisfaction de l'attribut par un objet n'est pas complète. Elle peut être représentée par une distribution de probabilités ou de possibilités au sens de [43] [42]. Plus précisément, dans une relation décrivant un contexte incertain, les cases sont remplies par des paires (α, β) où α est le degré de certitude que l'objet ait l'attribut et β est le degré de nécessité que l'objet n'ait pas l'attribut avec $min(\alpha, \beta) = 0$. Les situations (1, 0) et (0, 1) correspondent aux situations complètement informées où on sait avec certitude que l'objet a la propriété ou qu'il ne l'a pas. La situation (0, 0) reflète l'ignorance totale et celle où $0 < max(\alpha, \beta) < 1$ reflète l'ignorance partielle.
Dans toutes les approches qu'on vient de citer, les auteurs n'ont pas traité les implications d'attributs qui peuvent être obtenues à partir de ces différents contextes formels.

1.2 Approches et contributions

Dans cette thèse, nous proposons d'élargir la sémantique sous-jacente des implications d'attributs en considérant les "implications d'attributs disjonctives". Cette proposition se base sur l'utilisation d'opérateurs de dérivation autre que celui classiquement utilisé, à savoir les opérateurs de possibilités et de nécessité.

Notre première contribution est fondée sur la définition des opérateurs de dérivation et une nouvelle connexion de Galois en utilisant les opérateurs définis dans la théorie des possibilités qui sont l'opérateur de possibilité $(.)^{\Pi}$ et l'opérateur de nécessité $(.)^{N}$. Nous donnons d'abord une nouvelle caractérisation de la composition asymétrique des opérateurs de dérivation de la nécessité-possibilité $((.)^{N\circ\Pi})$. Il en résulte un nouveau type de concepts formels qui sont des paires "d'ouvert-fermé" appelés NΠ-paires. Les concepts formels organisés dans une hiérarchie (c'est-à-dire un ordre partiel), appelée NΠ-treillis, peuvent ensuite être extraits à partir d'un contexte formel. Nous exploitons le NΠ-treillis pour extraire des implications d'attributs disjonctives, à savoir des formules de la forme $a_1 \vee ... \vee a_n \longmapsto b_1 \vee ... \vee b_m$. En profitant de la structure algébrique de l'ensemble de toutes les NΠ-paires obtenues en utilisant la composition asymétrique $((.)^{N\circ\Pi})$, l'idée consiste à générer une base minimale d'implications d'attributs disjonctives. La minimalité est prouvée au moyen de "Pseudo-NΠ-Fermé" préalablement présentée dans cette thèse.

Notre deuxième contribution traite deux types de contextes formels, les contextes formels incomplets et les contextes formels incertains. En effet, il est largement admis que les contextes formels, peuvent être incomplets ou incertains dans les applications pratiques. Dans le cas d'un contexte formel incomplet, nous devons distinguer entre le cas de ne pas savoir si un objet possède un attribut ou non, et le cas où il est bien connu que l'objet ne possède pas cet attribut. Comme pour les contextes formels complets [94] et [25], il peut être utile de traiter les implications extraites de ces contextes formels incomplets. À cette fin, nous définissons ce que l'on appelle "implications certaines" qui tiennent dans tous les mondes possibles compatibles avec l'information incomplète et les "implications possibles" qui tiennent dans au moins une situation compatible avec l'information incomplète. Profitant de nos résultats précédents, une caractérisation des implications d'attributs possibles et certaines est proposée à la fois pour la sémantique conjonctive et disjonctive. Dans le cas d'un contexte formel incertain [43] [42], les cases sont remplies par des paires (α, β) où α est le degré de nécessité que l'objet ait l'attribut et β est le degré de nécessité que l'objet n'ait pas l'attribut avec $min(\alpha, \beta) = 0$. Nous proposons une approche qui nous permet d'obtenir, à partir d'un contexte formel incertain, des implications d'attributs à la fois pour la sémantique conjonctive et disjonctive. On calcule pour chaque implication son degré de certitude.

Notre troisième contribution apporte une proposition sur l'utilisation conjointe de l'ACF et des LDs. celle-ci consiste à appliquer nos résultats théoriques à des logiques de description (LDs) où nous allons considérer deux cas :

-Cas 1 : extraction des inclusions générales de concepts (GCIs) dans des environnements complets où on tient compte que de l'hypothèse du monde clos. Du moment que la base assertionnelle est complète, le contexte formel induit de cette ABox sera complètement renseigné (rempli). Nous allons proposer une approche qui va tenir compte des rôles et de la restriction existentielle afin d'extraire des GCIs exprimées en logique de description $\mathcal{EL}$. Nous allons proposer une autre approche qui permet de générer des GCIs disjonctive afin d'enrichir la base terminologique exprimée en logique de description $\mathcal{EL}$ et avoir des GCIs exprimées en logique de description $\mathcal{ELU}$.

-Cas 2 : extraction des GCIs dans des environnements incomplets où on tient compte de l'hypothèse du monde ouvert. Du moment que la base assertionnelle est incomplète, le contexte formel induit de cette ABox sera incomplet ce qui nécessite d'introduire une troisième valeur, notée " ?", pour exprimer l'ignorance, pour dire qu'on ne sait pas si un objet possède un attribut ou pas. Dans ce cas, tous les travaux qu'on trouve dans la littérature utilisent l'exploration d'attributs qui est l'un des outils de l'ACF qui permet de compléter la ABox afin d'avoir un environnement complet et cela en posant des questions à un expert du domaine. Dans ces approches, ils supposent que l'expert peut toujours répondre à toutes les questions. Cependant, notre approche repose sur des hypothèses différentes de ces travaux. En particulier, nous supposons que l'expert ne peut pas répondre à toutes les questions et, en conséquence, les connaissances obtenues après le processus d'exploration peuvent encore être incomplètes, ce qui conduit à deux types de GCIs, les GCIs certaines et les GCIs plausibles.

1.3 Plan de la thèse

Dans ce premier chapitre, nous avons présenté le contexte et la problématique de la thèse et nous avons résumé les différentes approches et contributions. Ce manuscrit est composé de six autres chapitres : les trois chapitres qui viennent après l'introduction apportent un état de l'art des domaines de recherche dans lesquels se situe notre travail ; les trois derniers chapitres comprennent les détails de nos approches et contributions.

ÉTAT DE L'ART

Le chapitre 2 porte sur l'analyse de concepts formels classique. Dans ce chapitre, nous commençons par donner les fondements mathématiques de la théorie de l'ACF. Nous donnons aussi certaines notions sur les structures algébriques de treillis et les concepts sous jacents de fermeture. Nous présentons aussi les implications d'attributs, la base minimale de Duquenne-Guigues ainsi que l'exploration d'attributs. Nous donnons un aperçu des principaux algorithmes développés dans le cadre de l'ACF, et évoquons les principales extensions et applications de l'ACF. Finalement, nous citons les principales approches qui ont tenté de considérer les différentes significations des valeurs d'un contexte formel.

Le chapitre 3 présente les principales approches qui permettent le traitement de l'information incertaine : la théorie des probabilités, la théorie des fonctions de croyance, et la théorie des possibilités. Un intérêt particulier sera porté sur la théorie des possibilités car elle constitue le cadre théorique que nous avons retenu dans notre travail de recherche. L'analyse des concepts formels et la théorie des possibilités sont deux cadres théoriques qui abordent différentes préoccupations dans le traitement de l'information. À savoir, l'ACF construit des concepts formels à partir d'une relation reliant les objets aux attributs qu'ils satisfont, tandis que la théorie des possibilités traite de la modélisation de l'incertitude épistémique (graduée). Cette différence d'attention explique pourquoi les deux cadres théoriques ont été développés indépendamment depuis longtemps. Cependant, on va présenter l'analogie formelle entre l'ACF et la théorie des possibilités. Les deux théories s'appuient fortement sur la comparaison des ensembles. Les quatre fonctions définies dans la théorie des possibilités déterminent effectivement toutes les positions relatives possibles de deux ensembles.

Le chapitre 4 présente les notions de base des LDs et rappelle la syntaxe et la sémantique des LDs. De manière plus approfondie, nous présentons la subsomption des inférences standard et la vérification de l'instance, et l'inférence non-standard. Ce chapitre présente aussi la modélisation des connaissances d'un domaine avec les LDs qui se réalise en deux niveaux. Le premier, le niveau terminologique ou TBox, décrit les connaissances générales d'un domaine alors que le second, le niveau factuel ou ABox, représente une configuration précise. Une TBox comprend la définition des concepts et des rôles, alors qu'une ABox décrit les individus en les nommant et en spécifiant en termes de concepts et de rôles, des assertions qui portent sur ces individus nommés. Nous avons jugé utile de présenter aussi un état de l'art sur les travaux portant sur l'utilisation conjointe de l'ACF et la LD. Nous classons ces travaux en deux catégories : la première catégorie concerne les travaux qui enrichissent l'ACF en empruntant des propriétés plus complexes semblables à celles des constructeurs de concepts dans les LDs ; la deuxième catégorie concerne les travaux qui emploient les méthodes de l'ACF pour résoudre certains problèmes rencontrés dans la représentation de connaissances à l'aide de LDs.

CONTRIBUTIONS :

Le chapitre 5 présente notre première contribution qui propose d'utiliser une nouvelle connexion de Galois basée sur les opérateurs définis dans la théorie des possibilités qui sont l'opérateur de possibilité $(.)^{\Pi}$ et l'opérateur de nécessité $(.)^{N}$. Nous donnons d'abord une nouvelle caractérisation de la composition asymétrique ($(.)^{N\circ\Pi}$) de ces opérateurs. Il en résulte un nouveau type de concepts formels (appelées NΠ-paires) qui sont organisés dans une hiérarchie à l'aide d'un ordre partiel afin de former un NΠ-treillis. Nous exploitons le NΠ-treillis pour extraire des implications d'attributs disjonctives, à savoir des formules de la forme $a_1 \vee ... \vee a_n \longmapsto b_1 \vee ... \vee b_m$. Nous avons aussi caractérisé la base minimale d'implications d'attributs disjonctives. La minimalité est prouvée au moyen de "Pseudo-NΠ-Fermé" présentée dans cette thèse. Les résultats de cette partie ont fait l'objet des articles [3–5]

Le chapitre 6 est réservé à notre deuxième contribution. Dans ce chapitre nous considérons les contextes formels incomplets et les contextes formels incertains.

Dans le cas d'un contexte formel incomplet, nous distinguons entre le cas de ne pas savoir si un objet possède un attribut ou non, et le cas où il est bien connu que l'objet ne possède pas cet attribut. Ainsi, considérer les implications extraites de ces contextes formels incomplets nous amène à proposer, de manière originale, des "implications certaines" qui tiennent dans tous les mondes possibles compatibles avec l'information incomplète et des "implications possibles" qui tiennent dans au moins une situation compatible avec l'information incomplète. Cette proposition est établie à la fois avec la sémantique conjonctive et disjonctive.

Dans la cas d'un contexte formel incertain, les cases sont remplies par des paires (α,β) où α est le degré de nécessité que l'objet ait l'attribut et β est le degré de nécessité que l'objet n'ait pas l'attribut avec $min(\alpha,\beta) = 0$. Nous proposons une approche qui nous permet d'obtenir, à partir d'un contexte formel incertain, des implications d'attributs à la fois pour la sémantique conjonctive et disjonctive. On calcule pour chaque implication son degré de certitude. Les résultats de cette partie ont fait l'objet des articles [3–5]

Le chapitre 7 présente notre troisième contribution qui met en évidence l'intérêt d'utiliser nos résultats théoriques aux logiques de description (LDs) afin de générer des GCIs à partir d'une ABox. Nous avons distingué deux cas : le premier cas concerne l'extraction des GCIs dans des environnements complets où on tient compte que de l'hypothèse du monde clos ; le deuxième cas concerne l'extraction des GCIs dans des environnements incomplets où on tient compte de l'hypothèse du monde ouvert, et lorsque le connecteur de négation est explicitement utilisé dans la ABox en utilisant les implications d'attributs certaines et les implications d'attributs possibles. Les résultats de cette partie ont fait l'objet des articles [2, 5]

Finalement, nous concluons en présentant les objectifs atteints dans ce travail et en évoquant de nouvelles perspectives pour de futures recherches.

CHAPITRE 2

L'Analyse de Concepts Formels

Sommaire

2.1 Introduction

L'Analyse de Concepts Formels (que nous désignerons par ACF dans la suite de ce document), désigné par le terme anglophone Formal Concept Analysis (FCA), a été introduite par Wille en 1982 [112], puis consolidée mathématiquement par Ganter et Wille en 1999 [66]. Il s'agit de la restructuration de la théorie des treillis définie par G. Birkhoff en adéquation avec la philosophie de la pensée humaine définie par C.S. Peirce. L'ACF est une théorie qui constitue un pont entre les mathématiques et la représentation de connaissances. Elle vise à identifier des clusters de connaissance qui sont des paires de sous-ensembles ($\{objets\}$, $\{attributs\}$), appelés concepts formels, induits à partir d'une relation binaire entre un ensemble d'objets et un ensemble d'attributs (propriétés). L'ACF vise aussi à ordonner ces clusters sous forme d'un treillis. Dans ce chapitre, nous dresserons un état de l'art qui couvre à la fois l'origine et le contexte philosophique de l'ACF, une présentation intuitive de l'ACF sans pour autant introduire de formalisme mathématique et nous consacrerons par la suite une section complète pour la présentation théorique de l'ACF sur la base de fondements mathématiques (algébriques), nous passons en revue les algorithmes de construction de treillis.

Les implications d'attributs sont une autre façon de structurer les données dans l'ACF. Nous les présentons dans la section 2.6. La base de Duquenne-Guigues, qui est un ensemble minimal d'implications d'attributs d'un contexte formel, est présentée dans la section 2.6.2. Nous montrons comment cela peut être déterminé à l'aide de l'algorithme Next-Closure. Les implications jouent également un rôle important dans le formalisme d'acquisition de connaissances interactif (assisté par des experts) de l'ACF qui s'appelle exploration d'attributs, qu'on a présenté dans la section 2.6.3. Dans la section 2.8 nous avons présenté les différentes extensions de l'ACF, et les applications de l'ACF sont données dans la section 2.9.

2.2 Origine et contexte philosophique

L'Analyse de Concepts Formel (ACF) [66, 112] a été présentée comme un domaine de mathématiques appliquées qui consiste à restructurer la théorie des treillis [21] afin de faciliter son utilisation dans des applications du monde réel et de permettre l'interprétation de ses notions en dehors du cadre théorique aussi bien par des mathématiciens que par des non-mathématiciens. L'objectif à travers la mise en place de l'ACF est d'atteindre une théorie structurée qui expose les pensées formelles selon des interprétations significatives et permettre ainsi des communications et des discussions critiques de leurs contenus [114]. Pour cela l'ACF a été centrée autour de la notion de concept qui, du point de vue philosophique, est considéré comme l'unité de base de la pensée humaine. De manière informelle, un concept peut être défini comme un groupement d'individus et de leurs propriétés communes. Toujours en liaison avec le point de vue philosophique, en particulier celui de Charles Sanders Peirce (un sémiologue et philosophe américain) qui considère que dans tout processus de raisonnement ou d'argumentation, on ne peut étudier qu'une part de la réalité, l'ACF extrait les concepts à partir de contextes restreints [35]. De tels contextes, appelés contextes formels, constituent le point de départ de l'ACF et sont définis dans la section suivante.

2.3 Présentation intuitive de l'analyse de concepts formels

L'analyse de concepts formels se base sur la notion de concepts formels qui est considéré du point de vue psychologique comme l'unité de base de la pensée humaine. Le concept peut être défini comme un ensemble d'objets et de leurs propriétés communes. Ces concepts formels sont extraits à partir d'une relation binaire définie entre un ensemble d'objets et un ensemble d'attributs (propriétés). Cette relation est appelée contexte formel qui peut être représenté sous la forme d'un tableau où les lignes correspondent aux objets et les colonnes correspondent aux attributs. Les cases du tableau sont remplies comme suit : si le $i^{\text{ème}}$ objet satisfait le $j^{\text{ème}}$ attribut alors la case intersection de la ligne i et la colonne j contient $\times$, sinon la case est vide. Dans l'exemple suivant on reprend le contexte formel donné dans [107].

Exemple 1. *Soit un ensemble d'animaux {lion, rouge-gorge, aigle, lièvre, autruche} décrit par rapport à certaines de leurs attributs {prédateur, vole, ovipare, mammifère}. Le contexte formel noté $\mathcal{K}$, avec la relation binaire $\mathcal{R}$, est représenté sous forme d'une table avec en lignes les animaux (correspondant aux objets) et en colonnes les attributs, telle que si l'objet x_i vérifie (resp. ne vérifie pas) l'attribut a_j alors la cellule $\mathcal{R}_{ij}$ est marquée par une croix (resp. reste vide).*

TABLE 2.1 – Représentation du contexte formel $\mathcal{K}$.

$\mathcal{R}$	Prédateur	Vole	Ovipare	Mammifère
Lion	×			×
Rouge-gorge		×	×	
Aigle	×	×	×	
Autruche			×	
Lièvre				×

Pour expliquer la notion de concept formel, nous considérons tous les attributs du rouge-gorge {vole, ovipare} et posons-nous la question : quels sont les animaux satisfaisant ces propriétés ? Nous obtenons alors l'ensemble $X = \{aigle, rouge - gorge\}$. Nous constatons que l'ensemble X est l'ensemble maximal des animaux satisfaisant tous les attributs de l'ensemble $A = \{vole, ovipare\}$. Il résulte que X est l'ensemble de tous les animaux vérifiant tous les attributs de A, et A est l'ensemble de tous les attributs vérifiés par tous les animaux de X. La paire (X, A) est appelée concept formel. L'élément X est appelé extension tandis que A est appelé intension.

Entre les concepts formels, il y a une relation d'ordre partiel c.à.d. la relation "Sous-concept, Super-concept". Etant donné deux concepts formels (X_1, A_1) et (X_2, A_2), on dit que (X_1, A_1) est un sous-concept de (X_2, A_2), (dualement (X_2, A_2) est un super-concept de (X_1, A_1)) si l'extension de (X_1, A_1) est un sous ensemble de l'extension de (X_2, A_2). c.à.d. $X_1 \subseteq X_2$ (dualement : l'intension de (X_1, A_1) est un sur-ensemble de l'intension de (X_2, A_2). c.à.d. $A_2 \subseteq A_1$).

Exemple 1 (suite)
Etant donné les deux concepts formels suivants : ({Aigle},{prédateur, vole, ovipare}) et ({Aigle, Rouge-gorge},{vole, ovipare}). ({Aigle},{prédateur, vole, ovipare}) est un sous concept de ({Aigle, Rouge-gorge},{vole, ovipare}). Dualement, ({Aigle, Rouge-gorge},{vole, ovipare}) est un super-concept de ({Aigle}, {predateur,vole,ovipare}). L'extension {Aigle} du sous-concept est un sous ensemble de l'extension {Aigle, Rouge-gorge} du super-concept. De la même manière l'intension {prédateur, vole, ovipare} du sous-concept est un sur-ensemble de l'intension {vole, ovipare} du super-concept. L'ensemble de tous les concepts formels présente une propriété algébrique importante : il constitue un treillis complet. Le treillis associé au contexte formel de l'exemple 1 est présenté dans la figure 2.1. Les sommets correspondent aux concepts formels et les arêtes à l'ordre partiel entre ces concepts formels.

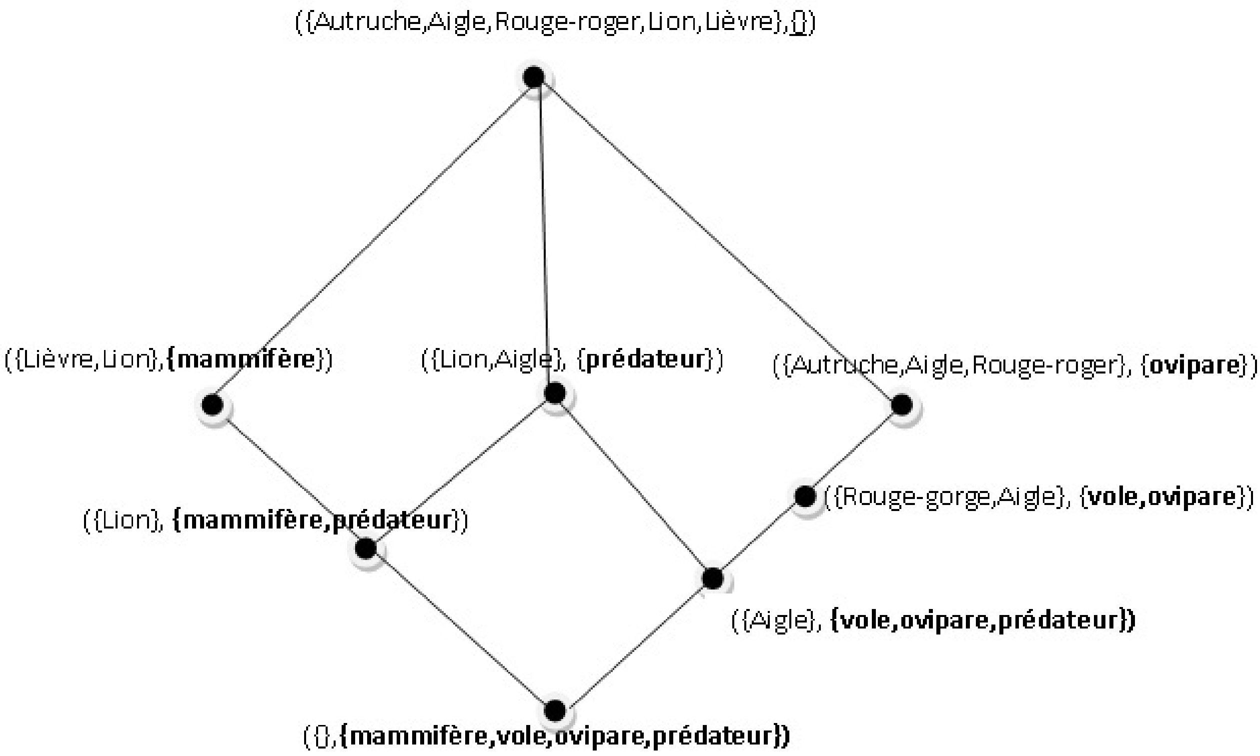

FIGURE 2.1 – Treillis de concepts formels

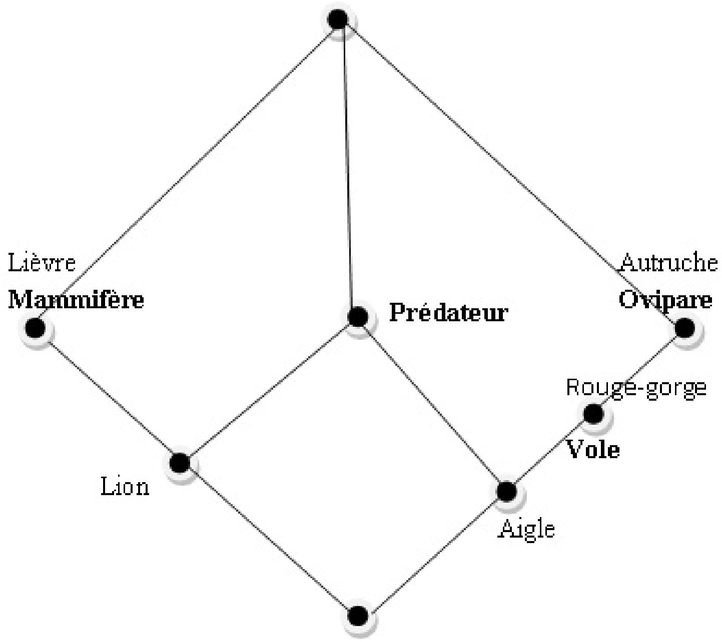

FIGURE 2.2 – Treillis de concepts formels simplifié

La figure 2.2 présente le treillis réduit associé au treillis de la figure 2.1. Afin de ne pas surcharger le graphe, seules les intensions et extension réduites ont été représentées. L'intension réduite d'un concept formel contient seulement les attributs qui n'apparaissent pas dans les intensions des concepts formels supérieurs. Réciproquement, l'extension réduite d'un concept formel contient seulement les objets qui n'apparaissent pas dans les extensions des concepts formels inférieurs. La lecture de ce treillis ce fait comme suit : les attributs sont placés au plus haut dans le treillis et à chaque fois qu'un nœud N est étiqueté par un attribut m, tous les descendants de N dans le treillis héritent l'attribut m. De façon duale, les objets sont placés au plus bas dans le treillis et à chaque fois qu'un nœud N est étiqueté par un objet g , g est hérité vers le haut, et tous les ancêtres de N le partagent. Ainsi l'extension X d'un concept formel (X, A) associé au nœud N est obtenue en considérant tous les objets qui apparaissent sur les descendants de nœud N dans le treillis et son intension A est obtenue en considérant tous les attributs qui apparaissent sur les ancêtres de nœud N dans le treillis.

2.4 Rappels mathématiques

Définition 1 (Relation binaire)**.**

Une relation binaire $\mathcal{R}$ entre deux ensembles $\mathcal{O}$ et $\mathcal{P}$ est un ensemble de couples d'éléments (x, a) tels que $x \in \mathcal{O}$ et $a \in \mathcal{P}$, autrement dit un sous ensemble du produit Cartésien $\mathcal{O} \times \mathcal{P}$. $(x, a) \in \mathcal{R}$ (aussi noté par $x\mathcal{R}a$) signifie que l'élément x est en relation $\mathcal{R}$ avec l'élément a.

Définition 2 (Relation d'ordre)**.**

Une relation binaire $\mathcal{R}$ sur un ensemble E est dite relation d'ordre partiel (ou simplement relation d'ordre) sur E si elle vérifie les conditions suivantes pour tout $x, y, z \in E$:

- $(x, x) \in \mathcal{R}$ *($\mathcal{R}$ est réflexive)*
- *si* $(x, y) \in R$ *et* $(y, x) \in \mathcal{R}$ *alors* $x = y$ *($\mathcal{R}$ est antisymétrique)*
- *si* $(x, y) \in \mathcal{R}$ *et* $(y, z) \in \mathcal{R}$ *alors* $(x, z) \in \mathcal{R}$ *($\mathcal{R}$ est transitive)*

Une relation d'ordre $\mathcal{R}$ est souvent notée par $\leq$.

Définition 3 (Ensemble ordonné)**.**

Un ensemble partiellement ordonné (ou simplement ensemble ordonné) est un couple $(E, \leq)$ où E est un ensemble et "$\leq$" est une relation d'ordre sur E.

Dans un ensemble ordonné $(E, \leq)$, deux éléments x et y de E sont dits comparables lorsque $x \leq y$ ou $y \leq x$, autrement ils sont dits incomparables. Un sous ensemble de $(E, \leq)$ dans lequel tous les éléments sont comparables est appelé chaîne. Un sous ensemble de $(E, \leq)$ dans lequel tous les éléments sont incomparables est appelé anti-chaîne.

Définition 4 (Majorant, minorant, supremum, infimum)**.**

Soient $(E, \preceq)$ un ensemble ordonné et S un sous ensemble de E. Un élément $a \in E$ est dit majorant de S lorsque $a \succeq s \ \forall s \in S$. De façon duale, $a \in E$ est dit minorant de S lorsque $a \preceq s \ \forall s \in S$.

Le plus petit majorant (respectivement plus grand minorant) de S, s'il existe, est appelé supremum ou borne supérieure (respectivement infimum ou borne inférieure) de S et noté $\bigvee S$ (respectivement $\bigwedge S$).

Dans le cas où $S = \{x, y\}$, $\bigvee S$ et $\bigwedge S$ sont aussi notés par $x \vee y$ et $x \wedge y$ respectivement. Dans tout ensemble ordonné, lorsque le supremum (respectivement l'infimum) existe, il est unique.

Définition 5 (Treillis, treillis complet)**.**

Un treillis est un ensemble partiellement ordonné $(E, \preceq)$ tel que $x \vee y$ et $x \wedge y$ existent pour tout couple d'éléments $x, y \in E$. Un treillis est dit complet si $\bigvee S$ et $\bigwedge S$ existent pour tout sous ensemble S de E. En particulier, un treillis complet admet un élément maximal (top) noté par $\top$ et un élément minimal (bottom) noté par $\bot$.

Définition 6 (Opérateur de fermeture)**.**

Un opérateur de fermeture défini sur un ensemble partiellement ordonné $(E, \preceq)$ est une fonction $\Psi : 2^E \longrightarrow 2^E$ satisfaisant $\forall\ U, V \in E$:

- $U \preceq \Psi(U)$
- $U \preceq V \Longrightarrow \Psi(U) \preceq \Psi(V)$
- $\Psi(\Psi(U)) = \Psi(U)$

Définition 7 (Système de fermeture)**.**

Un système de fermeture (dit aussi système de fermés) sur un ensemble E est un ensemble de parties de E contenant E et fermé pour l'intersection. Formellement, $\mathfrak{U} \subseteq 2^E$ est un système de fermeture si les conditions suivantes sont vérifiées :

- $E \in \mathfrak{U}$
- *si $V \subseteq \mathfrak{U}$ alors $\bigcap V \in \mathfrak{U}$ (l'intersection d'un ensemble arbitraire de fermés de V appartient aussi à $\mathfrak{U}$)*

Les deux notions, opérateur de fermeture et système de fermeture, sont fortement liées. En effet, tout système de fermeture peut être considéré comme l'ensemble de tous les fermés d'un opérateur de fermeture. Inversement, il est possible de définir un opérateur de fermeture sur tout système de fermeture.

Définition 8 (Opérateur d'ouverture)**.**

Un opérateur d'ouverture défini sur un ensemble partiellement ordonné $(E, \preceq)$ est une fonction $\Phi : 2^E \longrightarrow 2^E$ satisfaisant $\forall\ U, V \in E$:

- $\Phi(U) \leq U$
- $U \leq V \Longrightarrow \Phi(U) \leq \Phi(V)$
- $\Phi(\Phi(U)) = \Phi(U)$

Définition 9 (Connexion de Galois)**.**
Soient $\phi : P \to Q$ et $\psi : Q \to P$ deux applications entre deux ensembles ordonnés $(P, \preceq_P)$ et $(Q, \preceq_Q)$. ϕ et ψ forment une connexion de Galois entre $(P, \preceq_P)$ et $(Q, \preceq_Q)$ si et seulement si les conditions suivantes sont vérifiées pour tous $p, p_1, p_2 \in P$ et $q, q_1, q_2 \in Q$:

- *si $p_1 \preceq_P p_2$ alors $\phi(p_2) \preceq_Q \phi(p_1)$*
- *si $q_1 \preceq_Q q_2$ alors $\psi(q_2) \preceq_P \psi(q_1)$*
- *$p \preceq_P \psi(\phi(p))$ et $q \preceq_Q \phi(\psi(q))$*

Les conditions données dans la définition précédente sont équivalentes à la formule suivante :

$$p \preceq_P \psi(q) \Leftrightarrow q \preceq_Q \phi(p)$$

2.5 Théorie de l'analyse de concepts formels

L'ACF fournit un cadre théorique pour l'apprentissage de la hiérarchie de concepts formels. Cet apprentissage s'effectue à partir d'un contexte formel.

2.5.1 Contexte formel

Un contexte formel est un triplet $\mathcal{K} = (\mathcal{O}, \mathcal{P}, \mathcal{R})$ où $\mathcal{O}$ est un ensemble d'objets, $\mathcal{P}$ est un ensemble d'attributs (propriétés) et $\mathcal{R}$ est une relation binaire entre $\mathcal{O}$ et $\mathcal{P}$ appelée relation d'incidence de $\mathcal{K}$ et vérifiant $\mathcal{R} \subseteq \mathcal{O} \times \mathcal{P}$. Un couple $(x, a) \in \mathcal{R}$ (noté aussi $x\mathcal{R}a$) signifie que l'objet $x \in \mathcal{O}$ possède (satisfait) l'attribut $a \in \mathcal{P}$.

Un exemple de contexte formel reliant un ensemble d'objets $x_1, x_2, x_3, x_4, x_5, x_6$ à un ensemble d'attributs a_1, a_2, a_3, a_4, a_5 est illustré à travers la table 2.2. Par exemple, dans le contexte formel représenté par la table 2.2, l'objet x_1 possède les attributs a_3, a_4, a_5 et l'objet x_3 possède les attributs a_2, a_3.

TABLE 2.2 – Contexte formel $\mathcal{K}_S$.

$\mathcal{R}$	a_1	a_2	a_3	a_4	a_5
x_1			×	×	×
x_2	×				×
x_3		×	×		
x_4	×	×	×	×	
x_5			×	×	×
x_6			×		

2.5.2 Opérateurs de dérivation de Galois

Le paradigme de l'ACF [66] repose classiquement sur la paire d'opérateurs ensemblistes $(.)^\Delta : 2^\mathcal{O} \rightarrow 2^\mathcal{P}$ et $(.)^\Delta : 2^\mathcal{P} \rightarrow 2^\mathcal{O}$, (plus généralement appelés opérateurs de dérivation de Galois dans la littérature, désigné ici par $(.)^\Delta_\mathcal{K}$). Soit $\mathcal{K} = (\mathcal{O}, \mathcal{P}, \mathcal{R})$ un contexte formel, étant donnés deux ensembles $X \in 2^\mathcal{O}$ et $A \in 2^\mathcal{P}$, ces deux opérateurs sont définis comme suit :

$A^\Delta_\mathcal{K}$ est l'ensemble des objets possédant tous les attributs de A :

$$\begin{aligned} A^\Delta_\mathcal{K} \quad &= \{x \in \mathcal{O} \mid \forall a \in \mathcal{P}(a \in A \Rightarrow x\mathcal{R}a)\} \\ &= \{x \in \mathcal{O} \mid A \subseteq \mathcal{R}(x)\} \\ &= \textstyle\bigcap_{a \in A} \mathcal{R}(a) \end{aligned}$$

$X^\Delta_\mathcal{K}$ est l'ensemble des attributs communs à tous les objets de X :

$$\begin{aligned} X^\Delta_\mathcal{K} \quad &= \{a \in \mathcal{P} \mid \forall x \in \mathcal{O}(x \in X \Rightarrow x\mathcal{R}a)\} \\ &= \{a \in \mathcal{P} \mid X \subseteq \mathcal{R}(a)\} \\ &= \textstyle\bigcap_{x \in X} \mathcal{R}(x) \end{aligned}$$

Dans ce qui suit, lorsqu'il n'y a pas d'ambiguïté sur le contexte formel $\mathcal{K}$, nous supprimerons l'indice $(.)_\mathcal{K}$ dans $A^\Delta_\mathcal{K}$ et $X^\Delta_\mathcal{K}$ et nous écrirons directement A^Δ, X^Δ pour plus de simplicité.

Il est clair que A^Δ correspond à l'ensemble des objets qui satisfont tous les attributs de A : il suffit à un objet de posséder tous les attributs de A pour qu'il se trouve dans A^Δ (D'où le nom de "l'opérateur de suffisance" utilisé par Düntsch et Orlowska [58]). De même, X^Δ correspond à l'ensemble des attributs qui sont satisfaits par tous les objets de X. En particulier, $\{x\}^\Delta$ est l'ensemble des attributs (connus) de l'objet x.

Les propriétés suivantes des opérateurs de dérivation de Galois sont aisément établies.

Lemme 1. [113] *(Propriétés des opérateurs de dérivation de Galois). Soit $\mathcal{K} = (\mathcal{O}, \mathcal{P}, \mathcal{R})$ un contexte formel. Soit $X_1, X_2 \subseteq \mathcal{O}$ et $A_1, A_2 \subseteq \mathcal{P}$. Alors :*

- $X_1 \subseteq X_2$ *implique* $X_2^{\Delta} \subseteq X_1^{\Delta}$
- $A_1 \subseteq A_2$ *implique* $A_2^{\Delta} \subseteq A_1^{\Delta}$
- $X_1 \subseteq X_1^{\Delta\Delta}$
- $A_1 \subseteq A_1^{\Delta\Delta}$

2.5.3 Concept formel

Un concept formel de $\mathcal{K} = (\mathcal{O}, \mathcal{P}, \mathcal{R})$ est une paire d'ensembles (X, A) avec $X \subseteq \mathcal{O}, A \subseteq \mathcal{P}$ tel que

$$X^{\Delta} = A \text{ et } A^{\Delta} = X.$$

Ces ensembles sont fermés dans le sens où $A^{\Delta\Delta} = A$ et $X^{\Delta\Delta} = X$. Par exemple, $(\{x_3, x_4\}, \{a_2, a_3\})$ est un concept formel de $K_{\mathcal{S}}$. L'ensemble de tous les concepts formels sera noté par $\mathfrak{B}_{\Delta\Delta}$ et on notera par $\mathfrak{B}_{\Delta\Delta}(Int)$ l'ensemble de toutes les intensions de $\mathfrak{B}_{\Delta\Delta}$.

Nous pouvons remarquer qu'un concept formel correspond à un rectangle maximal du contexte forme tel que tout objet de l'extension vérifie tous les attributs de l'intension. Il est important de noter que cette notion de rectangle maximal est indépendante de l'ordre des lignes et des colonnes. Ces ensembles maximaux d'objets et d'attributs sont à la base de la définition d'un concept formel. Un sous-ensemble $A \subseteq \mathcal{P}$ est l'intension d'un concept formel dans $\mathfrak{B}_{\Delta\Delta}$ si et seulement si $A^{\Delta\Delta} = A$ (A est l'ensemble maximal de propriétés dans $\mathcal{P}$ communes aux objets ayant les propriétés dans A) et, de façon duale, un sous ensemble X de $\mathcal{O}$ est l'extension d'un concept formel dans $\mathfrak{B}_{\Delta\Delta}$ si et seulement si $X^{\Delta\Delta} = X$ (X est l'ensemble maximal d'objets dans $\mathcal{O}$ ayant les propriétés communes aux objets de X).

2.5.4 Treillis de Galois

L'ensemble de tous les concepts formels (noté par $\mathfrak{B}_{\Delta\Delta}$), équipé d'une relation d'ordre partiel $\preceq$ définie tel que $(X_1, A_1) \preceq (X_2, A_2)$ si $X_1 \subseteq X_2$ (équivalent à $A_2 \subseteq A_1$), forme un treillis de Galois (appelé aussi treillis complet ou treillis de concepts formels) qu'on notera par $\mathfrak{L}^{\mathcal{K}}$.

La relation $\preceq$ s'appuie sur deux inclusions duales, entre ensembles d'objets et ensembles d'attributs, et peut ainsi être interprétée comme une relation de généralisation/spécialisation entre les concepts formels. Un concept formel est plus général qu'un autre concept s'il contient plus d'objets dans son extension. En contre partie, les attributs partagés par ces objets sont réduits. De façon duale, un concept formel est plus spécifique qu'un autre s'il contient moins d'objets dans son extension. Ces objets ont plus d'attributs en commun. La figure 2.3 illustre le treillis correspondant au contexte formel donné dans la table 2.2.

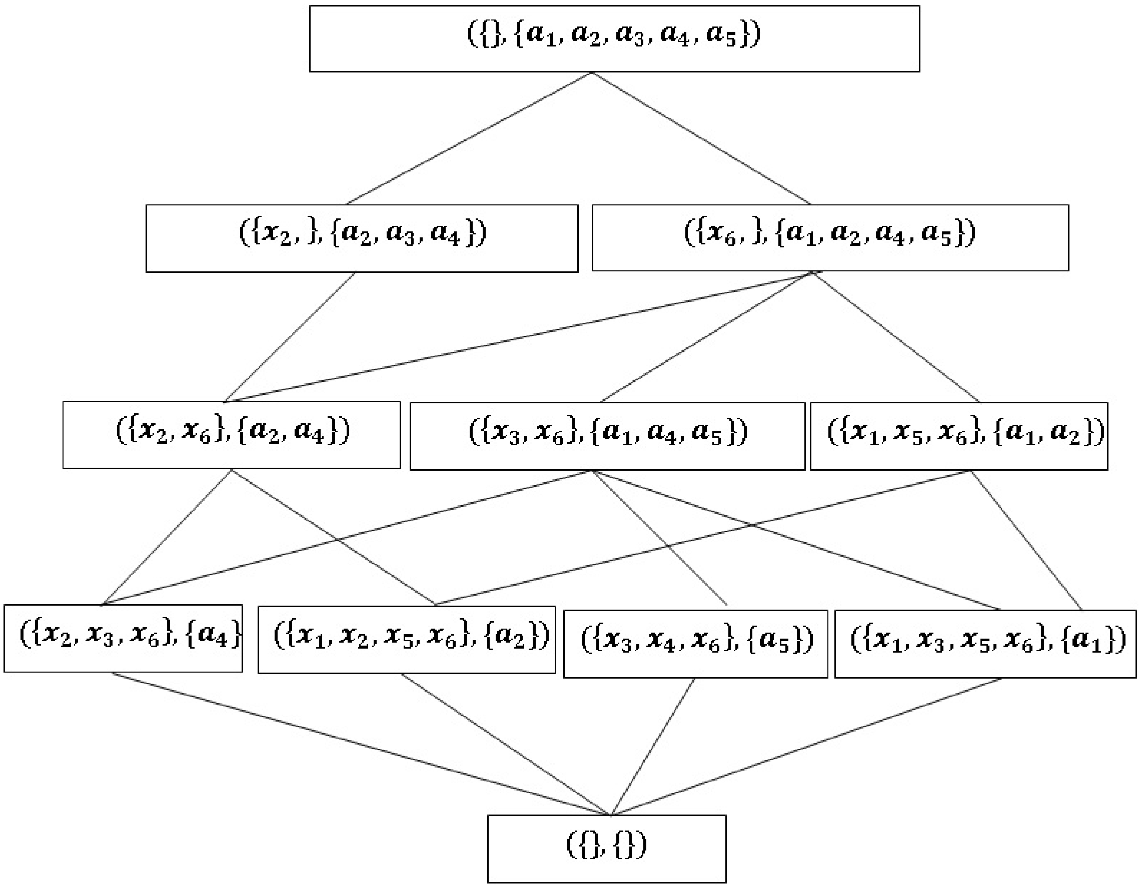

FIGURE 2.3 – Treillis de Galois

Le théorème fondamental de Ganter et Wille [66] donne le supremum (noté $\coprod$) et l'infimum (noté $\prod$) du treillis de concepts formels $\mathfrak{L}^{\mathcal{K}}$ comme suit :

$$\coprod_{j\in J}(X_j, A_j) = \Big(((\bigcup_{j\in J} X_j)^{\Delta})^{\Delta}, \bigcap_{j\in J} A_j\Big).$$

$$\prod_{j\in J}(X_j, A_j) = \Big(\bigcap_{j\in J} X_j, ((\bigcup_{j\in J} A_j)^{\Delta})^{\Delta}\Big);$$

Un treillis de concepts formels est une représentation isomorphe à un contexte formel qui met en avant les groupements possibles entre objets et attributs ainsi que les relations d'inclusion entre ces groupements. De plus, la représentation graphique du treillis de concepts formels, sous la forme d'un diagramme de Hasse, facilite la compréhension et l'interprétation de la relation entre les objets et les attributs d'une part et entre objets ou attributs d'autre part. L'avantage de cette représentation est qu'à partir d'un treillis de concepts formels il est toujours possible de retrouver le contexte formel correspondant et inversement.

La construction du treillis de concepts formels d'une relation binaire donnée peut être décomposée en trois parties [72] :
1. L'énumération des rectangles maximaux (les concepts formels),
2. La recherche de la relation d'ordre partiel entre ces concepts formels,
3. La construction du diagramme de HASSE correspondant au treillis.

Plusieurs approches ont été proposées pour la construction de treillis de concepts formels : Kuznetsov [77], Chein [31], Norris [92], Ganter [63, 65], Bordat [22], et ont fait l'objectif de plusieurs comparaisons [89]. Dans leur article, Kuznetsov et Obiedkov [78] analysent plusieurs algorithmes de construction de treillis de concepts formels. Ils présentent une étude de leurs complexités théoriques et une comparaison expérimentale sur des jeux de données artificiels. Les auteurs font des recommandations en fonction de la nature du contexte. De plus, il existe plusieurs outils qui permettent d'éditer des contextes formels, et de construire le treillis de concepts formels associé : ConExp et Galicia sont deux outils libres couramment utilisés. Une liste plus complète d'autres logiciels peut être consultée sur www.upriss.org.uk/fca.

Ces algorithmes ont une complexité polynomiale (au mieux quadratique dans [93]) par concept généré, et dépendent donc de la taille du treillis de Galois. La taille du treillis est bornée par $2^{|O+I|}$ dans le pire des cas, et par $|O+I|$ dans le meilleur des cas. Des études de complexité en moyenne sont extrêmement difficiles à mener, car la taille du treillis dépend à la fois de la taille des données à classifier, mais aussi de leur organisation et de leur diversité. Notons cependant que sa taille reste raisonnable en pratique, comme l'illustrent les expérimentations qui en ont déjà été faites [90]. De récents travaux [69] proposent un algorithme générique permettant à la fois d'unifier les algorithmes existants dans un même cadre, mais aussi de les comparer en fonction des propriétés des données, et par conséquent du treillis.

Nous allons à présent décrire plus précisément l'algorithme de construction du treillis de Galois le plus connu et le plus utilisé : NextClosure [63] qui calcule les concepts dans un ordre lectique (proche de l'ordre lexicographique), puis les ordonne par inclusion.

Définition 10. *L'ordre lectique se définit pour un ensemble $S = \{x_1, ..., x_n\}$ d'éléments indicés. Un fermé A est inférieur lectiquement à un fermé B si l'élément de plus petit indice i qui distingue A et B appartient à B, on note alors :*

$$A <_i B \text{ ssi } \exists x_i \in B \backslash A \text{ tel que } A|i = B|i$$

avec $A|i = A \cap x_1, x_2, ...x_{i-1}$, restriction de A aux $(i-1)$ premiers éléments de S.

Par exemple, pour $S = a, b, c$, on aura $ac <_1 bc, ab <_2 ac$ et $ab <_3 abc$. Clairement, l'ordre lectique définit un ordre total sur les parties de S et étend l'ordre d'inclusion i.e. si $A \subseteq B$ alors il existe $i \leq n$ tel que $A \leq_i B$.

Lemme 2. [63] *(Next-Closure). Soit X un ensemble fini. Soit cl un opérateur de fermeture sur X et $A \subseteq X$. S'il existe un plus petit ensemble fermé qui est plus grand que A alors :*

$$\mathfrak{cl}(\{x\} \cup \{y \in A \mid y < x\})$$

*Où x est le plus grand élément de X pour lequel $A <_x \mathfrak{cl}(\{x\} \cup \{y \in A \mid y < x\})$ est valide. Nous appelons $\mathfrak{cl}(\{x\} \cup \{y \in A \mid y < x\})$ la prochaine fermeture (**next closure**) de A.*

Next-Closure peut calculer tous les concepts formels d'un contexte. C'est une conséquence facile du lemme 1 d'où l'on obtient un opérateur de fermeture en appliquant consécutivement les deux opérateurs de dérivation de l'ACF. Une autre conséquence du lemme 1 est le résultat suivant. Il montre que les intensions des concepts formels d'un contexte $\mathcal{K}$ sont exactement les ensembles de la forme $B^{\Delta\Delta}$, où $B \subseteq \mathcal{P}$, i. e. Les intensions de concept sont les ensembles fermés de l'opérateur de fermeture $(.)^{\Delta\Delta}$. Par conséquent, ils peuvent être énumérés en utilisant Next-Closure.

L'algorithme 1 montre comment Next-Closure peut être utilisé pour trouver tous les concepts formels d'un contexte formel. Dans la ligne 6, l'ensemble Next-Closure de A_{i+1} est calculé en utilisant le lemme 2. Nous calculons d'abord :

$$x_{max} := max\{x \in \mathcal{P} \mid A_i <_x (\{x\} \cup \{y \in A_i \mid y < x\})^{\Delta\Delta}\}$$

et nous obtenons :

$$A_{i+1} := (\{x_{max}\} \cup \{y \in A_i \mid y < x_{max}\})^{\Delta\Delta}\}$$

Algorithme 1. [63] Next-Closure pour énumérer tous les concepts formels

1 : $\mathcal{K} := (\mathcal{O}, \mathcal{P}, \mathcal{R})$ {Entrée : contexte formel $\mathcal{K}$}
2 : $A_0 := (\emptyset)^{\Delta\Delta}$ {Intension de concept lectiquement plus petite}
3 : $C_0 := \{(A_0^{\Delta}, A_0)\}$ {Initialiser l'ensemble des concepts formels}
4 : $i := 0$
5 : **tant que** $A_i \neq \mathcal{P}$ **faire**
6 : A_{i+1} :=Sous-ensemble de $\mathcal{P}$ lectiquement plus petit qui est
        - lectiquement plus grande que A_i, et
        - fermé par rapport à $(.)^{\Delta\Delta}$
7 : $C_{i+1} := C_1 \cup \{(A_{i+1}^{\Delta}, A_{i+1})\}$
8 : $i := i + 1$
9 : **fin tant que** ;
10 : **retourne** C_i

2.6 Implications d'attributs (conjonctives)

L'ACF s'est avérée être un moyen utile pour extraire des implications d'attributs (conjonctives) à partir des contextes formels. Ce paradigme est décrit à travers cette section.

2.6.1 Présentation générale

Dans le paradigme de l'analyse de concepts formels [66, 73], une implication d'attribut, notée $A \longmapsto B$, affirme une relation sûre entre deux sous-ensembles d'attributs A, B ($A, B \in 2^{\mathcal{P}}$). Les ensembles d'attributs A et B sont appelés, respectivement, prémisse et conclusion de l'implication d'attributs $A \longmapsto B$. La définition donnée ci-dessous [66] exprime la satisfaction d'une telle implication intentionnelle.

Définition 11. *Un sous-ensemble $M \subseteq \mathcal{P}$ satisfait une implication d'attributs $A \longmapsto B$ (noté par $M \models A \longmapsto B$) ssi $A \nsubseteq M$ ou $B \subseteq M$.*

Dans la définition ci-dessus, M peut être remplacé par $\{x\}^{\Delta}$ en utilisant un objet x satisfaisant tous les attributs dans M et en falsifiant d'autres attributs. La satisfaction d'une implication d'attributs peut également être exprimée sous forme extensionnelle [66] comme suit :

Définition 12. *Un contexte formel $\mathcal{K} = (\mathcal{O}, \mathcal{P}, \mathcal{R})$ satisfait une implication d'attributs $A \longmapsto B$, où $A, B \subseteq \mathcal{P}$ (denoté par $\mathcal{K} \models A \longmapsto B$) ssi pour chaque $x \in \mathcal{O}$, $\{x\}^{\Delta} \models A \longmapsto B$.*

A partir de cette définition, on peut remarquer qu'une implication d'attributs de la forme $A \longmapsto B$ est vérifiée dans $\mathcal{K}$ ssi $A^{\Delta} \subseteq B^{\Delta}$ ce qui est équivalent à $B \subseteq A^{\Delta\Delta}$.

La sémantique d'une telle implication d'attributs est que, pour tout objet $x \in \mathcal{O}$, si x satisfait les attributs de A, alors x va aussi satisfaire les attributs de B. Par exemple, dans le contexte formel $K_{\mathcal{S}}$ donné dans la table 2.2, tous les objets qui ont les attributs "a_3" et "a_5" à savoir "x_1 et x_5" ont également l'attribut "a_4". Une telle connaissance est représentée par l'implication d'attributs suivante :

$$\{a_3, a_5\} \longmapsto \{a_4\}$$

Soit $\mathcal{B}$ un ensemble d'implications d'attributs et on dénote par $\delta(\mathcal{B})$ l'ensemble de sous-ensembles de $\mathcal{P}$ qui satisfont chaque implication d'attributs dans $\mathcal{B}$, i.e., $\delta(\mathcal{B}) = \{M \subseteq \mathcal{P} \mid M \models \mathcal{B}\}$, où $M \models \mathcal{B}$ correspond à la satisfaction par M de chaque implication d'attributs dans $\mathcal{B}$.

Définition 13. [66] *L'ensemble d'implications d'attributs $\mathcal{B}$ est :*

- *valide pour le contexte formel $\mathcal{K} = (\mathcal{O}, \mathcal{P}, \mathcal{R})$ ssi ce contexte formel vérifie chaque implication d'attributs de $\mathcal{B}$.*
- *complet pour le contexte formel $\mathcal{K} = (\mathcal{O}, \mathcal{P}, \mathcal{R})$ ssi chaque sous-ensemble de $\mathcal{P}$ qui satisfait $\mathcal{B}$ est une intension de contexte formel : $\delta(\mathcal{B}) \subseteq \mathfrak{B}_{\Delta\Delta}(Int)$.*
- *non redondant si aucune implication d'attributs $A \longmapsto B \in \mathcal{B}$ ne pourra être obtenu à partir de l'ensemble $\mathcal{B}\backslash\{A \longmapsto B\}$, i.e., Il existe un sous-ensemble de $\mathcal{P}$ qui satisfait $\mathcal{B}\backslash\{A \longmapsto B\}$ mais qui ne satisfait pas $A \longmapsto B$.*

Ensuite, nous pouvons définir une base pour un contexte formel donné :

Définition 14. *Soit $\mathcal{K} = (\mathcal{O}, \mathcal{P}, \mathcal{R})$ un contexte formel et $\mathcal{B} = \{A \longmapsto B \mid A, B \subseteq \mathcal{P}\}$ un ensemble d'implications d'attributs. l'ensemble $\mathcal{B}$ est appelé base pour $\mathcal{K}$ ssi il est valide, complet et non redondant pour ce contexte formel. il est notée par $\mathcal{B}^{\mathcal{K}}$.*

Selon cette définition, une base pour un contexte $\mathcal{K}$ est minimale pour l'inclusion, mais nous pouvons avoir plusieurs bases d'implications d'attributs.

2.6.2 Base de Duquenne-Guigues

Duquenne et Guigues [73] ont proposé une base d'implications d'attributs en utilisant les ensembles fermés d'attributs du contexte formel et les ensembles pseudo-fermés d'attributs du contexte formel selon la fermeture de la connexion de Galois. Il a été démontré dans [66, 73] que la base de Duquenne-Guigues est minimale relativement au nombre d'implications d'attributs qu'elle contient. Nous donnons ci-après sa définition. À cette fin, nous avons besoin de définir au préalable la notion de pseudo-fermé comme suit :

Définition 15. *Étant donné un contexte formel $\mathcal{K} = (\mathcal{O}, \mathcal{P}, \mathcal{R})$, un ensemble $A \subseteq \mathcal{P}$ est appelé un pseudo-fermé ssi Il ne s'agit pas d'une intension, i.e. $A^{\Delta\Delta} \neq A$, et il contient la fermeture ($\Delta\Delta$-closures) de tous ses sous-ensembles pseudo-fermé (Formellement, $Q^{\Delta\Delta} \subseteq A$ est valide pour chaque pseudo-fermé $Q \subset A$.).*

La définition d'un pseudo-fermé n'est pas intuitive. Un ensemble $A \subseteq \mathcal{P}$ est un ensemble pseudo-fermé s'il n'est pas fermé, c'est-à-dire si $A^{\Delta\Delta} \neq A$, et s'il contient les fermeture de tous ses sous-ensembles qui sont des ensembles pseudo-fermé. Il est nécessaire de considérer l'ensemble d'attributs vide $\emptyset$, qui est inclus dans tous les ensembles d'attributs, afin de déterminer les ensembles pseudo-fermé. Si l'ensemble vide $\emptyset$ n'est pas fermé, c'est-à-dire si un ensemble ($A \neq \emptyset$) est satisfait par tous les objets du contexte, alors ($\emptyset^{\Delta} = A$) et $\emptyset$ est un ensemble psudo-fermé. En conséquence tous les ensembles pseudo-fermés doivent contenir l'ensemble (A).

En particulier, nous devons d'abord rechercher des attributs singleton qui ne sont pas fermés ($(\{a\}^{\Delta})^{\Delta} \neq \{a\}$) et qui doivent être conformes à la définition de pseudo-fermé. À titre d'exemple, dans la table 2.2, il est clair que $\{a_2\}$ et $\{a_4\}$ ne sont pas des fermés et ils sont des pseudo-fermés. Les autres singletons sont fermés. Par conséquent, l'ensemble $\{a_1, a_3\}$ qui n'est pas une intension et ne contient aucun pseudo-fermé, est lui-même un pseudo-fermé. Les pseudo-fermés supplémentaires dans cet exemple sont$\{a_3, a_5\}$ et $\{a_2, a_3, a_4\}$. Il existe une définition non récursive équivalente pour les pseudo-fermés qui peut être trouvé dans [79].

Duquenne et Guigues [73] ont établi une caractérisation de leur base en utilisant la proposition suivante.

Proposition 1. [73] *L'ensemble des implications d'attributs $\{A \longmapsto A^{\Delta\Delta} \mid A$ est un pseudo-fermé de $\mathcal{K}\}$ est une base minimale. Elle est appelée base de Duquenne-Guigues.*

La base de Duquenne-Guiques est constituée de toutes les implications d'attributs dont la prémisse est un ensemble pseudo-fermé et la conclusion est la fermeture de la prémisse. Il a été démontré dans [66, 73] que la base de Duquenne-Guigues est minimale relativement au nombre d'implications d'attributs qu'elle contient car il ne peut exister de base contenant moins d'implications d'attributs qu'il existe d'ensembles pseudo-fermé dans un contexte formel.

Nous pouvons remplacer $A \longmapsto A^{\Delta\Delta}$ par $A \longmapsto A^{\Delta\Delta} \setminus A$. Dans la table 2.2, la base minimale est

$$\left\{ \begin{array}{ccc} \{a_2\} & \longmapsto & \{a_3\}, \\ \{a_4\} & \longmapsto & \{a_3\}, \\ \{a_1, a_3\} & \longmapsto & \{a_2, a_4\}, \\ \{a_3, a_5\} & \longmapsto & \{a_4\}, \\ \{a_2, a_3, a_4\} & \longmapsto & \{a_1\} \end{array} \right\}.$$

Il est important de rappeler que la sémantique sous-jacente des implications d'attributs est conjonctive. Pour $\{a_1, a_3\} \longmapsto \{a_2, a_4\}$, la lecture est a_1 *et* $a_3 \longmapsto a_2$ *et* a_4. Ce genre d'implications résulte de l'utilisation exclusive de la paire adjointe classique $((.)^{\Delta}, (.)^{\Delta})$.

Un algorithme pour obtenir la base de Duquenne-Guigues (ou l'ensemble de tous les pseudo-fermés) d'un contexte est basé sur l'observation suivante.

Lemme 3. [66] *(Système de fermeture des intensions et des pseudo-fermés). Soit $\mathcal{K} = (\mathcal{O}, \mathcal{P}, \mathcal{R})$ un contexte formel. L'ensemble de toutes les intensions et pseudo-fermés de $\mathcal{K}$ est un système de fermeture sur $\mathcal{P}$.*

Parce que l'ensemble de toutes les intensions et pseudo-fermés forme un système de fermeture, il peut être calculé en utilisant l'algorithme Next-Closure. À cette fin, nous devons identifier l'opérateur de fermeture correspondant et un moyen efficace de le calculer. Afin d'identifier l'opérateur de fermeture, nous définissons la fermeture implicationnelle d'un ensemble d'attributs.

Définition 16. *(Fermeture implicationnelle). Soit $A \subseteq \mathcal{P}$ un ensemble d'attributs et $\mathcal{L}$ un ensemble d'implications. Alors la fermeture implicationnelle de A par rapport à $\mathcal{L}$ est le plus petit ensemble $B, A \subseteq B \subseteq \mathcal{P}$, qui satisfait toutes les implications de $\mathcal{L}$. Nous indiquons la fermeture implicationnelle de A par rapport à $\mathcal{L}$ par $\mathcal{L}(A)$.*

On peut montrer que la fermeture implicationnelle $\mathcal{L}(.)$ est un opérateur de fermeture. Les ensembles fermés par rapport à $\mathcal{L}(.)$ sont exactement ceux qui satisfont toutes les implications de $\mathcal{L}$. $\mathcal{L}(A)$ peut être calculé en temps linéaire dans la taille de A et $\mathcal{L}$ [46].

En utilisant l'algorithme Next-Closure, on peut calculer les ensembles fermés de ce système de fermeture, c'est-à-dire les intensions et les pseudo-fermés du contexte formel sous-jacent. Nous commençons par la plus petite intension ou pseudo-fermé par rapport à l'ordre lectique (voir la définition 10), c'est-à-dire la fermeture de $\emptyset$, et appliquons successivement la lemme 2 jusqu'à ce que l'on atteigne la plus grande intension $\mathcal{P}$. L'opérateur de fermeture induit le système de fermeture des intensions et pseudo-fermés. Il est obtenu par une modification de l'opérateur de fermeture implicationnelle $\mathcal{L}(.)$ donnée dans la définition 16.

Lorsque nous voulons calculer l'intension ou le pseudo-fermé pour un ensemble donné A, il suffit de connaître les implications de la base de Duquenne-Guigues dont le côté gauche est strictement contenu dans A. Rappelons que l'ordre lectrique prolonge l'ordre du sous-ensemble. Puisque Next-Closure calcule les intensions et les pseudo-intensions dans l'ordre lectique, ces implications se situent dans la partie déjà calculée de la base de Duquenne-Guigues, ce qui nous donne un moyen efficace de calculer le pseudo-fermé d'un ensemble d'attributs. L'algorithme résultant est présenté dans l'algorithme 2. Il calcule non seulement les intensions et les pseudo-fermés mais aussi la base de Duquenne-Guigues.

Algorithme 2. [73] Calcul de la base de Duquenne-Guigues

$\mathcal{K} := (\mathcal{O}, \mathcal{P}, \mathcal{R})$ {Entrée : contexte formel $\mathcal{K}$}
$P_0 := \emptyset$ {Est toujours la plus petite intension ou pseudo-fermé}
$\mathcal{L}_0 := \emptyset$ {Initialisation de l'ensemble des implications}
$i := 0$
tant que $P_i \neq \mathcal{P}$ **faire**
  si $P_i \neq P_i^{\Delta\Delta}$ **alors**
    $\mathcal{L}_{i+1} := \mathcal{L}_i \cup \{P_i \longmapsto P_i^{\Delta\Delta}\}$ {$P_i$est un pseudo-fermé}
  fin si
  P_{i+1} := Sous-ensemble de $\mathcal{P}$ lectiquement plus petit qui est
  - lectiquement plus grande que P_i, et
  - satisfait toutes les implications de $\mathcal{L}_{i+1}$
  $i := i + 1$
fin tant que ;
retourne $\mathcal{L}_i$

2.6.3 Exploration d'attributs

Dans la section précédente, nous avons vu comment calculer la base de Duquenne-Guigues d'un contexte formel donné. Dans le cas où le contexte formel ne contient pas tous les exemples de la réalité, nous obtiendrons des implications qui sont valides dans le contexte formel mais qui ne sont pas vraies dans la réalité, car dans la réalité on peut avoir un contre-exemple qui n'est pas présent dans le contexte formel.

A titre d'exemple, considérons le contexte formel $\mathcal{K}_0$ de la table 2.3 . L'implication *Femme* $\longmapsto$ *Mère* est valide dans le contexte formel $\mathcal{K}_0$ mais ne l'est pas dans la réalité car il est possible de trouver un contre-exemple qui n'existe pas dans le contexte formel (i.e. le contexte formel n'est pas complet).

L'exploration d'attributs [64] est une méthode issue de l'ACF, qui permet de compléter les contextes formels de manière interactive, tout en calculant la base de Duquenne-Guigues pour le contexte complet [63]. La procédure d'exploration d'attributs prend comme entrée un contexte formel et obtient plus d'informations en interrogeant un expert humain. On suppose que l'expert a une connaissance du contexte complet (la réalité). L'algorithme d'exploration d'attributs calcule les implications et les présente à l'expert. Pour chaque implication, l'expert a le choix de l'accepter ou de la rejeter. S'il l'accepte, elle sera ajoutée à la base d'implication qui est calculée. Si il la rejette, on lui demande de fournir un contre-exemple sous la forme d'un objet et de son intension. L'algorithme complet est décrit dans l'algorithme 3. Il s'agit d'une modification de Next-Closure pour le calcul de la base de Duquenne-Guigues. On donne la figure 2.4 pour une meilleure compréhension de l'algorithme (Algorithme 3) d'exploration d'attributs.

Algorithme 3. [64] Exploration d'attributs

$\mathcal{K}_0 := (\mathcal{O}_0, \mathcal{P}, \mathcal{R}_0)$ {Entrée : contexte formel $\mathcal{K}$}
$P_0 := \emptyset$ {Est toujours la plus petite intension ou pseudo-fermé}
$\mathcal{L}_0 := \emptyset$ {Initialisation de l'ensemble des implications}
$i := 0, j := 0$
tant que $P_i \neq \mathcal{P}$ **faire**
  tant que $P_i \neq P_i^{\Delta\Delta}$ et l'expert rejette $P_i \longmapsto P_i^{\Delta\Delta}$ **faire**
    Demandez à un expert un nouvel objet x et son intension x^{Δ}
    $\mathcal{O}_{j+1} := \mathcal{O}_j \cup \{x\}$
    $\mathcal{R}_{j+1} := \mathcal{R}_j \cup \{(x, a) \mid a \in x^{\Delta}\}$
    $\mathcal{K}_{j+1} := (\mathcal{O}_{i+1}, \mathcal{P}, \mathcal{R}_{i+1})$
    $j := j + 1$
  fin tant que
  si $P_i \neq P_i^{\Delta\Delta}$ **alors**
    $\mathcal{L}_{i+1} := \mathcal{L}_i \cup \{P_i \longmapsto P_i^{\Delta\Delta}\}$ {$P_i$est un pseudo-fermé}
  fin si
  P_{i+1} := Sous-ensemble de $\mathcal{P}$ lectiquement plus petit qui est
  - lectiquement plus grande que P_i, et
  - satisfait toutes les implications de $\mathcal{L}_{i+1}$
  $i := i + 1$
fin tant que ;
retourne $\mathcal{L}_i$

Exemple 2. *Considérant le contexte formel $\mathcal{K}_0 = (\mathcal{O}_0, \mathcal{P}, \mathcal{R}_0)$ de la table 2.3. L'objectif est de trouver toutes les implications pertinentes, c'est-à-dire les implications de la base*

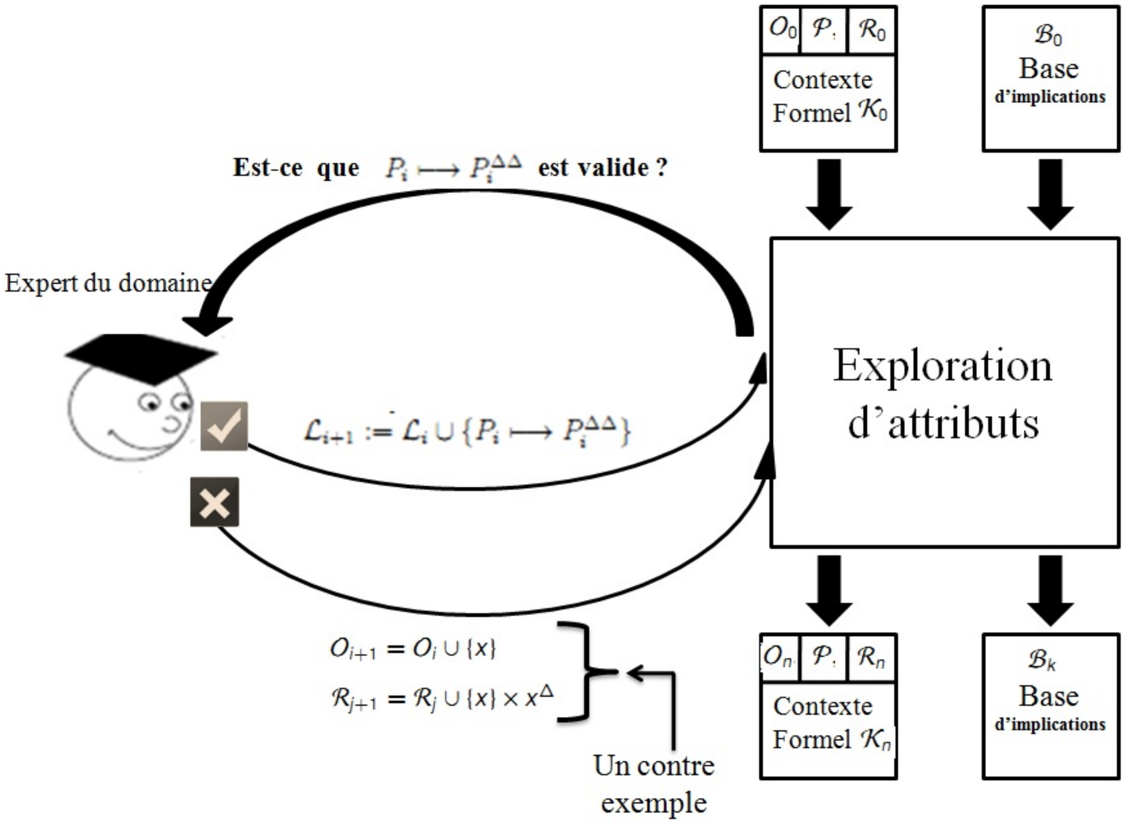

FIGURE 2.4 – Schéma de l'algorithme d'exploration d'attributs

de Duquenne-Guigues en appliquant l'algorithme 3 et en interagissant avec un expert humain.

TABLE 2.3 – Contexte formel $\mathcal{K}_0$.

$\mathcal{R}_0$	Femme	Homme	Mère	Père	Parent
Ali		×		×	×
Kahina	×		×		×
Salim		×			

La première implication trouvée par l'algorithme 3 est

$$\{\text{Père }\} \longmapsto \{Homme, \text{Père}, Parent\}$$

Comme cette implication est vraie, l'expert l'accepte. De même pour les implications

$$\{\text{Mère}\} \longmapsto \{Femme, \text{Mère}, Parent\} \text{ } et$$
$$\{Homme, Parent\} \longmapsto \{Homme, \text{Père}, Parent\}$$

Elles sont acceptées et ajoutées à la base d'implication. La quatrième implication est

$$\{Femme\} \longmapsto \{Femme, \text{Mère}, Parent\}$$

Cette implication ne tient pas dans la réalité. Par conséquent, l'expert rejette l'implication et fournit le contre-exemple de Lisa qui sera ajouté dans le nouveau contexte formel $\mathcal{K}_1 = (\mathcal{O}_1, \mathcal{P}, \mathcal{R}_1)$ (table 2.4). Les deux dernières implications sont acceptées et aucun contre-exemple n'est fourni. Après ces dernières questions, l'exploration d'attributs se termine :

$$\{Femme, Parent\} \longmapsto \{Femme, \text{Mère}, Parent\}$$
$$\{Femme, Homme\} \longmapsto \{Femme, Homme, \text{Mère}, \text{Père}, Parent\}$$

TABLE 2.4 – Contexte formel $\mathcal{K}_1$.

$\mathcal{R}_1$	Femme	Homme	Mère	Père	Parent
Ali		×		×	×
Kahina	×		×		×
Salim		×			
Lisa	×				

2.7 Généralisation de la représentation d'un contexte formel

Classiquement, l'ACF s'applique à un contexte formel (c.à.d. une relation binaire) où un objet vérifie pleinement un attribut ou ne le vérifie pas du tout. Une telle relation est Booléenne. Plusieurs approches ont été proposées afin de prendre en considération la diversité de la nature des informations manipulées dans un contexte formel. Ces informations peuvent exprimer une certaine gradualité auquel cas, la représentation du contexte formel est multi-valué. Ces informations peuvent être aussi incomplètes Nous présentons dans cette section ces différents types de représentations et les approches y afférentes.

2.7.1 Représentation graduelle : l'Analyse de Concepts Formels Floue

Un contexte formel flou est un contexte multi-valué où les valeurs de la relation $objet \times attribut$ représentent un degré de vérité qui exprime, de manière graduelle, à quel niveau l'objet est en relation avec l'attribut. L'Analyse de Concepts Formels Floue (ACFF) est l'approche qui permet de prendre en considération de pareilles relations (contextes). De nombreux travaux ont été effectués, souvent indépendamment les uns des autres, dans le cadre de l'ACFF. Brusco et Fuentes Gonzales [1] ont été les premiers à considérer des contextes formels flous (ils ont utilisé une S-implication) ensuite plusieurs approches sont venues entres autres l'approche de Pollandt (1997) [96], de Ben Yahia et al. (2001) [116], de Belohlavek (2002) [19] , de Latiri et al. (2003) [80], de Georgescu (2004) [70], de Belohlavek et al. (2005) [100], et de Medina et al. (2009) [84]. Toutes ces approches utilisent une algèbre résiduée. Un exemple de contexte formel flou est illustré dans la table 2.5

TABLE 2.5 – Exemple d'un contexte formel flou

Région \ Climat	chaud	froid	pluvieux	humide
région$_1$	0.5	0.5	1.0	1.0
région$_2$	1.0	0.0	1.0	1.0
région$_3$	0.5	0.5	0.0	0.0

2.7.1.1 Rappels sur la théorie des ensembles flous

La théorie des ensembles flous a été introduite par Lotfi Zadeh en 1965, dans son article "fuzzy sets" [119], Zadeh définit un ensemble flou comme "une collection telle que l'appartenance à cette collection peut prendre toute valeur entre 0 et 1". C'est une extension de la théorie des ensembles classiques où l'appartenance ne peut avoir que la valeur 0 (n'appartient pas) ou la valeur 1 (appartient).

Définition 17 (sous-ensemble flou)**.** *Soit X un univers de discours et soit x un élément quelconque de X. Un sous-ensemble flou A de X est défini comme l'ensemble des couples :*
$A = \{(x, \mu_A(x)) \mid x \in X\}$
avec
$\mu_A : X \to [0, 1]$

Ainsi, un sous-ensemble flou A de X est caractérisé par une fonction d'appartenance $\mu_A(x)$ qui associe à chaque élément de x de X un degré dans l'intervalle [0,1]. Ce degré noté $\mu_A(x)$ exprime la caractéristique de transition graduelle et non brutale entre l'appartenance complète et la non appartenance totale de l'élément x à l'ensemble A. Trois cas sont possibles :
1- $\mu_A(x) = 0$ si x n'appartient pas à A (non appartenance totale),
2- $0 < \mu_A(x) < 1$ si x appartient partiellement à A (plus $\mu_A(x)$ se rapproche de 1, plus x appartient à A,
3- $\mu_A(x) = 1$ si x appartient entièrement à A (appartenance complète).

Exemple 3. *Soit A l'ensemble de personnes de taille moyenne, c'est-à-dire l'ensemble des personnes ayant une taille comprise entre 1m 60 et 1 m 80. En théorie des ensembles dits classiques, la fonction caractéristique de l'ensemble A renvoie :*
- 0 pour les tailles hors de l'intervalle [1m 60, 1m 80]

- 1 pour les tailles appartenant à cet intervalle.

Ce qui fait qu'une personne mesurant 1m 59 ne sera pas considérée de taille moyenne, alors qu'une personne plus grande d'un centimètre 1m 60 le sera.

En théorie des ensembles flous, on peut définir un ensemble flou des personnes de taille moyenne. Cet ensemble sera caractérisé par une fonction d'appartenance qui, contrairement à une fonction caractéristique classique, peut renvoyer des valeurs dans l'intervalle [0, 1]. Ces valeurs représentent le degré d'appartenance à l'ensemble des personnes de taille moyenne. Par exemple une personne de taille de 1m 60 aura un degré d'appartenance égale à 1 et une personne mesurant 1m 59 aura un degré d'appartenance de 0,9.

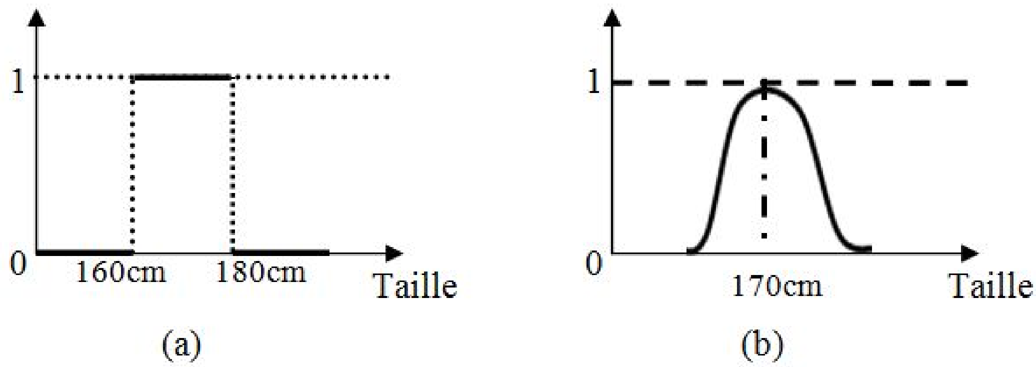

Figure 2.5 – Représentation graphique d'une fonction d'appartenance caractérisant (a) un ensemble classique et (b) un ensemble flou.

Dans ce qui suit on va définir les opérations appliquées sur les sous-ensembles flous.

Egalité

Deux sous-ensembles flous A et B de X sont égaux, si leurs fonctions d'appartenance prennent la même valeur pour tous les éléments x de X. Formellement $A = B$ si et seulement si :

$$\forall x \in X, \mu_A(x) = \mu_B(x)$$

Complément

Le complémentaire d'un sous-ensemble flou A de X noté $\overline{A}$ est défini par :

$$\forall x \in X, \mu_{\overline{A}}(x) = 1 - \mu_A(x)$$

Inclusion

Soit A et B deux sous-ensembles flous de X. Si pour n'importe quel élément x de X, x appartient toujours moins à A qu'à B, alors on dit que A est inclus dans B ($A \subseteq B$). Formellement, $A \subseteq B$ si et seulement si :

$$\forall X, \mu_A(x) \leq \mu_B(x)$$

Intersection

Afin de généraliser l'intersection (et par extension la conjonction de propositions) sur les sous-ensembles flous, les normes triangulaires (t-normes) ont été définies comme suit :

Définition 18 (t-norme)**.**

Une t-norme (appelée aussi norme triangulaire) est une fonction $\otimes : [0,1] \times [0,1] \to [0,1]$ *qui possède les propriétés suivantes* $\forall x, y, z, v \in [0,1]$ *:*

- *Commutativité :* $(x \otimes y) = (y \otimes x)$
- *Associativité :* $x \otimes (y \otimes z) = (x \otimes y) \otimes z$
- *Monotonie :* $(x \otimes y) \leq (z \otimes v)$ *si* $x \leq z$ *ou* $y \leq v$
- *1 est l'élément neutre :* $(x \otimes 1) = x$
- $\forall x, y \in [0,1], (x \otimes y) \leq x$ *et* $(x \otimes y) \leq y$

Il existe une infinité de t-normes, les plus courantes sont données dans la table 2.6.

TABLE 2.6 – Les t-normes les plus courantes.

Auteur	**t-norme** (x ⊗ y)
Zadeh	Min (x, y)
Lukasiewicz	Max (x + y -1, 0)
Weber	x si y=1 y si x= 1 0 sinon

L'intersection des deux ensembles flous A et B, (AB) est donnée par :

$$\mu_{A\cap B}(x) = (\mu_A(x) \otimes \mu_B(x))$$

Union

L'union (et par extension la disjonction de proposition) est généralisée sur les sous-ensembles flous, par les conormes triangulaires (t-conormes) définies comme suit :

Définition 19 (t-conorme)**.**

Une t-conorme (appelée aussi conorme triangulaire) est une fonction $\oplus : [0,1] \times [0,1] \to [0,1]$ *qui possède les propriétés suivantes* $\forall x, y, z, v \in [0,1]$ *:*

- *Commutativité :* $(x \oplus y) = (y \oplus x)$

- *Associativité :* $x \oplus (y \oplus z) = (x \oplus y) \oplus z$
- *Monotonie :* $(x \oplus y) \leq (z \oplus v)$ *si* $x \leq z$ *ou* $y \leq v$
- *0 est l'élément neutre :* $(x \oplus 0) = x$
- $\forall x, y \in [0, 1], (x \oplus y) \leq x$ *et* $(x \oplus y) \leq y$

Il existe une infinité de t-conormes, les plus courantes sont données dans la table 2.7.

TABLE 2.7 – Les t-conormes les plus courantes.

Auteur	**t-conorme** (x ⊕ y)
Zadeh	Max (x, y)
Lukasiewicz	Min (x + y, 1)
Weber	x si y=0 y si x= 0 1 sinon

L'intersection des deux ensembles flous A et B, (AB) est donnée par :

$$\mu_{A\cap B}(x) = (\mu_A(x) \otimes \mu_B(x))$$

L'union des deux ensembles flous A et B, $(A \cup B)$ est donnée par :

$$\mu_{A\cup B}(x) = (\mu_A(x) \oplus \mu_B(x))$$

Les implications floues

Dans le cas classique (non flou), une implication $p \rightarrow q$ est interprétée à vrai ou à faux (p et q prennent aussi des valeurs booléennes). Une extension floue de cette implication a été proposée lorsque les valeurs des propositions p et q varient dans l'intervalle [0, 1]. Toute implication floue est une fonction (notée $\tilde{\rightarrow}$) définie par :

$$\begin{aligned} \tilde{\rightarrow} : [0,1] \times [0,1] &\rightarrow [0,1] \\ (p, q) &\mapsto (p \tilde{\rightarrow} q) \end{aligned}$$

et elle est d'autant plus vraie (respectivement fausse) que son résultat est proche de 1 (respectivement 0). Les implications floues se divisent principalement en deux familles : les S-implications et les R-implications que nous présentons ci-après.

S-implication

L'appellation S-implications vient de l'expression anglaise *Strong implication*. On définit la classe des S-implications à partir de l'expression : ((non p) ou q) de la manière suivante :

$$p \to q = (1 - p \oplus q)$$

R-implication
La seconde classe d'implications floues est la R-implication, ainsi dénommées parce qu'elles utilisent le principe de *résiduation*. Une R-implication est définie comme suit :

$$p \to q = sup\{u \in [0,1] \mid (p \otimes u) \leq q\}$$

2.7.1.2 Contexte formel flou et concepts formels flous

Formellement, un contexte formel flou est un tuple $\mathcal{K} = (L, \mathcal{O}, \mathcal{P}, \mathcal{R})$ avec $L = [0,1]$. la relation floue $\mathcal{R} \in L^{(\mathcal{O}\times\mathcal{P})}$ est une fonction définie $\mathcal{O} \times \mathcal{P} \to L$ (i.e $\mathcal{R} : \mathcal{O} \times \mathcal{P} \to [0,1]$) qui assigne à chaque objet $x \in O$ et à chaque attribut $a \in P$ un degré $\mathcal{R}(x,a)$ qui exprime à quelle niveau l'objet x vérifie (possède) l'attribut a.

Cette relation est représentée sous forme d'une table. Généralement les lignes représentent les objets et les colonnes les propriétés, chaque cellule exprime une valeur appartenant à $L \in [0,1]$, comme le montre la table 2.5. Dans la théorie des ensembles flous, l'ensemble des parties floues d'un ensemble d'objets $\mathcal{O}$ est représenté par $L^{\mathcal{O}}$ et l'ensemble des parties floues de l'ensemble des attributs $\mathcal{P}$ est représenté par $L^{\mathcal{P}}$.

L'ACFF consiste à induire tous les concepts formels flous $(\widetilde{X}, \widetilde{A})$ où $(\widetilde{X} \in L^{\mathcal{O}}, \widetilde{A} \in L^{\mathcal{P}}$ et $\widetilde{X}^{\Delta} = \widetilde{A}$ et $\widetilde{A}^{\Delta} = \widetilde{X}$.

Pollandt [96] et Belohlavek [18] définissent l'opérateur de dérivation $(.)^{\Delta} : L^{\mathcal{O}} \to L^{\mathcal{P}}$ comme suit :

$$(\widetilde{X})^{\Delta}(a) = \bigwedge\nolimits_{x\in\mathcal{O}}(\widetilde{X}(x)) \to_R \mathcal{R}(x,a)$$
$$(\widetilde{A})^{\Delta}(x) = \bigwedge\nolimits_{a\in\mathcal{P}}(\widetilde{A}(a)) \to_R \mathcal{R}(x,a)$$

- $\to_R$ est une R-implication.
- $(\widetilde{X})^{\Delta}(a)$ représente un degré de vérité de l'attribut a satisfait par tous les objets de X
- $(\widetilde{A})^{\Delta}(x)$ représente un degré de vérité de l'objet x satisfaisant tous les attributs de A
- L'ensemble de tous les concepts formels flous est aussi équipé d'une relation d'ordre définie comme suit : $(\widetilde{X}_1, \widetilde{A}_1) \preceq (\widetilde{X}_2, \widetilde{A}_2)$ ssi $\widetilde{X}_1 \subseteq \widetilde{X}_2$ ou $\widetilde{A}_2 \subseteq \widetilde{A}_1$
- L'ensemble de tous ces concepts formels flous muni d'une relation d'ordre (notée $\preceq$) forme un treillis complet.

2.7.2 Représentations incomplètes : aspects sémantiques

Des recherches récentes [43] [42] [37] [47] [51] ont soulevé différentes questions sémantiques liées à la généralisation d'un contexte formel. Dans [43] [42], les auteurs ont traité la notion d'incomplétude dans le cadre de la théorie des possibilités. Ils ont également introduit la notion de typicalité pour les objets et la notion d'importance pour les attributs.

Une première interprétation correspond au cas où les degrés représentent à quel niveau un objet satisfait un attribut. Cette interprétation est appelé interprétation à échelle unipolaire positive et $\mathcal{R}(a)$, qui désigne l'ensemble des objets possédant l'attribut a, représente le support du sous-ensemble flou des objets satisfaisant l'attribut a. Considérer la convention opposée où $\mathcal{R}(a)$ serait le noyau du sous-ensemble flou des objets satisfaisant la propriété a revient à introduire des degrés dans les cases vides.

Une seconde interprétation consiste à remplacer les cases ayant une croix et les cases vides par des degrés d'appartenance de $L = [0, 1]$. l'idée de base consiste à considérer une valeur pivot (par exemple a = 0.5) qui permet de séparer le cas où l'objet satisfait plus la propriété qu'il ne la satisfait pas. Ce type d'interprétation est appelé interprétation à échelle bipolaire. Les interprétations positives permettent d'avoir un contexte flou à partir d'un contexte à attributs quantitatifs [112].

D'autres aspects de degrés sont traités comme l'incertitude (incomplétude). Dans ce cas, les attributs peuvent rester non flous mais la certitude de la satisfaction de l'attribut par un objet n'est pas complète. Elle peut être représentée par une distribution de probabilités ou de possibilités au sens de [43] [42]. Plus précisément, dans une table décrivant un contexte incertain, les cases sont remplies par des paires (α,β) où α est le degré de nécessité que l'objet ait l'attribut et β est le degré de nécessité que l'objet n'aît pas l'attribut avec $min(\alpha,\beta) = 0$. Les situations (1, 0) et (0, 1) correspondent aux situations complètement informées où on sait avec certitude que l'objet a la propriété ou qu'il ne l'a pas. La situation (0, 0) reflète l'ignorance totale et celle où $0 < max(\alpha,\beta) < 1$ reflète l'ignorance partielle.

Des approches introduisant la notion d'incomplétude sont proposées dans [37] [47] [51]. Dans ces travaux, les extensions sont fondées sur la définition des opérateurs de dérivation et des connexions de Galois en utilisant les mesures définies dans la théorie des possibilités : la possibilité (Π) et son dual la nécessité (N), la suffisance (Δ) et son dual la certitude potentielle (∇). La suffisance est la mesure utilisée dans le cadre de l'ACF classique. Les opérateurs de nécessité et de possibilité sont appliqués à des contextes flous. L'intérêt de ces travaux est d'introduire une nouvelle connexion de Galois, inspirée par une lecture possibiliste de l'ACF concernant les données binaires [37] [51] [47] et pour les contextes hétérogènes [7]. Une des connexions de Galois met en évidence la décomposition du contexte en plusieurs sous-contextes indépendants. L'idée de ces travaux est de représenter la valeur d'une case dans un contexte (elle peut être binaire, numérique, ou symbolique) par une distribution de possibilités et ensuite d'appliquer les opérateurs de dérivation pour construire le treillis de concepts dans le cas où les opérateurs forment une connexion de Galois.

Comme cet aspect constitue l'essentiel de notre travail, il sera décrit de manière plus détaillée dans la suite de ce document.

2.8 Nouvelles tendances en analyse de concepts formels

Plusieurs travaux de recherche ont été menés dans le but d'étendre le paradigme de l'ACF. Ces différentes extensions sont présentées ci-après.

2.8.1 L'Analyse de concepts logiques

L'Analyse de Concepts Logiques (ACL) [61] consiste à étendre les résultats de l'ACF aux contextes logiques. Un contexte logique est un contexte multi-valué dans lequel les attributs sont des descriptions qui prennent comme valeurs des formules logiques décrivant les objets du contexte. Formellement, une logique en ACL est définie comme suit.

Définition 20. *Une logique $\mathcal{L}$ est définie par le 6-uplet ($L, \sqsubseteq, \sqcap, \sqcup, \top, \bot$), où*

- *L est un langage de formules logiques*
- *$\sqsubseteq$ est la relation de subsomption sur L*
- *$\sqcap$ est l'opérateur de conjonction (correspond à $\cap$ en ACF)*
- *$\sqcup$ est l'opérateur de disjonction (correspond à $\cup$ en ACF)*
- *$\top$ est la tautologie dans L*
- *$\bot$ est la contradiction dans L*

De cette manière, la logique devient un paramètre formel dans la théorie de l'ACF.

2.8.1.1 Contexte logique

Un contexte logique est un triplet $(\mathcal{O}, \mathcal{L}, \delta)$ où $(\mathcal{O}$ est un ensemble fini d'objets, $\mathcal{L}$ est un langage, et δ est une fonction de $(\mathcal{O}$ dans δ qui associe à chaque objet une formule décrivant les attributs de l'objet (ou une description logique).

Exemple 4. *Un exemple de contexte logique est donné dans la table 2.8.*

TABLE 2.8 – Descriptions des objets du contexte logique $\mathcal{K}_{CL}$.

objet	description
x	h
y	f
z	$c \wedge (a \vee b)$

Les lettre h, f, et c sont les abréviations respectives de homme, femme et chauve. La logique employée est la logique propositionnelle $\mathcal{P}$ pour un ensemble de propositions atomiques $A = \{h, f, c\}$. Le contexte logique $\mathcal{K}_{CL}$ est défini par $(\mathcal{O}_{CL}, \mathcal{P}, \delta)$, où $\mathcal{O}_{CL} = \{x, y, z\}$ et où δ la fonction qui décrit les objets est définie dans la table 2.8.

2.8.1.2 Concept logique et treillis de concepts logiques

Les opérateurs de dérivation entre $2^{\mathcal{O}}$ et $\mathcal{L}$ sont définis comme suit.

$$\sigma : 2^{\mathcal{O}} \rightarrow \mathcal{L}, \sigma(X) = \bigsqcup_{x \in X} \sigma(x)$$
$$\tau : \mathcal{L} \rightarrow 2^{\mathcal{O}}, \tau(A) = \{x \in \mathcal{O} \mid \sigma(x) \sqsubseteq A\}$$

$\sigma(X)$ est l'expression par une formule logique des attributs communs à tous les objets dans X et $\tau(A)$ est l'ensemble de tous les objets de $\mathcal{O}$ dont la description est subsumée par la formule A. Les deux opérateurs forment une connexion de Galois entre $2^{\mathcal{O}}$ et $\mathcal{L}$. Ils sont à l'origine de la définition de concept logique et du treillis de concepts logiques correspondant à un contexte logique donné. Un concept logique est une paire (X, A) tel que $\sigma(X) = A$ et $\tau(A) = X$. X et A sont respectivement l'extension et l'intension du concept. Les concepts logiques peuvent être ordonnés selon l'inclusion entre leurs extensions ou de manière équivalente selon la subsomption entre leurs intensions. L'ensemble des concepts logiques d'un contexte logique ordonnés de cette façon forme un treillis de concepts logiques.

Exemple 4 (suite) Le treillis de concepts logiques correspondant au contexte logique donné dans la table 2.8 est donné dans la figure 2.6. Les nœuds du diagramme de Hasse de ce treillis sont étiquetés selon l'étiquetage réduit des concepts logiques [61].

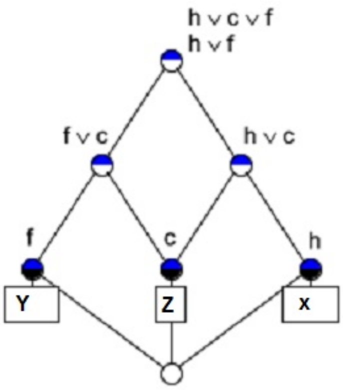

FIGURE 2.6 – Treillis de concepts formel du contexte $\mathcal{K}_{CL}$.

Le passage de l'analyse de concepts à l'ACL garde ses propriétés fondamentales, à savoir les concepts, les intensions et les extensions. L'ajout d'une logique dans l'analyse de concepts permet aussi d'obtenir de nouvelles informations grâce à la relation de subsomption, établissant de nouvelles relations entre formules logiques. Un point fort à relever dans cette technique est le principe de généricité. En effet, n'importe quelle logique peut être définie grâce aux définitions précédentes. Ainsi, l'apport d'un contexte (l'étude d'un domaine particulier) permet de définir des connaissances contextuelles logiques. Ensuite, à l'instar de l'analyse de concepts, l'ACL permettra d'obtenir un treillis de concepts logique.

2.8.2 Les structures de patrons

Les structures de patrons ont été introduites en 2001 par Ganter et Kuznetsov [67] pour construire un treillis de concepts à partir de données complexes sans les avoir transformées sous forme binaire et sans perte d'information. La connexion de Galois classique met en évidence une relation entre les éléments du treillis de $(2^{\mathcal{O}}, \subseteq)$ et les éléments du treillis d'attributs $(2^{\mathcal{P}}, \subseteq)$ et vice versa. Ces treillis sont partiellement ordonnés. Pour construire un treillis à partir d'un contexte hétérogène, il faut donc trouver un ordre partiel sur les descriptions des objets dans le contexte multivalué. Pour cela, Ganter et Kuznetsov ont introduit les structures de patrons en formalisant l'ensemble des objets de $\mathcal{O}$ et les descriptions (appelées patrons) d de l'inf-demi-treillis des descriptions ordonnées. Soient A et B deux sous-ensembles d'attributs, on a $A \subseteq B \Leftrightarrow A \cap B = A$. L'intersection ensembliste possède les propriétés d'un infimum dans un demi-treillis, d'où l'idée d'ordonner les patrons suivant la relation :

$$c \sqsubseteq d \Leftrightarrow c \sqcap d = c$$

Formellement, on considère le triplet $(\mathcal{O}, (D, \sqcap), \delta)$ où $\mathcal{O}$ est l'ensemble des objets, $(D, \sqcap)$ est l'inf-demi-treillis des descriptions, et δ est la fonction qui associe à chaque objet sa description dans D. Le triplet $(\mathcal{O}, (D, \sqcap), \delta)$ est appelé structure de patrons et la construction du treillis suit l'ACF classique. Pour établire la connexion de Galois entre $(2^{\mathcal{O}}, \subseteq)$ et $(D, \sqsubseteq)$, les opérateurs de dérivation sont les suivantes : soit $X \subseteq \mathcal{O}$

$$X^{\square} = \textstyle\bigsqcap_{x \in X} \delta(x)$$

Soit $d \subseteq (D, \sqcap)$

$$d^{\square} = \{x \in X \mid d \sqsubseteq \delta(x)\}$$

Le premier opérateur retourne pour un sous-ensemble X d'objets de $\mathcal{O}$ l'infimum de leurs descriptions. De manière duale, le second opérateur retourne pour toute description d dans l'inf-demi-treillis le sous-ensemble d'objets ayant une description qui subsume la description d.

Les concepts de $(\mathcal{O}, (D, \sqcap), \delta)$ sont des couples (X, d) tel que $X^{\square} = d$ et $d^{\square} = X$. Ils sont ordonnés par $(X_1, d_1) \preceq (X_2, d_2)) \Leftrightarrow (X_1 \subseteq X_2)$ $(\Leftrightarrow d_2 \sqsubseteq d_1)$ pour former le treillis de concepts de $(\mathcal{O}, (D, \sqcap), \delta)$.

En ce qui concerne les contextes formels, les implications peuvent être définies. Pour $c, d \in D$, l'implication patron $c \rightarrow d$ est valide si $c^{\square} \subseteq d^{\square}$, c'est-à-dire que le parton d se trouve dans une description d'objet si le patron c se trouve aussi dans la même . De même, pour $X, Y \subseteq \mathcal{O}$, l'implication l'objet $X \rightarrow Y$ est valide si $X^{\square} \sqsubseteq Y^{\square}$, ce qui signifie que tous les patrons qui se trouvent dans tous les objets de l'ensemble X se trouvent également dans tous les objets de l'ensemble Y.

Les algorithmes de l'ACF existants peuvent être réutilisés avec de légère modifications pour calculer les structures de patrons.

2.8.3 L'Analyse de concepts triadiques

L'ACF a été étendue depuis 1995 au cas triadique afin de traiter les données tridimensionnelles [81] [66]. Cependant, très peu de travaux se sont intéressés à l'analyse de concepts triadiques. Avec l'arrivée des folksonomies comme étant la structure centrale des réseaux sociaux [60], la focalisation sur l'analyse de concepts triadiques s'est considérablement accrue. Le concept de folksonomie, structuré sous la forme d'un contexte triadique, est considéré comme faisant partie intégrante du Web 2.0 (ou web social) [111].

Wille et Ganter ont introduit dans [66] une définition d'un contexte triadique qui est l'extension d'un contexte classique par ajout d'une nouvelle colonne appelée condition permettant la représentation de données supplémentaires.

Définition 21. *(Contexte triadique)*
Les données sont représentées par un contexte triadique noté $\mathcal{K} = (\mathcal{O}, \mathcal{P}, \mathcal{C}, \mathcal{R})$, où , $\mathcal{O}$, $\mathcal{P}$ et $\mathcal{C}$ sont respectivement appelés ensembles d'objets, attributs et conditions, et $\mathcal{R} \subseteq \mathcal{O} \times \mathcal{P} \times \mathcal{C}$. $(x, a, c) \in \mathcal{R}$ signifie que "L'objet x possède l'attribut a sous la condition c".

Exemple 5. *Un exemple d'un tel contexte triadique est donné dans la table 2.9 dans lequel la première croix (vers la gauche) indique le fait que "l'objet x_2 satisfait l'attribut a_1 sous la condition c_1, c'est-à-dire $(x_2, a_1, c_1) \in \mathcal{R}$. Dans cette représentation tabulaire, chaque table correspond à la projection du contexte triadique pour une condition. Un autre choix aurait pu être fait.*

	c_1			c_2			c_3		
	a_1	a_2	a_3	a_1	a_2	a_3	a_1	a_2	a_3
x_1			×	×	×	×	×		×
x_2	×	×		×	×		×		
x_3		×	×	×	×	×	×	×	×
x_4		×	×		×	×		×	×

TABLE 2.9 – Contexte triadique $K = (\mathcal{O}, \mathcal{P}, \mathcal{C}, \mathcal{R})$.
avec le concept triadique $(\{x_3, x_4\}, \{a_2, a_3\}, \{c_1, c_2, c_3\})$.

Définition 22. *(Concept triadique)*
Un concept triadique de $K = (\mathcal{O}, \mathcal{P}, \mathcal{C}, \mathcal{R})$ est un triple (X_1, A_1, C_1) avec $X_1 \subseteq \mathcal{O}$, $A_1 \subseteq \mathcal{P}$ et $A_1 \subseteq \mathcal{R}$ satisfaisant les deux instructions suivantes : (i) $X_1 \times A_1 \times C_1 \subseteq \mathcal{R}$ et (ii) pour $X_2 \times A_2 \times C_2 \subseteq \mathcal{R}$, nous avons $X_1 \subseteq X_2$, $A_1 \subseteq A_2$, $C_1 \subseteq C_2$ implique $(X_1, A_1, C_1) = (X_2, A_2, C_2)$. $K = (\mathcal{O}, \mathcal{P}, \mathcal{C}, \mathcal{R})$ est représenté par une table à 3 dimensions, et (i) signifie qu'un concept est un cube rempli de croix, tandis que (ii) caractérise la maximalité. Pour un concept triadique (X_1, A_1, C_1), X_1 est l'extension, A_1 l'intension et C_1 le modus.

Exemple 5 (suite) $(\{x_3, x_4\}, \{a_2, a_3\}, \{c_1, c_2, c_3\})$ est un concept triadique dans le contexte triadique représenté par la table 2.9. En représentant le contexte triadique en tant que boîte, où chaque condition est une couche, on peut observer que ce concept triadique désigne un parallélépipédique maximal rectangulaire des croisements (modulo lignes, colonnes et couches permutations).

Plusieurs concepts formels dyadiques (classiques) résultent d'un contexte formel ternaire $\mathcal{K} = (\mathcal{O}, \mathcal{P}, \mathcal{C}, \mathcal{R})$, en considérant les différents produits cartésiens $\mathcal{O} \times \mathcal{P}$, $\mathcal{O} \times \mathcal{C}$ et $\mathcal{P} \times \mathcal{C}$.

Soit $X \subseteq \mathcal{O}$ et $A \subseteq \mathcal{P} \times \mathcal{C}$, l'opérateur de dérivation de suffisance $(.)^{\Delta_\mathcal{O}}$ ($\mathcal{O}$ pour l'objet) est donné comme suit :

$$X^{\Delta_\mathcal{O}} = \{(p, c) \in \mathcal{P} \times \mathcal{C} \mid \forall x \in X : (x, p, c) \in \mathcal{R}\}$$

$$A^{\Delta_\mathcal{O}} = \{x \in \mathcal{X} \mid \forall (p, c) \in A : (x, p, c) \in \mathcal{R}\}$$

Les concepts formels dyadiques induits, en utilisant l'opérateur de dérivation $(.)^{\Delta_\mathcal{O}}$, sont de la forme (X, A), $X \in 2^{\mathcal{O}}$, $A \in 2^{\mathcal{P} \times \mathcal{C}}$ tel que $X^{\Delta_\mathcal{O}} = A$ et $A^{\Delta_\mathcal{O}} = X$.

$(.)^{\Delta_\mathcal{P}}$ (respectivement. $(.)^{\Delta_\mathcal{C}}$) est défini de façon similaire, les concepts formels dyadiques induits sont de la forme $(P, Y) \subseteq 2^{\mathcal{P}} \times 2^{\mathcal{O} \times \mathcal{C}}$ (respectivement. $(C, Z) \subseteq 2^{\mathcal{C}} \times 2^{\mathcal{O} \times \mathcal{P}}$).

2.9 Application de l'analyse de concepts formels

Une dizaine d'années après la publication du papier fondateur en 1982 (Wille, 1982), l'ACF est restée restreinte à un cadre exclusivement théorique de par sa nature mathématique. Un petit groupe de chercheurs en Allemagne avaient alors développé des aspects purement algébriques.

Par la suite, l'ACF est devenue une communauté de recherche internationale avec des applications dans de nombreux domaines, comme la classification, la fouille de données, l'apprentissage, la construction d'ontologies, la recherche d'informations, etc.. Dans cette section, nous nous limitons à présenter les plus importants domaines d'utilisation de l'ACF, à savoir : la recherche d'informations et la représentation de connaissances.

2.9.1 Analyse de concepts formels et recherche d'information

La recherche d'information (RI) a été l'une des premières applications phare des treillis de concepts à la découverte de ressources. Les premiers travaux [27,71] ont étudié la possibilité d'utiliser les treillis de concepts comme support pour la recherche documentaire. Des collections de documents sont alors représentées sous la forme de contextes formels. Les objets du contexte sont des documents et les attributs sont les termes d'indexation de ces documents. Chaque concept du treillis correspondant est vu comme un couple formé par une requête, dont les mots clés sont les termes contenus dans l'intension du concept, et l'ensemble de documents pertinents pour cette requête sont les documents contenus dans l'extension du concept.

Le calcul de la réponse à une requête donnée revient à identifier, dans le treillis, le concept dont l'intension est identique à la requête. Les liens de spécialisation/généralisation entre les concepts permettent d'effectuer une recherche progressive dans le treillis. Cette façon de procéder suppose que la requête existe déjà dans le treillis. Pour assurer cette condition, des algorithmes de construction incrémentale de treillis de concepts sont utilisés pour l'insertion des requêtes dans un treillis déjà construit. De cette manière, un premier mode de recherche par treillis a été défini : la recherche par interrogation. La structure hiérarchique des treillis de concepts permet la définition d'un deuxième mode de recherche : la recherche par navigation.

2.9.1.1 La recherche par interrogation et par navigation

La recherche par interrogation est un mode de recherche qui est facilité par la mise en place d'algorithmes performants pour la construction incrémentale des treillis de concepts. La définition de requête consiste à spécifier directement les termes d'indexation qui décrivent les documents à trouver. La requête est ensuite insérée dans le treillis. La recherche des documents pertinents revient à localiser le concept le plus général incorporant les termes spécifiés dans la requête.

La recherche par navigation utilise le treillis de concepts formels, consiste à explorer la structure hiérarchique de treillis visualisée pas un diagramme de Hasse. Ce dernier offre une interface de navigation permettant de se déplacer d'un concept formel à un autre.

L'utilisation des treillis de concepts permet de définir deux modes de recherche, par interrogation et par navigation, qui peuvent être combinés. Le treillis de concepts sert d'espace de recherche commun et assure la cohérence des résultats de recherche des deux modes. De plus, le passage d'un mode à l'autre peut se faire à tout moment pendant la recherche. La recherche se ramène ainsi à effectuer une combinaison libre de (i) spécification directe de requête, résultant en un saut dans le concept le plus générale incorporant les termes de la requête, et (ii) la navigation libre en suivant les liens entre les concepts du treillis.

Carpineto et Romano [29] ont développé un système qui permet la recherche par interrogation et par navigation dans un treillis de concepts formels. Ce système est constitué de deux modules : GALOIS et ULYSSES. GALOIS est responsable d'organiser les informations extraites du corpus en une structure de treillis. Cependant, ULYSSES sert d'une interface graphique qui permet l'accès et la récupération de l'information contenue dans la structure construite à l'aide de la première composante.

2.9.1.2 Classement des résultats de la recherche par treillis

Des documents n'apparaissant pas dans le même concept formel que la requête peuvent être pertinents. Cependant, leurs pertinence est inférieure à ceux dans lesquels la requête apparait. De ce fait, un classement de l'ensemble de document en réponse s'impose pour refléter cette différence de pertinence. La structure hiérarchique des treillis a été exploitée pour déduire automatiquement un tel classement. Djouadi [41] a proposé une approche déterministe pour le classement des résultats d'une requête en s'appuyant sur la distance entre les concepts formels.

Alors que les poids des termes dans les documents constituent une information très importante, les précédentes approches de la RI basées sur le treillis de concepts formels ne considèrent que les relations booléennes. Pour prendre en compte les poids des termes lors d'une recherche dans un treillis, il a été proposé d'étendre les connexion de Galois au cas flou [80].

Une autre limite des systèmes présentés auparavant, est que les requêtes sont limitées à une forme conjonctive. Djouadi [40] a proposé d'utiliser les nouveaux opérateurs de dérivation décrits par Dubois et al. dans [47] afin d'exprimer la disjonction et la négation.

2.9.1.3 Approches récentes de recherche d'information par treillis

Pendant les dernières années, les approches de recherche d'information par treillis ont donné lieux à des méta-moteurs de recherche d'information. Ces méta-moteurs se placent entre les utilisateurs et les moteurs de recherche tels que Google et Yahoo. Ils ont pour rôle d'améliorer la présentation des résultats de recherche aux utilisateurs. Le fonctionnement général commence tout d'abord par une requête utilisateur qu'est passée à un moteur de recherche. Le résultat retourné par le moteur de recherche, sous forme de liens vers des documents et un ensemble de "synsets" (mots décrivant le document : souvent une ou deux lignes du document qui contiennent un ou plusieurs mots clés de la requête), est transformé sous la forme d'un contexte formel. Les objets sont les liens et les attributs sont formés par l'ensemble des éléments des synsets utilisés. Après construction du treillis correspondant à ce contexte, on propose à l'utilisateur le nouveau classement du résultat fourni par le treillis. Ce type d'approche est implémenté dans plusieurs systèmes opérationnels tels que CREDO [30], CREDINO [28], FooCA [76], Insighter [74], CRECHAINDO [87] et SearchSleuth [55].

Le Système FooCA est le seul système qui permet de visualiser le résultat retourné par le moteur de recherche, Google, sous la forme d'un contexte formel (une matrice) et le diagramme de Hasse du treillis correspondant à ce contexte. A travers cette visualisation, FooCA permet à l'utilisateur de modifier le contexte formel en supprimant des mots clés qui lui paraissent inutiles pour sa recherche. Les modifications apportées sont ensuite propagées au treillis pour la visualisation de la nouvelle classification.

CREDO est le premier système proposé. Après classification du résultat d'une requête sous la forme d'un treillis de concept, il affiche la partie supérieur du treillis (Iceberg) sous la forme d'une hiérarchie de liens similaire à celle des hiérarchies de fichiers dans les systèmes hiérarchiques. Chaque nœud de la hiérarchie est un mot qui indexe un certain nombre de documents dans le résultat. Ce nombre est aussi indiqué avec le mot clé.

Le système CreChainDo reprend le même principe et les mêmes fonctionnalités que CREDO et ajoute la possibilité d'interagir et de modifier le contexte formel. Il s'agit des fonctionnalités similaires à celle de FooCA mais avec soumission d'une nouvelle requête à Google et reconstruction du treillis après chaque modification apportée au contexte.

Le système Insighter diffère des autres par la représentation de chaque concept du treillis par un seul mot, le plus représentatif de l'intension du concept. Le choix du mot représentatif s'appuie sur une heuristique qui consiste à construire un graphe biparti complet 23 (ou biclique) entre les mots utilisés dans les documents et à effectuer la pondération des arcs du graphe selon les cooccurrences des paires de mots. Le mot le plus représentatif d'un ensemble de mots est celui dont les liens ont les poids maximaux.

Le système SearchSleuth fournit des résultats supplémentaires en exploitant les relations entre le concept requête et les autres concepts du treillis. Cette catégorisation consiste à répartir les concepts du treillis sur quatre ensembles. Le premier ensemble, affiché comme résultat principal, est formé par les liens aux documents qui forment l'extension du concept requête. Le deuxième et le troisième ensemble sont formés respectivement par les sous concepts et les super-concepts de la requête dans le treillis. Le quatrième ensemble est formé par la fratrie du concept requête dans le treillis, les concepts ayant un parent commun avec la requête. Contrairement au premier ensemble, les autres ensembles ne sont pas affichés mais sont accessibles via des liens de raffinement en fonction du besoin en spécialisation, généralisation ou la combinaison des deux.

2.9.2 Système d'information logique

Un Système d'Information Logique (SIL) reprend tous les termes de l'ACL. C'est donc un système permettant de combiner la navigation, l'interrogation, et la logique. Plusieurs fonctionnalités ont été développées pour adapter l'ACL à un SIL. La navigation à l'intérieur du treillis de concepts de l'ACL s'effectue maintenant pas-à-pas par l'utilisateur. Le treillis va en fait être calculé à chaque pas, afin de proposer les liens de navigation pertinents (raffinements) et les objets qui y sont rattachés, et ainsi permettre à l'utilisateur de se déplacer sur un nouveau concept particulier. L'utilisateur n'a donc plus besoin de connaissances à priori sur les données. Comme pour l'ACL, le SIL possède lui aussi une logique générique, ce qui lui permet d'être appliqué à n'importe quel domaine. Seulement, afin d'éviter aux développeurs de concevoir pour chaque domaine une logique propre et démontrable, des composants ont été développés. Ces composants, appelés foncteurs logiques, permettent de créer une base pour la conception d'une logique à vocation de système d'information. On trouvera par exemple le foncteur Interval, permettant les logiques du type in 1999...2000, ou String pour la gestion de chaînes de caractères. Plusieurs applications ont été développés grâce à ces SIL : LISFS permet la gestion d'un système de fichiers, CAMELIS l'applique sur un ensemble de données personnelles, GEOLIS [17] ajoute aux SIL une navigation géographique.

2.9.3 Représentation de connaissances

La représentation des connaissances et le raisonnement est un domaine de l'intelligence artificielle dont le but fondamental est de représenter les connaissances d'une manière à faciliter les inférences. Une bonne représentation des connaissances devrait avoir la capacité de stocker et de récupérer des connaissances avec précision et rapidité. Par conséquent, Le treillis de Galois est une structure efficace pour la représentation des connaissances.

Il existe de nombreux travaux exploitant l'analyse de concepts formels pour les ontologies. Ainsi, dans [110] Stumme et Maedche exploitent l'ACF pour fusionner deux ontologies en s'appuyant sur le contexte d'apparition des termes et ainsi rapprocher les termes ayant des contextes similaires. Sur un plan plus formel, Baader et al. [15], Sertkaya [108] et Rudolph [103], se sont intéressés aux liens entre l'analyse de concepts formels et les logiques de description, formalisme souvent utilisé pour la représentation des connaissances. Comme cet aspect constitue l'essentiel de notre travail, il sera décrit de manière plus détaillée dans la suite de ce document.

2.10 Conclusion

Dans ce chapitre, nous avons d'abord présenté un état de l'art de l'analyse de concepts formels en définissant les notions mathématiques sous-jacentes. Nous avons vu ensuite certaines propriétés algébriques des opérateurs de dérivation de Galois notamment la propriété de fermeture. Nous avons particulièrement mis l'accent sur les notions nécessaires aux treillis et aux implications d'attributs conjonctives. Le treillis est une façon d'analyser un contexte formel, les implications en sont un autre. Dans ce chapitre nous avons vu comment obtenir les implications d'attributs à partir d'un treillis de Galois ou directement à partir d'un contexte formel binaire.

Les implications d'attributs obtenues à partir d'un contexte formel binaire sont limitées à une sémantique conjonctive car elles s'appuient sur l'utilisation de l'opérateur de dérivation classique de Galois (C'est-à-dire l'opérateur de suffisance). Dans le chapitre suivant on va présenter l'extension possibiliste de l'ACF qui utilise 3 autres opérateurs de dérivation tel que l'opérateur de possibilité, l'opérateur de nécessité ainsi que l'opérateur dual de l'opérateur de suffisance. Ces opérateurs sont exploités dans le chapitre 5 afin de générer des implications avec une sémantique disjonctive.

CHAPITRE 3

Théorie des possibilités & Analyse de concepts formels

Sommaire

3.1 Introduction

Fort souvent, les informations manipulées dans la réalité présentent un caractère d'imprécision, d'incertitude, d'incomplétude, d'incohérence, de conflit, d'ambiguïté, d'erreur, etc... Dans la littérature, toutes les informations présentant un pareil caractère sont généralement regroupées sous la notion générique d'"information imparfaite". Afin de bien situer la notion d'incomplétude -objectif de ce travail- nous passerons d'abord en revue dans ce chapitre, les grandes familles d'imperfection qui sont rapportées dans la littérature à savoir : l'imprécision, l'incertitude et l'incomplétude. Nous présenterons par la suite les cadres théoriques permettant de modéliser les informations incomplètes. Les fondements de la théorie des possibilités, qui est le cadre théorique retenu dans le cadre de ce travail, sera présentée de manière détaillée.

L'analyse des concepts formels et la théorie des possibilités sont deux cadres théoriques qui abordent différentes préoccupations dans le traitement de l'information. L'ACF construit des concepts formels à partir d'une relation reliant les objets aux attributs qu'ils satisfont, tandis que la théorie des possibilités traite de la modélisation de l'incertitude épistémique (graduée) et de l'incomplétude.

Constatant que les quatre fonctions (ensemblistes) définies dans la théorie des possibilités déterminent effectivement toutes les positions relatives possibles de deux ensembles, nous donnons dans ce chapitre une interprétation (lecture) possibiliste de l'analyse de concepts formels qui permet d'établir trois nouveaux opérateurs de dérivation. Nous présentons aussi les compositions symétriques et asymétriques qui peuvent résulter de la combinaison des opérateurs ainsi établis. Nous présentons enfin un cas d'utilisation concret d'une composition symétrique qui permet de décomposer une relation en sous-contextes indépendants.

3.2 Information imparfaite

La réalité nous amène souvent à manipuler des informations imparfaites. Parmi les différents types d'imperfections qui peuvent altérer une information et qui sont rapportées dans la littérature, nous pouvons généralement trouver les aspects suivants : incomplétude, imprécision, incertitude, incohérence, conflit, ambiguïté (on ne sait pas à quoi se rapporte l'information), biais (erreur systématique), bruit (erreur aléatoire), etc..

Plusieurs auteurs ont cherché à analyser, répertorier et classifier ces différents cas d'imperfections. Il existe donc plusieurs taxonomies des types d'imperfection qui ont fait l'objet de nombreux travaux [23, 32, 83, 91, 109, 119]. Dans [23, 32, 83, 109], les auteurs distinguent trois types d'imperfection donnés ci-après (voir la figure 3.1).

- imprécision,
- incertitude,

- incomplétude.

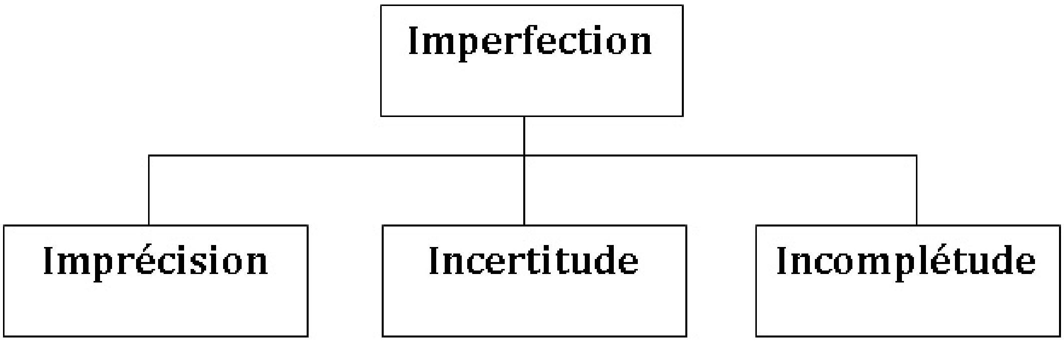

FIGURE 3.1 – Grands types d'imperfection.

De manière plus générale, l'imprécision concerne le contenu de l'information, l'incertitude concerne la validité de l'information, tandis que l'incomplétude fait référence à une situation partiellement informé ou à une situation d'ignorance totale. Mais avant de commencer la présentation des notions d'imprécision, d'incertitude et d'incomplétude, il convient de préciser certains aspects annexes. Une première distinction apparaît entre informations dites objectives, issues de mesures ou de capteurs ou plus généralement de l'observation directe de phénomènes, et celles dites subjectives exprimées par des individus et conçues sans le recours à l'observation directe du réel.

Une autre distinction importante doit être faite entre information singulière et information générique. L'information singulière concerne un fait particulier, résultant d'une observation ou d'un témoignage. L'information générique se réfère à une classe de situations (ce peut être une loi physique, un modèle statistique issu d'un ensemble représentatif d'observations, ou encore une connaissance de sens commun comme "les oiseaux volent").

3.2.1 Imprécision

L'imprécision concerne la difficulté d'exprimer clairement et précisément un état de la réalité par une proposition (par exemple "dans la salle il y a environ une centaine de personnes", ou "le poids de la table est d'environ 25 kg", ou "Jean est grand". Modéliser l'imprécision consiste à formaliser des quantificateurs tels que de "environ", ou des qualificatifs tels que "centaine" ou "grand" [106].

Une information est dite imprécise si elle est insuffisante pour permettre à l'agent de répondre à une question qu'il se pose.

Soit S l'ensemble des (descriptions des) états possibles du monde. Un sous-ensemble A de S est un événement, que l'on peut voir comme une proposition qui affirme $v \in A$.

La notion d'imprécision n'est pas absolue. Si on s'intéresse à l'âge d'une personne, le terme mineur est précis si le référentiel est $S = \{mineur, majeur\}$ et que la question est :

est-ce que la personne a le droit de vote ? En revanche, s'il s'agit de connaître son âge et si $S = \{0, 1, ..., 150\}$ (en nombre d'années), le terme mineur est (très) imprécis.

La forme type d'une information imprécise est $v \in S$ où A est un sous-ensemble d'un référentiel S contenant plus d'un élément. Une remarque importante est le fait que les éléments de A, vus comme valeurs possibles de v sont mutuellement exclusifs (car la grandeur n'a qu'une seule valeur). Donc une information imprécise prend la forme d'une disjonction de valeurs mutuellement exclusives (ne peuvent pas se produire en même temps). Par exemple dire que Ali a entre 20 et 25 ans, soit $v = age(Ali) \in \{20, 21, 22, 23, 24, 25\}$, c'est supposer $v = 20$ ou $v = 21$ ou $v = 22$ ou $v = 23$ ou $v = 24$ ou $v = 25$.

3.2.2 Incertitude

L'incertitude concerne un doute sur la validité d'une connaissance. Elle est due à la fiabilité de l'observateur peu sûr de lui ou prudent qui ne peut pas déterminer la valeur de vérité de la connaissance. Exemple : "je crois que dans la salle, il y a 100 personnes" [106].

Une information est dite incertaine pour un agent lorsque l'agent ne sait pas si cette information est vraie ou fausse. Une information élémentaire exprimée par une proposition est modélisée par un sous-ensemble de valeurs possibles, de la forme $v \in A$, et on affecte à cette information un marqueur d'incertitude. Ce marqueur se situe au méta-niveau par rapport aux informations elles-mêmes. Il peut être numérique ou linguistique. Par exemple, considérons les phrases :

- La probabilité pour que l'opération prenne plus d'une heure est 0,7
- Il est très possible qu'il neige demain
- Il n'est pas absolument certain que Ali vienne à la réunion

La représentation la plus courante de l'incertitude consiste à attribuer à chaque proposition ou événement A, sous-ensemble de S, un nombre $g(A)$ dans l'intervalle unité. $g(A)$ mesure la confiance de l'agent dans la vérité de la proposition $v \in A$. Cette proposition n'est par convention que vraie ou fausse, même si l'agent peut ignorer cette valeur de vérité. Les conditions suivantes sont naturellement requises : $g(\emptyset) = 0$; $g(S) = 1$; ainsi que la monotonie dans l'inclusion :

$$\text{Si } A \subset B \text{ alors } g(A) \leq g(B)$$

En effet, la proposition contradictoire $\emptyset$ est impossible, et la tautologie S est certaine. De plus si A est plus spécifique que B (et donc l'implique), l'agent ne peut pas avoir plus confiance en A qu'en B. Quand S est infini, on ajoute des propriétés de continuité par rapport à des suites monotones d'ensembles. Une telle fonction g est appelée tantôt capacité, tantôt mesure floue, ou encore fonction de plausibilité. Des conséquences importantes de ces postulats sont :

$$g(A \cap B) \leq min(g(A), g(B))$$

$$g(A \cup B) \geq max(g(A), g(B))$$

Une idée naturelle est alors de s'intéresser aux mesures de confiance g telles que $g(A \cup B)$ ne dépendent que de $g(A)$ et de $g(B)$.

Les différents cadres classiques de représentation et de manipulation de l'incertitude et l'imprécision sont comme suit :

- la théorie des sous-ensembles flous (présentée dans la section 2.7.1.1)qui fournit des outils simples et bien adaptés à la représentation d'informations imprécises ;
- la théorie des probabilités, des possibilités et des fonctions de croyance pour la gestion de l'incertitude (ces théories sont présentées dans la section 3.3).

3.2.3 Incomplétude

L'incomplétude correspond à l'idée d'information manquante et l'ignorance totale, c'est-à-dire l'impossibilité d'obtenir certains renseignements. il s'agit d'une absence de connaissance ou d'une connaissance partielle. Elle est due à une incomplétude dans les données, à l'absence d'une connaissance explicite ou à l'existence d'une connaissance générale.

Par exemple, des informations incomplètes apparaissent dans les bases de données relationnelles, lorsqu'un fait (tuple) doit être inséré dans une relation, et les valeurs de certaines colonnes obligatoires sont manquantes. Les informations incomplètes sont représentées par le symbole " ?", différent des symboles du domaine de la base de données. À titre d'exemple, considérons la base de données des employés dans la table 3.1 :

TABLE 3.1 – Base de données des employés

Employés	âge	adresse	numéro de téléphone
Karim	24	? - Alger	?
Lisa	65	?	123
Samir	17	Rue Colonel Amirouche - Tizi Ouzou	456

Dans la table 3.1 le numéro de téléphone de Karim est inconnu (ignorance totale), tout comme l'adresse personnelle de Lisa. Pour ce qui est de l'adresse personnelle de Karim, on sait qu'il habite à Alger mais on ignore le nom de la rue (ignorance partielle), par contre le troisième tuple a des informations complètes. Il existe de nombreuses raisons pour de telles informations manquantes, par ex. l'insertion a été effectuée via une vue, ou le tuple incomplet provient d'une autre base de données qui n'enregistre pas ces champs.

3.2.4 Interdépendance et non-exclusivité

Il convient de constater que l'imperfection dans les données peut prendre plusieurs formes non exclusives l'une de l'autre. A titre d'exemple, nous pouvons avoir une donnée imprécise et incertaine à la fois.

D'autre part, les différents types d'imperfections peuvent être interdépendants. En effet, les incomplétudes entraînent des incertitudes. Les imprécisions peuvent de même être associées à des incomplétudes, et elles engendrent des incertitudes au cours de leurs manipulations [23].

De ces dépendances peut naître de l'ambiguïté. L'ambiguïté est issue de problèmes mélangeant à la fois de l'incertitude et de l'imprécision et elle peut être l'effet d'un conflit (ou d'un désaccord) dans l'information, ou encore d'un problème de spécificité des informations utilisées.

3.3 Traitement de l'incomplétude : les cadres théoriques

Pendant longtemps, on n'a considéré que le cadre probabiliste était le seul cadre adapté à la représentation et à la manipulation de données imparfaites. Dans les quarante dernières années, d'autres théories de gestion de l'incomplet, l'imprécis et l'incertain on vu le jour, en raison notamment du constat que réduire l'imperfection d'une donnée à son caractère aléatoire était loin d'être satisfaisant. C'est dans cet esprit que l'intelligence artificielle, de par ses préoccupations de représentation des connaissances, a été amenée à s'intéresser à différents cadres de traitement de l'incomplétude [6] à savoir : la théorie des probabilités, mais aussi les théories plus récentes des possibilités et des fonctions de croyance.

Ces différentes approches se différencient par leur mode de représentation de connaissances et de description de l'incomplétude, ainsi que par les outils de traitement et de raisonnement permettant de manipuler les informations incomplètes.

Dans cette section nous présentons succinctement les principales approches qui permettent le traitement des informations incomplètes : la théorie des probabilités, la théorie des fonctions de croyance, et la théorie des possibilités.

3.3.1 Théorie des probabilités

La théorie des probabilités est la théorie la plus répandue pour modéliser les informations affectées par l'incertitude [55] (c'est-à-dire un et un seul singleton peut se produire), l'objectif étant de découvrir cet unique singleton. Dans le cadre de cette théorie, la connaissance liée à la réalisation des singletons x est représentée par une distribution de probabilités $Pr(.)$. Cette distribution associe à chaque singleton x_n de Ω, une valeur $Pr(x_n)$ désignant la probabilité pour que le singleton x_n soit l'unique singleton qui s'est produit [56] :

$$\begin{aligned} Pr : \Omega &\rightarrow [0, 1] \\ x_n &\rightarrow Pr(x_n) \end{aligned}$$

L'occurrence d'un événement $A \subseteq \Omega$ (A est un sous-ensemble de Ω) signifie que l'unique singleton qui s'est produit appartient à cet événement. Cette occurrence est caractérisée par une mesure $Pr(A)$ dite mesure de probabilité, qui est la somme des probabilités élémentaires affectées aux singletons formant cet événement. En d'autres termes, la valeur de cette mesure $Pr(A)$ peut être interprétée comme le degré de croyance pour que l'unique singleton qui s'est produit, appartienne au sous-ensemble A. Cette mesure de probabilité $Pr(A)$ est définie comme une mesure ensembliste qui vérifie les propriétés suivantes :

- $Pr(\emptyset = 0$ et $Pr(\Omega) = \sum_{x_n \in \Omega} Pr(x_n) = 1$
- $\forall A, B \subseteq \Omega$, $Pr(A \cup B) = Pr(A) + Pr(B) - Pr(A \cap B)$
- $Pr(A) + Pr(\overline{A}) = 1$ avec $\overline{A}$ l'événement complémentaire de A.

La théorie des probabilités repose sur des bases mathématiques solides et riches et elle constitue un outil puissant pour représenter fidèlement les informations entachées par l'incertitude. Néanmoins, cette théorie ne permet pas la modélisation l'incertitude qualitative et l'ignorance totale. Afin de remédier aux limites de la théorie des probabilités, la théorie des possibilités a été introduite.

3.3.2 Théorie des fonctions de croyance

La théorie des fonctions de croyance, aussi appelée théorie de Dempster-Shafer, est issue des travaux de Dempster en 1967 sur les probabilités inférieures et supérieures [58], et ensuite reprise par Shafer sous le nom de la théorie de l'évidence [59]. Elle est devenue un outil très populaire en intelligence artificielle pour aborder les problèmes liés à la représentation des connaissances et à la prise de décision. Cette théorie traite le cas où l'univers des contenus informationnels Ω est exhaustif et exclusif et permet de modéliser à la fois l'incertitude et l'imprécision exprimée d'une façon ensembliste. La modélisation de cette forme d'imperfection se réalise au travers de deux fonctions, l'une dite de crédibilité $Cr(.)$ et l'autre dite de plausibilité $Pl(.)$, toutes deux dérivées d'une fonction de masses de croyance $m(.)$. Dans cette théorie, les fonctions de masses de croyance sont définies sur tous les sous-ensembles A (événements) possibles de l'univers Ω et pas seulement sur les singletons comme dans la théorie des probabilités. La fonction $m(.)$ de masses de croyance est définie comme suit :

$$\begin{aligned} m : 2^{\Omega} &\rightarrow [0, 1] \\ A &\rightarrow m(A) \end{aligned}$$

où $m(A)$ désigne la croyance en la réalisation de l'évènement A : c'est donc la croyance que le singleton unique qui s'est produit, appartienne effectivement à cet évènement A. Par ailleurs, cette fonction $m(.)$ vérifie les propriétés suivantes :

- $m(\emptyset) = 0$
- $\sum_{A \subseteq \Omega} m(A) = 1$

Il est important de noter que la masse de croyance est affectée à chaque sous-ensemble de singletons A d'une façon globale, et qu'il n'est pas possible de la répartir sur les singletons composant l'événement A. Ainsi, à partir de la fonction $m(.)$, on définit respectivement les fonctions de crédibilité $Cr(.)$ et de plausibilité $Pl(.)$ comme suit :

$$Cr(A) = \sum_{B \subseteq A} m(B)$$

$$Pr(A) = \sum_{B \cap A \neq \emptyset} m(B) = 1 - Cr(\overline{A})$$

où $\overline{A}$ représente l'événement complémentaire de A. La fonction de crédibilité $Cr(A)$ peut être interprétée comme la croyance minimale en la réalisation de A tandis que la fonction de plausibilité $Pl(A)$ mesure la croyance avec laquelle on ne doute pas de la réalisation de A et peut être interprétée comme la croyance maximale en la réalisation de A. En effet, cette capacité de la théorie de Dempster-Shafer pour représenter le doute entre la réalisation des singletons permet de traiter des informations incertaines de nature probabiliste ainsi que certains types de l'imprécision, mais avec une interprétation différente de la théorie des probabilités (i.e. ensembliste). Néanmoins, cette théorie comme la théorie des probabilités, ne permet pas de traiter les informations ambigües.

3.3.3 Théorie des possibilités

Lotfi Zadeh a introduit la théorie des possibilités dans [118]. Cette théorie propose un cadre dans lequel les connaissances imprécises, incertaines et incomplètes peuvent coexister et être traitées conjointement. Elle permet donc de manipuler des informations imprécises et dont l'incertitude associée est de nature non probabiliste. C'est donc une théorie efficace pour la manipulation de l'information ambiguë [53] et [54].

Etant donné que la théorie des possibilités constitue le cadre que nous avons retenu dans ce travail de recherche, un intérêt particulier lui sera porté. En conséquent, nous lui consacrerons la section suivante toute entière.

3.4 Fondements théoriques de la théorie des possibilités

La théorie des possibilités fournit un cadre pour modéliser l'incomplétude. Cette théorie examine dans quelle mesure un événement est possible et dans quelle mesure on est certain de la réalisation d'un événement. La théorie des possibilités fait intervenir une mesure de possibilité Π et une mesure de nécessité N afin de formaliser ces deux évaluations subjectives.

Dans la théorie des possibilités, l'incertitude liée à la réalisation d'un événement $A \subseteq \Omega$ (A peut être un sous-ensemble algébrique de Ω, ou même un ensemble flou défini sur Ω) est exprimée par deux mesures ensemblistes : la mesure de possibilité et celle de nécessité [48]. L'utilisation de ces deux mesures dans la théorie des possibilités permet d'encadrer la probabilité de réalisation de cet événement A. Dans cette section, nous allons présenter les concepts de base de cette théorie tels que la distribution de possibilités, la mesure de possibilité, la mesure de nécessité.

3.4.1 Distribution des possibilités

Considérons un univers composé de N singletons $\Omega = x_1, x_2, ..., x_N$ et supposons que l'on se trouve dans un contexte d'incomplétude (c'est-à-dire qu'un seul singleton de Ω se produit mais nous ne le connaissons pas). La théorie des possibilités repose sur la notion de distribution de possibilités, notée $\pi(.)$, attribuant à chaque singleton x_n de Ω une valeur dans l'intervalle [0,1] qui encode notre état de connaissance ou croyance, sur la possibilité de la réalisation éventuelle de chaque singleton x_n de Ω. La valeur $\pi(x_n)$ encapsule nos connaissances liées à l'occurrence du singleton x_n. En d'autre termes, $\pi(x_n)$ représente dans quelle mesure il est possible que le singleton x_n soit l'unique singleton qui s'est produit. Dans ce contexte, deux cas extrêmes des connaissances sont donnés :

- Connaissance complète : $\exists! x_n \in \Omega, \pi(x_n) = 1$ et $\pi(x_m) = 0, \forall x_m \in \Omega, x_m \neq x_n$
- Ignorance totale : $\forall x_n \in \Omega, \pi(x_n) = 1$ (tous les singletons sont considérés comme tout à fait possible)

Dans le cas d'une connaissance complète, seul un singleton dispose de la valeur maximale de possibilité tandis que tous les autres singletons ont une possibilité d'occurrence nulle. En revanche, le cas de l'ignorance totale désigne la situation où nous ne disposons d'aucune connaissance permettant de favoriser (ou défavoriser) les différents singletons ; par conséquent, tous les singletons se voient attribuer le même niveau maximal de possibilité.

La valeur de possibilité maximale, $h(\pi) = max_{x_n \in \Omega}(\pi(x_n))$, appelée la hauteur de la distribution de possibilités, est un indicateur qui mesure le degré de consistance (ou compatibilité) avec les connaissances disponibles (représentées par la distribution de possibilités π) concernant la réalisation des singletons. Cet indicateur détermine le (ou les) singleton(s) ayant le degré de compatibilité le plus élevé avec les connaissances disponibles.

La distribution $\pi(.)$ est appelée distribution de possibilités normale (elle est dite consistante avec les connaissances disponibles) s'il existe au moins un singleton $x_n \in \Omega$ qui est totalement possible (i.e. $h(\pi) = 1$). Dans le cas contraire (i.e. $h(\pi) < 1$), cette distribution est dite non normalisée ou inconsistante avec les connaissances disponibles et, dans ce cas, un nouvel indicateur $Inc(\pi) \in [0, 1]$ peut être utilisé pour caractériser ce degré d'inconsistance de la distribution de possibilités π :

$$Inc(\pi) = \max_{x_n \in \Omega}(\pi(x_n)) = 1 - h(\pi)$$

3.4.2 Mesure de possibilité et mesure de nécessité

En se basant sur le concept de distribution de possibilités, deux mesures sont dérivées pour caractériser l'incertitude liée à l'occurrence de chaque sous-ensemble (aussi appelé un événement) $A \subseteq \Omega$: la mesure de possibilité Π et la mesure de nécessité N. Ces deux mesures sont définies par :

$$\begin{aligned} \Pi(\mathrm{A}) = & \max_{x_n \in A}(\pi(x_n)) \\ \mathrm{N}(\mathrm{A}) = & \ 1 - \Pi(\overline{\mathrm{A}}) \end{aligned}$$

où $\overline{A}$ est l'événement complémentaire de A (i.e. $A \cup \overline{A} = \Omega$ et $A \cap \overline{A} = \emptyset$).

La mesure de possibilité $\Pi(\mathrm{A})$ estime le niveau de consistance (avec les connaissances disponibles représentées par la distribution de possibilités π) de la réalisation de l'événement A. Elle représente dans quelle mesure il est possible que l'unique singleton x_n de Ω qui s'est produit, soit dans le sous-ensemble A. Si $\Pi(\mathrm{A}) = 1$ (resp. $\Pi(\mathrm{A}) = 0$), l'événement A est considéré comme un événement totalement possible (resp. impossible) à s'être produit (i.e. l'unique singleton x_n de Ω qui s'est produit est totalement possible qu'il soit dans A).

Notons que cette mesure de possibilité satisfait les propriétés suivantes :

- $\Pi(\emptyset) = 0$ et $\Pi(\Omega) = 1$
- $\forall A, B \subseteq \Omega, \Pi(\mathrm{A} \cup \mathrm{B}) = \max(\Pi(\mathrm{A}), \Pi(\mathrm{B}))$
- $\forall A, B \subseteq \Omega, \Pi(\mathrm{A} \cap \mathrm{B}) \leq \min(\Pi(\mathrm{A}), \Pi(\mathrm{B}))$

En revanche, la mesure de nécessité N(A) représente le niveau de certitude sur la réalisation de l'événement A, impliqué par les connaissances disponibles (représentées par la distribution de possibilités π). Elle représente dans quelle mesure on est certain que l'unique singleton x_n de Ω qui s'est produit, soit dans le sous-ensemble A. Si $\mathrm{N}(\mathrm{A}) = 1$ (resp. $\mathrm{N}(\mathrm{A}) = 0$), la réalisation de l'événement A est considérée comme totalement certaine (resp. incertaine) de s'être produite.

La mesure de nécessité satisfait les propriétés suivantes :

- $\mathrm{N}(\emptyset) = 0$ et $\mathrm{N}(\Omega) = 1$
- $\forall A, B \subseteq \Omega, \mathrm{N}(\mathrm{A} \cup \mathrm{B}) \geq \max(\mathrm{N}(\mathrm{A}), \mathrm{N}(\mathrm{B}))$
- $\forall A, B \subseteq \Omega, \mathrm{N}(\mathrm{A} \cap \mathrm{B}) = \min(\mathrm{N}(\mathrm{A}), \mathrm{N}(\mathrm{B}))$

L'approche possibiliste distingue trois états épistémiques extrêmes :

- la certitude que $x_n \in A$ est vrai : $\mathrm{N}(\mathrm{A}) = 1$, donc $\Pi(\mathrm{A}) = 1$
- la certitude que $x_n \in A$ est faux : $\Pi(\mathrm{A}) = 0$, donc $\mathrm{N}(\mathrm{A}) = 0$
- l'ignorance quand à $x_n \in A$: $\Pi(\mathrm{A}) = 1$, donc $\mathrm{N}(\mathrm{A}) = 0$

3.5 Interprétation possibiliste de l'ACF

L'analyse des concepts formels et la théorie des possibilités sont deux cadres théoriques qui abordent différentes préoccupations dans le traitement de l'information. L'ACF construit des concepts formels à partir d'une relation reliant les objets aux attributs qu'ils satisfont, Ce cadre théorique trouve des applications dans l'exploration de données. La théorie des possibilités traite de la modélisation de l'incertitude épistémique (graduée) et de l'incomplétude. Cette différence d'attention explique pourquoi les deux cadres théoriques ont été développés indépendamment depuis longtemps.

Cependant, il est possible de construire une analogie formelle entre l'ACF et la théorie des possibilités. En effet, les deux théories s'appuient fortement sur la comparaison des ensembles. Les quatre fonctions ensemblistes définies dans la théorie des possibilités déterminent effectivement toutes les positions relatives possibles de deux ensembles [47].

Il s'avère que l'opérateur de Galois classiquement utilisé en ACF définit l'ensemble des objets satisfaisant un ensemble d'attributs. Cette définition qui est à la base de la notion de concept formel, semble être la contrepartie de la fonction qui exprime une possibilité forte (ou garantie) définie dans la théorie des possibilités.

Sur la base de ce constat, l'idée maîtresse de ce travail de thèse consiste à utiliser deux autres fonctions ensembliste définies dans le cadre la théorie des possibilités au profit de le théorie de l'ACF et à leur conférer une sémantique dans cette théorie. Ceci nous conduit à considérer de nouvelles expressions algébriques de points fixes en termes de nouveaux opérateurs.

3.5.1 Généralisation possibiliste des opérateurs ensemblistes de dérivation

Rappelons que l'opérateur de suffisance $(.)^{\Delta}$ est l'opérateur de dérivation utilisé dans l'ACF (opérateur de Galois). À la fin du siècle dernier, cet opérateur a été identifié comme une alternative aux opérateurs de possibilité et de nécessité modale dans les logiques modales par Düntsch et Orlowska [58]. Ces modalités sont donc naturellement applicables à l'ACF. Les mêmes auteurs ont étudié un cadre algébrique pour ces opérateurs [59]. Quelques années plus tard, Dubois *et al.* [47,51] ont mis en évidence les liens étroits existant entre l'opérateur de suffisance dans l'ACF et l'une des quatre fonctions-ensemble de la théorie des possibilités [49], à savoir la fonction de possibilité garantie [50]. Ces résultats ont conduit à considérer trois autres opérateurs de dérivation dans l'ACF, A savoir, l'opérateur de possibilité (noté $(.)^{\Pi}$), l'opérateur de nécessité (noté $(.)^{\mathrm{N}}$) ainsi que l'opérateur dual de l'opérateur de suffisance (noté $(.)^{\nabla}$). Soit $A \subseteq \mathcal{P}$ et $X \subseteq \mathcal{O}$ les opérateurs de la forme X^{Π}, A^{Π}, X^{N}, A^{N}, X^{∇} et A^{∇} sont définis dans ce qui suit.

3.5.1.1 L'opérateur de possibilité

X^{Π} est l'ensemble des attributs satisfaits par au moins un objet dans X :

$$\begin{array}{rcl} X^{\Pi} & = & \{a \in \mathcal{P} \mid \exists x \in X, a\mathcal{R}x\} \\ & = & \{a \in \mathcal{P} \mid X \cap \mathcal{R}(a) \neq \emptyset\} \\ & = & \bigcup_{x \in X} \mathcal{R}(x) \end{array}$$

A^{Π} est l'ensemble des objets qui satisfaits au moins un attribut dans A :

$$\begin{array}{rcl} A^{\Pi} & = & \{x \in \mathcal{O} \mid \exists a \in A, x\mathcal{R}a\} \\ & = & \{x \in \mathcal{O} \mid A \cap \mathcal{R}(x) \neq \emptyset\} \\ & = & \bigcup_{a \in A} \mathcal{R}(a) \end{array}$$

Exemple 6. *Pour le contexte formel $\mathcal{K}_s$ illustré dans la table2.2, nous avons :*
$\{a_3, a_4, a_5\}^{\Pi} = \{x_1, x_2, x_3, x_4, x_5, x_6\}$,
puisque $\bigcup_{a \in \{a_3, a_4, a_5\}} \mathcal{R}(a) = \mathcal{R}(a_3) \cup \mathcal{R}(a_4) \cup \mathcal{R}(a_5) = \{x_1, x_3, x_4, x_5, x_6\} \cup \{x_1, x_4, x_5\} \cup \{x_1, x_2, x_5\}$.

Il est à noter que pour les singletons, $\{a\}^{\Pi} = \{a\}^{\Delta} = \mathcal{R}(a)$ et $\{x\}^{\Pi} = \{x\}^{\Delta} = \mathcal{R}(x)$.

3.5.1.2 L'opérateur de nécessité

X^{N} est l'ensemble des attributs que seuls les objets de X vérifient :

$$\begin{array}{rcl} X^{\mathrm{N}} & = & \{a \in \mathcal{P} \mid \forall x \in \mathcal{O}(a\mathcal{R}x \Rightarrow x \in X)\} \\ & = & \{a \in \mathcal{P} \mid \mathcal{R}(a) \subseteq X\} \\ & = & \bigcap_{x \notin X} \overline{\mathcal{R}}(x) \end{array}$$

A^{N} est l'ensemble des objets que seuls les attributs de A vérifient :

$$\begin{array}{rcl} A^{\mathrm{N}} & = & \{x \in \mathcal{O} \mid \forall a \in \mathcal{P}(x\mathcal{R}a \Rightarrow a \in A)\} \\ & = & \{x \in \mathcal{O} \mid \mathcal{R}(x) \subseteq A\} \\ & = & \bigcap_{a \notin A} \overline{\mathcal{R}}(a) \end{array}$$

Exemple 7. *Pour le contexte formel $\mathcal{K}_s$ illustré dans la table2.2, nous avons :*
$\{a_3, a_4, a_5\}^{\mathrm{N}} = \{x_1, x_5, x_6\}$,
puisque $\mathcal{R}(x_1), \mathcal{R}(x_5), \mathcal{R}(x_6) \subseteq \{a_3, a_4, a_5\}$ *et* $\mathcal{R}(x_2), \mathcal{R}(x_3), \mathcal{R}(x_4) \nsubseteq \{a_3, a_4, a_5\}$.

3.5.1.3 L'opérateur de suffisance duale

X^{∇} est l'ensemble des attributs qui ne sont pas satisfaits par au moins un objet en dehors de X :

$$\begin{aligned} X^{\nabla} &= \{a \in \mathcal{P} \mid \exists x \in \overline{X}, x\overline{\mathcal{R}}a\} \\ &= \{a \in \mathcal{P} \mid X \cup \mathcal{R}(a) \neq \mathcal{O}\} \\ &= \textstyle\bigcup_{x \notin X} \overline{\mathcal{R}}(x) \end{aligned}$$

A^{∇} est l'ensemble d'objets qui ne sont pas satisfaits par au moins un attribut en dehors de A :

$$\begin{aligned} A^{\nabla} &= \{x \in \mathcal{O} \mid \exists a \in \overline{A}, x\overline{\mathcal{R}}a\} \\ &= \{x \in \mathcal{O} \mid A \cup \mathcal{R}(x) \neq \mathcal{P}\} \\ &= \textstyle\bigcup_{a \notin A} \overline{\mathcal{R}}(a) \end{aligned}$$

3.5.1.4 Propriétés algébriques

Considérons dans ce qui suit que la notation $x\overline{\mathcal{R}}a$ désigne que l'objet x ne satisfait pas l'attribut a. En se basant sur l'hypothèse du monde fermé (CWA : Closed World Assumption), nous définissons le contexte formel complémentaire $\overline{\mathcal{K}}$ de $\mathcal{K}$ comme $\overline{\mathcal{K}} = (\mathcal{O}, \mathcal{P}, \overline{\mathcal{R}})$ où $\overline{\mathcal{R}} = \{(x, a) \in \mathcal{O} \times \mathcal{P} \mid (x, a) \notin \mathcal{R}\}$. Dans le reste de cette thèse, pour faciliter la lecture, lorsque les opérateurs de dérivation $(.)^{\Pi}$, $(.)^{\mathrm{N}}$, $(.)^{\Delta}$ sont appliqués au contexte complémentaire $\overline{\mathcal{K}}$ nous utiliserons la notation explicite $(.)^{\Pi}_{\overline{\mathcal{K}}}$, $(.)^{\mathrm{N}}_{\overline{\mathcal{K}}}$, $(.)^{\Delta}_{\overline{\mathcal{K}}}$.

Etant donné $X \subseteq \mathcal{O}$ et $\overline{X}$ son ensemble complémentaire (i.e., $\mathcal{O} \setminus X$), nous rappelons ci-après, quelques propriétés prouvées dans [39,57], nécessaires dans la suite de ce travail.

Proposition 2.

$P_1 : X^{\Delta}_{\overline{\mathcal{K}}} = \overline{(X^{\Pi})}$	$P_5 : X \subseteq (X^{\Pi})^{\mathrm{N}}$
$P_2 : X^{\Delta}_{\overline{\mathcal{K}}} = (\overline{X})^{\mathrm{N}}$	$P_6 : (X^{\mathrm{N}})^{\Pi} \subseteq X$
$P_3 :$ *If* $X_1 \subseteq X_2$ *then* $(X_1)^{\Pi} \subseteq (X_2)^{\Pi}$	$P_7 : X^{\mathrm{N}} = \overline{(\overline{X})^{\Pi}}$
$P_4 :$ *If* $X_1 \subseteq X_2$ *then* $(X_1)^{\mathrm{N}} \subseteq (X_2)^{\mathrm{N}}$	$P_8 : X^{\Pi} = ((X^{\Pi})^{\mathrm{N}})^{\Pi}$

P_1 et P_2 expriment l'opérateur de possibilité et de nécessité en termes de suffisance pour le contexte complémentaire. P_7 exprime la dualité entre X^{Π} et X^{N}. P_3 et P_4 établissent que les opérateurs de possibilité et de nécessité sont isotones. P_5 affirme que l'opération combinée $\mathrm{N} \circ \Pi$ et extensive, tandis que P_6 affirme que l'opération combinée $\Pi \circ \mathrm{N}$ est contractif. P_8 exprime que $\Pi \circ \mathrm{N}$ est un opérateur idempotent. Ces propriétés sont dualement vérifiées pour tout ensemble A d'attributs ($A \subseteq \mathcal{P}$).

Si l'on considère deux sous-ensembles d'objets X_1 et X_2, il est intéressant d'étendre certaines des propriétés ci-dessus à l'intersection $X_1 \cap X_2$ et l'union $X_1 \cup X_2$ de deux sous ensembles X_1 et X_2 comme suit.

Propriété 1. *$\forall X_1, X_2 \in \mathcal{O}$,*

- $(X_1 \cup X_2)^{\mathrm{N}} \supseteq (X_1)^{\mathrm{N}} \cup (X_2)^{\mathrm{N}}$
- $(X_1 \cup X_2)^{\Pi} = (X_1)^{\Pi} \cup (X_2)^{\Pi}$
- $(X_1 \cup X_2)^{\Delta} = (X_1)^{\Delta} \cap (X_2)^{\Delta}$
- $(X_1 \cup X_2)^{\nabla} \subseteq (X_1)^{\nabla} \cap (X_2)^{\nabla}$
- $(X_1 \cap X_2)^{\mathrm{N}} = (X_1)^{\mathrm{N}} \cap (X_2)^{\mathrm{N}}$
- $(X_1 \cap X_2)^{\Pi} \subseteq (X_1)^{\Pi} \cap (X_2)^{\Pi}$
- $(X_1 \cap X_2)^{\Delta} \supseteq (X_1)^{\Delta} \cup (X_2)^{\Delta}$
- $(X_1 \cap X_2)^{\nabla} = (X_1)^{\nabla} \cup (X_2)^{\nabla}$

3.5.1.5 Généralisation des opérateurs possibilistes aux contextes formels flous

Dans [40] les auteurs ont proposé d'étendre les opérateurs $((.)^{\Pi})$, $((.)^{\mathrm{N}})$ et $((.)^{\nabla})$ à des contexte formels flous (relations multi-valués). La généralisation floue des opérateurs de possibilité, de nécessité et de suffisance duale découle naturellement de leur définition logique (donnée respectivement en section 3.5.1.1, section 3.5.1.2 et section 3.5.1.3). Ainsi, étant donné que les expressions des opérateurs de suffisance et de nécessité sont basées sur des inclusions, leur généralisation repose naturellement sur des implications floues. Les expressions logiques des opérateurs de possibilité et le dual de l'opérateur de suffisance exprimant que l'intersection des sous-ensembles est non vide, de ce fait, leur généralisation est naturellement donnée par la plus grande valeur sur a (ou x) d'une conjonction floue des degrés d'appartenance. Les différentes définitions sont données comme suit :

$$\widetilde{X}^{\Delta}(a) \quad \bigwedge_{x\in\mathcal{O}}(\widetilde{X}(x) \to \mathcal{R}(x,a))$$

$$\widetilde{X}^{\Pi}(a) = \quad \bigvee_{x\in\mathcal{O}}(\widetilde{X}(x) * \mathcal{R}(x,a))$$

$$\widetilde{X}^{\mathrm{N}}(a) \quad \bigwedge_{x\in\mathcal{O}}(\mathcal{R}(x,a) \to \widetilde{X}(a))$$

$$\widetilde{X}^{\nabla}(a) \quad \bigvee_{x\in\mathcal{O}}(\sim \widetilde{X}(x) * \sim \mathcal{R}(x,a))$$

Il y'a lieu de constater que la généralisation des opérateurs possibilistes aux contextes formels flous pose le problème du choix pertinent d'une implication foue $\rightarrow$ et d'une conjonction floue $*$ [39].

3.5.2 Composition des opérateurs possibilistes pour l'ACF

A partir des quatre operateurs ensemblistes, $((.)^{\Delta})$ $((.)^{\Pi})$, $((.)^{N})$ et $((.)^{\nabla})$, il est possible de construire 16 compositions d'opérateurs. Nous présentons dans cette section certaines d'entre elles.

3.5.2.1 Composition symétrique

Dans [51] les auteurs ont proposé une nouvelle connexion de Galois, définie sur la base de la composition symétrique $(.)^{N \circ N}$ de l'opérateur possibiliste $(.)^{N}$. S'inspirant de la définition d'un concept formel, les auteurs ont proposé de considérer les paires (X, A) tel que $X^{N} = A$ et $A^{N} = X$. Il a été prouvé [37, 51] que les paires (X, A) tel que $X^{N} = A$ et $A^{N} = X$ forment des sous-contextes indépendants. (c'est-à-dire qu'ils n'ont pas d'objets communs et pas d'attributs communs) et sont donc intéressants pour la décomposition d'un contexte formel en sous contextes indépendants. Cela s'exprime par la propriété suivante : Ainsi, (X, A) et $(\overline{X}, \overline{A})$ sont deux sous-contextes indépendants dans $\mathcal{R}$, dans le sens où il n'y a pas de paire *objet \ attribut* (x, a) du contexte $\mathcal{R}$ dans $X \times \overline{A}$ ni en $\overline{X} \times A$. Ceci est aisément prouvé à l'aide de la proposition suivante :

Proposition 3. *Les propriétés suivantes des paires* (X, A) *sont équivalentes*

- $X^{N} = A$ *et* $A^{N} = X$
- $\overline{X}^{N} = \overline{A}$ *et* $\overline{A}^{N} = \overline{X}$
- $X^{\Pi} = A$ *et* $A^{\Pi} = X$
- $\mathcal{R} \subseteq (X \times A) \cup (\overline{X} \times \overline{A})$

La proposition ci-dessus n'implique aucune revendication de minimalité dans la propriété d'inclusion 4 de la proposition ci-dessus. En particulier, la paire (x, a) la satisfait trivialement. Cependant, ce résultat conduit à une décomposition de $\mathcal{R}$ dans une union disjointe de sous-contextes indépendants minimaux. En effet, supposons que deux paires (X_1, A_1), (X_2, A_2) satisfont la proposition ci-dessus. Cela implique que, par exemple, la paire $(X_1 \cap X_2, A_1 \cap A_2)$ la satisfait (on peut vérifier que $(X_1 \cap X_2)^{N} = A_1 \cap A_2$), et de même avec n'importe quel élément de la partition regroupant les deux partitions (X_1, X_1) et (X_2, X_2), ceci découle de la propriété 4 de la proposition ci-dessus, ce qui donne

$$\mathcal{R} \subseteq ((X_1 \times A_1) \cup (\overline{X_1} \times \overline{A_1})) \cap ((X_2 \times A_2) \cup (X_2 \times A_2) \cup (\overline{X_2} \times \overline{A_2}))$$

Où l'intersection du côté droit se résume à l'union des sous-contextes $(X_1 \cap X_2) \times (A_1 \cap A_2)$, $(X_1 \cap \overline{X_2}) \times (A_1 \cap \overline{A_2})$, $(\overline{X_1} \cap X_2) \times (\overline{A_1} \cap A_2)$, $(\overline{X_1} \cap \overline{X_2}) \times (\overline{A_1} \cap \overline{A_2})$,

La décomposition de $\mathcal{R}$ en sous-composantes minimales est obtenue en prenant l'intersection suivante

$$\bigcap_{(X,A):X_{\mathrm{N}}=A,A^{\mathrm{N}}=X} (X \times A) \cup (\overline{X} \times \overline{A}).$$

Exemple 8. *La table 3.2 présente un contexte formel. Les paires* $(\{x_6, x_7, x_8\}, \{c, d, e\})$*, ou* $(\{x_5, x_6, x_7, x_8\}, \{d, e\})$*, ou* $(\{x_2, x_3, x_4\}, \{g, h\})$ *sont des exemples de concepts formels, tandis que* $(\{x_5, x_6, x_7, x_8\}, \{a, b, c, d, e\})$*,* $(\{x_2, x_3, x_4\}, \{f, g, h\})$*,* $(\{x_1\}, \{i\})$ *sont des sous-contextes minimaux.*

TABLE 3.2 – Contexte formel et sous-contextes.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
a							×	
b					×	×		
c						×	×	×
d					×	×	×	×
e					×	×	×	×
f		×		×				
g		×	×	×				
h		×	×	×				
i	×							

3.5.2.2 Composition asymétrique

De la même manière que nous pouvons construire des compositions symétriques d'opérateurs de dérivation (exemple $(.)^{\Delta\circ\Delta}$ et $(.)^{\mathrm{N}\circ\mathrm{N}}$), il est aussi intéressant de construire des compositions asymétriques qui permettent d'offrir des propriétés topologiques intéressantes (i.e. fermeture et ouverture).

Ainsi, en utilisant les opérateurs possibiliste de nécessité et de possibilité, les paires (X, A) tel que $X = A^{\mathrm{N}}$ et $A = X^{\Pi}$ qui peuvent être définies de manière équivalente à $((X^{\Pi})^{\mathrm{N}}, X^{\Pi})$ qui sont des paires ouvert-fermé. Alors que les paires (X, A) telque $X = A^{\Pi}$ et $A = X^{\mathrm{N}}$ qui peuvent être définies de manière équivalente comme $((X^{\mathrm{N}})^{\Pi}, X^{\mathrm{N}})$ sont des paires de fermé-ouvert.

Topologiquement parlant, $(.)^{\mathrm{N}\circ\Pi} = ((.)^{\Pi})^{\mathrm{N}}$ est un opérateur de fermeture qui fournit une approximation haute, tandis que l'opérateur $(.)^{\Pi\circ\mathrm{N}} = ((.)^{\mathrm{N}})^{\Pi}$ est un opérateur d'ouverture qui fournit une approximation basse, de l'ensemble auquel on applique cet opérateur. En effet, $(X)^{\Pi\circ\mathrm{N}} \subseteq X \subseteq (.)^{\mathrm{N}\circ\Pi}$. Les concepts formels ainsi obtenus correspondront à des paires (X, A) tel que X sera un point fixe au sens de la fermeture et A sera un point fixe au sens de l'ouverture ($X = A^{\mathrm{N}}, A = X^{\Pi}$), ou inversement ($X = A^{\Pi}, A = X^{\mathrm{N}}$).

Propriété 2. *$\forall X, Y, Z \in \mathcal{O}$ tel que $X \subseteq Y \subseteq Z$ les propriétés suivantes sont valides :*

- *si $Z \subseteq Y^{\Delta\Delta}$ alors $Z^{\Delta\Delta} = Y^{\Delta\Delta}$*
- *si $Z \subseteq Y^{\mathrm{N}\Pi}$ alors $Z^{\mathrm{N}\Pi} = Y^{\mathrm{N}\Pi}$*
- *si $X \supseteq Y^{\Pi\mathrm{N}}$ alors $X^{\Pi\mathrm{N}} = Y^{\Pi\mathrm{N}}$*
- *si $X \supseteq Y^{\nabla\nabla}$ alors $X^{\nabla\nabla} = Y^{\nabla\nabla}$*

3.6 Conclusion

Dans ce chapitre, nous avons présenté le cadre général que constitue une information imparfaite. Après être rapidement revenu sur la notion de donnée imparfaite, nous avons ensuite rappelé les différents cadres classiques de représentation et de manipulation de l'incomplétude. Nous avons présenté les cadres théoriques capables de modéliser et de gérer des informations incomplètes à savoir : la théorie des probabilités, la théorie des fonctions de croyance, et la théorie des possibilités.

Nous avons porté un intérêt particulier à la théorie des possibilités car elle constitue une partie importante dans notre travail de recherche. Nous avons présenté l'analogie formelle entre l'ACF et la théorie des possibilités. Les deux théories s'appuient fortement sur la comparaison des ensembles. Les quatre fonctions définies dans la théorie des possibilités déterminent effectivement toutes les positions relatives possibles de deux ensembles. Ces propriétés seront utilisées dans le cadre de cette thèse.

CHAPITRE 4

Logiques de description

Sommaire

4.1 Introduction

Un courant de recherche très actif, dont l'origine est le système KL-ONE de rachman et Schmolze (1985) [24], s'est développé autour de l'idée que les connaissances du domaine sont représentées par des entités qui ont une description syntaxique à laquelle est associée une sémantique. Ce courant de recherche, qui s'est nourri d'études effectuées sur la logique des prédicats, les réseaux sémantiques et les langages de frames, a donné naissance à une famille de langages de représentation de connaissances appelés Logiques de

Description (LD). En anglais, ces logiques ont été désignées entre autres par les expressions Description Logic (DL), Terminological logic et Concept Language. La première expression semble être devenue leur nom définitif.

La recherche sur les logiques de description est très active aux État Unis, en Allemagne et en Italie, et connaît un intérêt grandissant en France. L'ouvrage de référence est toujours celui de Bernhard Nebel [88], tandis que ceux de Woods et Schmolze [115], Donini et al. [45] et Baader et .al [10] sont des synthèses très complètes.

Dans le formalisme des logiques de description, un concept permet de représenter un ensemble d'individus, tendis qu'un rôle représente une relation binaire entre individus. Un concept correspond à une entité générique d'un domaine d'application et un individu, à une entité particulière, instance d'un concept.

On peut définir la logique de description comme étant un formalisme de représentation des connaissances se divisant en deux parties bien distinctes : la partie terminologique, qui permet de définir un ensemble de concepts ainsi que leurs rôles et la partie assertionnelle, qui permet d'une part la création d'individus appartenant aux différents concepts de la partie terminologique et d'autre part aussi les opérations de classification et d'inférence. Par ailleurs, il est à noter que la classification, qui constitue la base du raisonnement en logique de description, s'appuie sur la relation de subsomption qui permet d'organiser les concepts et les rôles en une hiérarchie de termes.

Dans ce chapitre nous donnons une brève introduction aux LDs. Nous présentons les notions de base des LDs. Nous définissons la syntaxe et la sémantique des LDs, ainsi que la subsomption des inférences standard, la vérification de l'instance, et l'inférence non-standard.

Nous présentons aussi un état de l'art sur les travaux portant sur l'utilisation conjointe de l'analyse de concepts formels et la logique de description. Ces travaux qui permettent de tirer profit des avantages de chaque formalisme, peuvent être classés en deux catégories : la première catégorie concerne les travaux qui enrichissent l'ACF en empruntant des propriétés plus complexes semblables à celles des constructeurs de concept dans les LDs ; la deuxième catégorie concerne les travaux qui emploient les méthodes de l'ACF pour résoudre certains problèmes rencontrés dans la représentation de connaissances à l'aide de LDs.

4.2 Syntaxe

Pour définir des concepts dans une base de connaissances en LD, on commence par un ensemble N_C de prédicats unaires nommés noms de concepts et un ensemble N_R de prédicats binaires appelés noms de rôles et construit des descriptions de concepts plus complexes en utilisant les constructeurs fournis par le langage de description utilisé.

Concept primitif : les concepts primitifs servent de base à la construction de concepts définis. Ce type de concept représente le niveau le plus atomique de la connaissance en logique de description.

Concept défini : ils sont construits à partir des rôles qu'ils entretiennent avec des concepts primitifs ou avec d'autres concepts définis. Ce type de concept représente les niveaux de connaissances les plus complexes. Contrairement aux concepts primitifs, les concepts définis possèdent une définition complète et suffisante pour en déduire l'appartenance certaine d'une instance.

Un rôle : représente une relation binaire entre deux concepts. C'est une relation qui a pour fonction de caractériser le premier concept en précisant sa relation avec le second. Le concept non caractérisé par le rôle est appelé le co-domaine. Des restrictions de cardinalité peuvent être attribuées aux rôles. Par exemple, le rôle $enfant$, caractérisant le concept $Parent$, devrait être défini avec une cardinalité $n \geq 1$.

Les concepts et les rôles atomiques peuvent être combinés au moyen de constructeurs pour former respectivement des concepts et des rôles composés. Les différentes LDs se distinguent par les constructeurs qu'elles proposent. Dans Table 4.1, nous illustrons certaines familles de langage de logiques de description. Dans ce tableau, r représente un nom de rôle, A un nom de concept et C, D des descriptions de concepts. Les noms de concept spéciaux TOP ($\top$) et BOTTOM ($\bot$) dénotent respectivement le plus général et le plus spécifique. Intuitivement, l'extension de TOP inclut tous les individus possibles tandis que celle de BOTTOM est vide.

$A \sqcap D$ permet de faire la conjonction de deux concepts composés, $A \sqcup D$ permet de faire la disjonction de deux concepts composés, $\neg A$ est utilisé pour évoquer la négation d'un concept atomique. La restriction universelle et existentielle sont respectivement des expressions de la forme $\exists r.C$ et $\forall r.C$. Les concepts de la forme $\exists r.\top$ sont généralement appelées restrictions existentielles non typées (non qualifiées), écrit $\exists r$.

TABLE 4.1 – Langages de LD.

LD	$\top$	$\bot$	$\neg A$	$\neg C$	$C \sqcap D$	$C \sqcup D$	$\exists r.C$	$\forall r.C$
$\mathcal{EL}$	×				×		×	
$\mathcal{ELU}$	×				×	×	×	
$\mathcal{ALE}$	×	×	×		×		×	×
$\mathcal{ALC}$	×	×	×	×	×	×	×	×

Définition 23. *(Définition de concepts et GCI) Une définition de concepts est une expression de la forme $A \equiv C$, où A est un nom de concept, et C est une description de concepts complexe. Il attribue le nom de concept A à la description de concepts C. Les noms de concepts apparaissant sur le côté gauche d'une définition de concept sont appelés concepts définis et les autres sont appelés concepts primitifs. Une inclusion générale de*

concepts (GCI) est une expression de la forme $C \sqsubseteq D$, où C et D sont deux descriptions de concepts éventuellement complexes. Il énonce une relation de sous-concept/super-concept entre les deux descriptions de concepts.

Généralement, une base de connaissances de LD se compose de deux parties appelées partie terminologique (TBox en abrégé) et partie assertionnelles (factuelle) ou (ABox en abrégé).

Définition 24. *(TBox) Une TBox est un ensemble fini non ambigu de définitions de concepts, c'est-à-dire, chaque nom a au plus une définition. Elle est appelée TBox acyclique si aucune des définitions de concepts dans la TBox ne fait référence directement ou indirectement au nom qu'il définit. Sinon, elle est appelée TBox cyclique. Un ensemble fini de GCIs est appelé TBox général.*

La Figure 4.1 présente un exemple de TBox.

$TBox$
$Femelle \sqsubseteq \top \sqcap \neg Mâle$
$Mâle \sqsubseteq \top \sqcap \neg Femelle$
$Animal \equiv Mâle \sqcup Femelle$
$Humain \sqsubseteq Animal$
$Femme \equiv Humain \sqcap Femelle$
$Homme \equiv Humain \sqcap \neg Femelle$
$Mère \equiv Femme \sqcap \exists relationParentEnfant$
$Père \equiv Homme \sqcap \exists relationParentEnfant$
$MèreSansFille \equiv Mère \sqcap$ $\forall relationParentEnfant.\neg Femme$
$relationParentEnfant \sqsubseteq \top_R$

FIGURE 4.1 – Exemple de TBox

Définition 25. *(ABox) Une ABox est un ensemble fini d'assertions de concept et de rôle. Une assertion de concept est une expression de la forme $C(a)$ où C est une description de concepts et a est un nom d'individu. Une assertion de rôle est une expression de la forme $r(a, b)$ Où r est un nom de rôle, et a et b sont des noms d'individus. Nous représentons une base de connaissance comme $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ où $\mathcal{T}$ est une TBox et $\mathcal{A}$ est une ABox.*

La Figure 4.2 présente un exemple de ABox.

4.3 Sémantique

A l'instar de la logique classique, une sémantique est associée aux descriptions de concepts et de rôles : les concepts sont interprétés comme des sous-ensembles d'un domaine d'interprétation $\triangle^{\mathcal{I}}$ et les rôles comme des sous-ensembles du produit $\triangle^{\mathcal{I}} \times \triangle^{\mathcal{I}}$.

$ABox$
$Humain(Anne)$ $Femelle(Anne)$ $Femme(Sophie)$ $Humain(Robert)$ $\neg Femelle(Robert)$ $Homme(David)$ $relationParentEnfant(Sophie, Anne)$ $relationParentEnfant(Robert, David)$

FIGURE 4.2 – Exemple de ABox

Pour un concept C, $C^{\mathcal{I}}$ correspond au sous-ensemble des éléments du domaine $\triangle^{\mathcal{I}}$ qui appartiennent à l'extension de C, et pour un rôle r, $r^{\mathcal{I}}$ correspond au sous-ensemble des couples d'éléments du produit $\triangle^{\mathcal{I}} \times \triangle^{\mathcal{I}}$ qui appartiennent à l'extension de r.

Définition 26. *(Interprétation) Une interprétation $\mathcal{I} = (\triangle^{\mathcal{I}}, (.)^{\mathcal{I}})$ est la donnée d'un ensemble $\triangle^{\mathcal{I}}$ appelé domaine de l'interprétation et d'une fonction d'interprétation $(.)^{\mathcal{I}}$ qui fait correxpondre à un concept un sous-ensemble de $\triangle^{\mathcal{I}}$ et à un role un sous-ensemble de $\triangle^{\mathcal{I}} \times \triangle^{\mathcal{I}}$. L'extension de $(.)^{\mathcal{I}}$ à des descriptions de concepts est induite de façon inductive comme indiqué dans Table 4.2.*

TABLE 4.2 – Syntaxe et sémantique des constructeurs couramment utilisés.

nom du constructeur	syntaxe	sémantique
top	$\top$	$\triangle^{\mathcal{I}}$
bottom	$\bot$	$\emptyset$
négation atomique	$\neg A$	$\triangle^{\mathcal{I}} \backslash A^{\mathcal{I}}$
négation	$\neg C$	$\triangle^{\mathcal{I}} \backslash C^{\mathcal{I}}$
conjonction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
disjonction	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
restriction existentielle	$\exists r.C$	$\{x \in \triangle^{\mathcal{I}} \mid \exists y : (x, y) \in r^{\mathcal{I}} ety \in C^{\mathcal{I}}\}$
restriction universelle	$\forall r.C$	$\{x \in \triangle^{\mathcal{I}} \mid \forall y : (x, y) \in r^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}}\}$

Intuitivement, l'interprétation de $\top$ est le domaine $\triangle^{\mathcal{I}}$ tout entier tandis que celle de $\bot$ se réduit à l'ensemble vide. L'interprétation d'une conjonction (respectivement d'une disjonction) de concepts se ramène à l'intersection (respectivement l'union) des interprétations des concepts. L'interprétation de la négation d'un concept C se ramène au complémentaire de l'interprétation de C. L'interprétation de $(\forall r.C)$ précise le type du co-domaine du rôle r, tandis que celle de $(\exists r.C)$ affirme l'existence d'un couple d'éléments (x, y) en relation par l'intermédiaire du rôle r où C est le type de y.

L'interprétation de la construction $(\exists r)$ est un cas particulier de celle de $(\exists r.C)$, où $C \equiv \top$: $(\exists r)^{\mathcal{I}} = \{x \in \triangle^{\mathcal{I}} \mid \exists y : (x, y) \in r^{\mathcal{I}}\}$, il existe au moins un couple (x, y) tel que x et y soient en relation par l'intermédiaire de r (le type de y est ici quelconque).

Après avoir donné la sémantique des concepts et des rôles, nous pouvons définir la satisfaction des assertions et des GCIs données dans Table 4.3. Le symbole dans les GCIs $\sqsubseteq$ signifie la relation (est-un(e)). Autrement dit, l'inclusion de concepts $C \sqsubseteq D$ indique que chaque objet qui est dans l'interprétation de C est également dans l'interprétation de D de même, l'inclusion d'un rôle $r_1 \sqsubseteq r_2$ indique que chaque paire d'objets qui est dans l'interprétation de r_1 est également dans l'interprétation de r_2. L'appartenance des assertions aux concepts et rôles dans une ABox déclare simplement qu'un individu participe à un concept et qu'une paire d'individus participe à un rôle.

Table 4.3 – Satisfaction des assertions, définition de concept et GCI

nom du constructeur	syntaxe	sémantique
définition de concept	$A \equiv C$	$A^{\mathcal{I}} = C^{\mathcal{I}}$
inclusion de concept	$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
assertion de concept	$C(a)$	$a^{\mathcal{I}} \in C^{\mathcal{I}}$
assertion de rôle	$r(a, b)$	$(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$

Définition 27. *(Modèle). Nous disons qu'une interprétation $\mathcal{I}$ est un modèle d'une TBox $\mathcal{T}$ ssi elle satisfait toutes les définitions de concepts dans $\mathcal{T}$, c'est-à-dire que $A^{\mathcal{I}} = C^{\mathcal{I}}$ est valide pour tout $A \equiv C$ dans $\mathcal{T}$. Elle est un modèle de TBox $\mathcal{T}$ ssi elle satisfait toutes les GCIs dans $\mathcal{T}$, c'est-à-dire, $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ est valide pour tout $C \sqsubseteq D$ dans $\mathcal{T}$.*

4.4 Description de concepts dans $\mathcal{EL}$ et $\mathcal{ELU}$

Le langage $\mathcal{EL}$ est l'un plus simples parmi ceux de la famille des Logiques de description. En effet, il ne fournit que trois constructeurs qui sont le concept top $(\top)$, la conjonction $(\sqcap)$ et la restriction existentielle $(\exists)$.

Les descriptions de concepts $\mathcal{EL}$ sont définies inductivement comme suit.

- tous les noms de concepts $A \in N_C$ et le concept top $\top$ sont des descriptions concepts $\mathcal{EL}$,
- si C et D sont des descriptions de concepts $\mathcal{EL}$, alors $C \sqcap D$ est une description de concept $\mathcal{EL}$,
- Si C est une description de concept $\mathcal{EL}$ et $r \in N_R$ est un nom de rôle, alors $\exists r.C$ est une description de concept $\mathcal{EL}$.

D'autres langues fournissent plus que les trois constructeurs à la base du langage $\mathcal{EL}$. Nous pouvons évoquer le langage $\mathcal{ELU}$ qui possède un connecteur de disjonction en plus des connecteurs du langage $\mathcal{EL}$.

4.5 Modèles de raisonnement dans les LDs

Les LDs ne sont pas seulement utilisées pour stocker des définitions de concepts, et des assertions de concept ou de rôle, mais aussi pour leur raisonnement, c'est-à-dire en déduisant des connaissances implicites de la connaissance explicitement représentée.

4.5.1 Les inférences standard

L'un des problèmes d'inférence les plus traditionnels dans les LDs est le problème de décider des relations de sous-concept / super-concept entre les descriptions de concepts, à savoir la relation de subsomption. En utilisant la notion d'interprétation, cette relation peut être définie de la manière suivante :

Définition 28. *(subsomption) Un concept C est subsumé par un concept D (respectivement D subsume C), ce qui se note $C \sqsubseteq D$, si et seulement si $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ pour toute interprétation $\mathcal{I}$.*

La subsomption est à la base de la classification dans la logique de description. C'est une relation d'ordre partiel qui permet d'organiser les concepts et les rôles en une hiérarchie. Chaque concept possède une description qui permet de le comparer aux autres et ainsi de déterminer les relations de subsomption qu'il entretient avec les autres concepts (Même chose pour les rôles). La subsomption est une relation :

- reflexive : un concept C est donc subsumé par lui-même
- transitive : ce qui signifie que si un concept D subsume un concept C et que ce concept C subsume un concept E, alors le concept D subsume automatiquement le concept E
- antisymétrique : ce qui signifie que si un concept D subsume un concept C et que le concept C subsume à son tour le concept D, alors $C = D$

Les différentes relations de subsomption qu'entretient un concept permettront de déduire l'emplacement que ce concept devrait occuper dans la hiérarchie. Un concept maximal, représentant la totalité du domaine d'interprétation, est présent au sommet de la hiérarchie. Ce concept représente l'ensemble le plus général et subsume tous les autres, permettant ainsi d'obtenir un point de départ pour la taxonomie. Inversement, un concept minimal est présent au bas de la hiérarchie. Ce concept est subsume par tous les autres ; il représente en fait le vide.

En plus de la subsomption, il existe d'autres inférences standard importantes, qui sont formellement définies comme suit :

Définition 29. *(Satisfiabilité d'un concept, équivalence, incompatibilité)*

- *Un concept C est satisfiable ou cohérent si et seulement s'il existe une interprétation $\mathcal{I}$ telle que $C^{\mathcal{I}} \neq \emptyset$; C est non satisfiable ou incohérent sinon.*
- *Deux concepts C et D sont dits équivalents, ce qui se note $C \equiv D$, si et seulement si $C^{\mathcal{I}} = D^{\mathcal{I}}$ pour toute interprétation $\mathcal{I}$.*
- *Deux concepts C et D sont incompatibles ou disjoints si et seulement si $C^{\mathcal{I}} \cap D^{\mathcal{I}} = \emptyset$ pour toute interprétation $\mathcal{I}$.*

4.5.2 Les inférences non standard

En plus des inférences standards mentionnées dans la section précédente, des inférences dites non standard ont été introduites et étudiées dans la communauté LD. Dans le présent travail, nous nous intéressons exclusivement au calcul du plus petit commun subsumeur (en Anglais, lcs : least common subsumer). Il s'agit d'un cas d'inférence non standard que nous présentons ci-après.

Définition 30. *(plus petit commun subsumeur)*

Soit $\mathcal{L}$ une logique de description. Etant donnée une collection $C_1, ..., C_n$ de descriptions de concepts de $\mathcal{L}$, le plus petit commun subsumeur (ppcs) de $C_1, ..., C_n$ dans $\mathcal{L}$ est la description de concept la plus spécifique dans $\mathcal{L}$ qui subsume $C_1, ..., C_n$. C'est-à-dire, la description de concept D telle que :

1) $C_i \sqsubseteq D$ (D est un subsumeur commun)

2) Si E est une description de concept dans $\mathcal{L}$ satisfaisant $C_i \sqsubseteq E$ pour chaque $i = 1, ..., n$, alors $D \sqsubseteq E$

4.6 Utilisation conjointe de l'ACF et des LDs

L'analyse de concepts formels [66, 112] est un domaine des mathématiques qui vise à formaliser les notions d'un concept formel et d'une hiérarchie de concepts formels au moyen d'outils mathématiques. D'autre part, les Logiques de Description [10] sont un formalisme de représentation de connaissances basé sur la logique. Ce formalisme est utilisé pour représenter la connaissance conceptuelle d'un domaine d'application de manière structurée.

Bien que la notion de concept en tant que collection d'objets partageant certaines propriétés (attributs) et la notion de hiérarchie conceptuelle soient fondamentales à la fois pour l'ACF et pour les LDs, la façon dont les concepts sont décrits et obtenus diffèrent considérablement entre ces deux domaines de recherche. Dans les LDs, les concepts pertinents du domaine d'application sont formalisés par des descriptions de concepts, qui sont des expressions construites à partir de prédicats unaires (appelés concepts atomiques) et de prédicats binaires (appelés rôles atomiques) à l'aide de constructeurs fournis par le langage de la LD. Ensuite, dans une deuxième étape, ces descriptions de concepts sont utilisées pour décrire les propriétés (attributs) des individus (objets) se produisant dans le domaine, et les rôles sont utilisés pour décrire les relations entre ces individus (objets). En revanche, dans l'ACF, on part d'un contexte dit formel, qui dans sa forme la plus simple est un moyen de spécifier quels attributs sont satisfaits par quels objets. Un concept formel d'un tel contexte est une paire constituée d'un ensemble d'objets appelé extension et d'un ensemble d'attributs appelé intension de telle sorte que l'intension consiste exactement dans les attributs que les objets de l'extension ont en commun et l'extension comprend exactement les objets qui partagent tous les attributs dans l'intension. Il existe plusieurs travaux qui proposent l'utilisation conjointe de ces deux formalismes de représentation de connaissances afin de tirer profit des avantages de chaque formalisme, on peut les classer en deux catégories :

1) Les travaux qui enrichissent l'ACF en empruntant des propriétés plus complexes semblables à celles des constructeurs de concept dans les LDs [62, 97, 98, 101, 120].

2) Les travaux qui emploient les méthodes de l'ACF pour résoudre certains problèmes rencontrés dans la représentation de connaissances en Logique de description [2, 8, 11, 13–16, 33, 34, 36, 102, 104].

Cette section donne un aperçu de ces travaux.

4.6.1 Enrichissement de l'ACF avec les constructeurs des LDs

Logique d'attributs terminologique

Dans [97], Prediger a travaillé sur l'introduction de constructeurs logiques dans l'ACF. Elle a enrichi l'ACF avec des relations, des quantificateurs existentiels et universels, et la négation, en obtenant un langage de la famille $\mathcal{ALC}$ appelé logique d'attributs terminologique (terminologische Merkmalslogik). Dans le même travail, elle a aussi présenté des applications de son approche pour : i) enrichir des contextes formels avec de nouvelles connaissances, ii) obtenir un contexte formel binaire à partir d'un contexte formel multivalué.

Analyse de concepts relationnels

Dans [101], Rouane et al. ont proposé une adaptation de l'ACF aux LDs. Cette adaptation, appelée analyse de concepts relationnels, est destinée à analyser des objets décrits par des attributs relationnels dans l'exploration de données. L'approche est basée sur une collection de contextes formels, appelée famille de contextes relationnels, et les relations entre ces contextes. Les relations entre les contextes sont des relations binaires entre des paires d'ensembles d'objets qui appartiennent à deux contextes différents. Le traitement de ces contextes et de ces relations avec des méthodes d'analyse de concepts relationnels permet d'obtenir un ensemble de treillis de concepts (un pour chaque contexte d'entrée) de sorte que les concepts formels des différents treillis soient liés par des attributs relationnels similaires aux rôles dans les LD ou aux associations dans UML. Une caractéristique de cette approche est que les concepts formels et les relations entre les concepts formels de contextes différents peuvent être mappés dans les descriptions de concepst dans un sous-langage $\mathcal{ALE}$, qui est appelé$\mathcal{FL}^{-}\mathcal{E}$ [97]. Le langage $\mathcal{FL}^{-}\mathcal{E}$ permet la conjonction, la restriction de valeur, la restriction existentielle et les concepts haut et bas. Dans cette approche, après que les concepts formels et les relations aient été obtenus et mappés dans des descriptions de concepts $\mathcal{FL}^{-}\mathcal{E}$, le raisonnement de LDs est utilisé pour classer et vérifier la cohérence de ces descriptions.

4.6.2 Utilisation de l'ACF pour les LDs

Hiérarchie de subsomption des conjonctions de concepts

Dans [8], Baader a employé l'ACF pour un calcul efficace d'une hiérarchie de subsomption d'un ensemble de concepts de LDs. Plus précisément, il a employé l'exploration d'attributs pour le calcul de la hiérarchie de subsumption de toutes les conjonctions d'un ensemble de concepts de LD. La motivation principale pour ce travail était de déterminer l'interaction entre les concepts définis, qui ne pourraient pas être trouvés facilement en regardant juste la hiérarchie de subsumption des concepts définis. Afin d'expliquer ceci, l'exemple suivant a été donné

Exemple 9. *Soit* ***NoDaughter****,* ***NoSon*** *et* ***NoSmallChild*** *trois concepts définis comme suit :*
NoDaughter *: représente ces personnes qui n'ont aucune fille*
NoSon *: représente ces personnes qui n'ont aucun fils*
NoSmallChild *: représente ces personnes qui n'ont aucun petit enfant.*

Évidemment, il n'y a aucun rapport de subsumption entre ces trois concepts d'une part, et d'autre part :

$$(\boldsymbol{NoSon} \sqcap \boldsymbol{NoDaughter} \sqsubseteq \boldsymbol{NoSmallChild}),$$

*c'est-à-dire, si un individu (**a**) appartient à **NoSon** et à **NoDaughter**, il appartient également à **NoSmallChild**. Cependant, ceci ne peut pas être dérivé de l'information que (**a**) appartient à **NoSon** et à **NoDaughter** en regardant juste la hiérarchie de subsumption.*

L'exemple 9 démontre que des inférences d'exécution au sujet des individus peuvent être rendues plus rapides par la hiérarchie de subsumption non seulement pour des concepts définis, mais également pour toutes les conjonctions des concepts définis.

À cette fin, Baader a défini un contexte formel dont les attributs étaient les concepts définis dans la LD, et dont les objets étaient tous des contre-exemples possibles aux rapports de subsumption, c.-à-d., interprétations ainsi qu'un élément de domaine d'interprétation. Ce contexte formel a la propriété que son treillis de concept est isomorphe à la hiérarchie de subsumption requise, à savoir la hiérarchie de subsumption des conjonctions des concepts définis de la LD. Cependant, ce contexte formel a un inconvénient, c'est que l'algorithme standard de subsomption ne peut pas être employé par un expert du domaine dans l'exploration d'attributs. Afin de surmonter ce problème, l'approche a été reconsidérée dans [14] et un nouveau contexte formel qui a les mêmes propriétés mais pour qui un algorithme de subsomption a été présenté afin d'être employé comme expert.

Plus petit commun subsumeur (LCS)

Dans [13] Baader et Molitor ont utilisé l'ACF pour construire la hiérarchie ascendante (bottom-up) des descriptions de concepts. Dans l'approche bottom-up, l'expert ne définit pas directement les concepts de son domaine d'application, mais il donne des exemples typiques d'un concept, et la recherche du LCS repose sur une description de concept pour ces exemples. Le processus de calcul d'une telle description de concept consiste à calculer d'abord les concepts les plus spécifiques (MSC : most specific concepts) auxquels appartiennent les exemples donnés, et calculer ensuite le plus petit commun subsumeur (LCS : least common subsumer) de ces concepts. Ici le choix des exemples est crucial pour la qualité de la description de concept résultante. Si les exemples sont trop semblables, la description de concept résultante sera trop spécifique ; réciproquement, si la description de concept est trop distincte, la description de concept résultante sera trop générale. Afin de surmonter ceci, Baader et Molitor ont employé l'exploration d'attributs pour calculer la hiérarchie de subsumption de tous les LCS d'un ensemble donné de concepts. Dans cette hiérarchie on peut facilement voir la position du LCS ausaquels appartiennent les exemples donnés, et décident si ces exemples sont appropriés pour obtenir la description prévue de concept.

Exploration relationnelle (relational exploration)

Dans sa thèse de doctorat [103], Rudolph utilise l'ACF pour raffiner (refining) les bases de connaissances de la LD. Plus précisément, la LD se sert de la méthode interactive d'acquisition de connaissance de l'ACF, et ACF tire bénéfice des LDs en termes d'exprimer la connaissance relationnelle.

Dans [102, 103] le même auteur emploie la LD ($\mathcal{FLE}$) qui tient compte du constructeurs de conjonction, de la restriction existentielle, et de la restriction de valeur. Dans son travail précédent [105], il emploie la LD ($\mathcal{EL}$), qui tient compte du constructeur de conjonction et de la restriction existentielle.

Dans les deux cas, il définit la sémantique au moyen d'une paire spéciale de contextes formels appelés (binary power context family), qui sont employés pour exprimer des relations en ACF. Les (binary power context family), ont été également employés pour donner la sémantique aux graphs conceptuels.

Les attributs de ce contexte formel sont des descriptions de $\mathcal{FLE}$-concept, et les objets sont les éléments du domaine au-dessus duquel ces descriptions de concept sont interprétées. Dans ce contexte, un objet g est en relation avec un attribut m si et seulement si g est dans l'interprétation de m. Ainsi, une implication tient dans ce contexte formel si et seulement si dans le modèle donné, la description de concept résultant de la conjonction des attributs dans la partie prémisse de l'implication est subsumée par la description de concept de la conclusion. C'est ainsi que les implications dans les $\mathcal{FLE}$-contextes induisent des rapports de subsumption entre les descriptions de concept de $\mathcal{FLE}$.

Afin d'obtenir la connaissance complète au sujet des rapports de subsumption dans le modèle donné entre les concepts arbitraires de $\mathcal{FLE}$, Rudolph donne un algorithme d'exploration Multi-Étape (Multi-Step). Dans la première étape de l'algorithme, il commence par un $\mathcal{FLE}$-context dont les attributs sont les concepts atomiques se produisant dans une base de connaissance. Dans l'étape (i+1) d'exploration, il définit l'ensemble d'attributs comme l'union de l'ensemble d'attributs de la première étape et de l'ensemble de descriptions de concept formé par la restriction universelle de tous les attributs du contexte à l'étape (i) par rapport à tous les rôles atomiques, et l'ensemble de descriptions de concept formé par restriction existentielle de toutes les intensions de concept du contexte à l'étape (i) par rapport à tous les rôles atomiques.

Rudolph précise que, à une étape de l'exploration, il peut y avoir quelques descriptions de concept dans l'ensemble d'attributs qui sont équivalentes, c.-à-d., les attributs qui peuvent être réduits. À ce titre, il présente une méthode qu'il appelle (réduction empirique d'attribut (empiric attribute reduction). En principe, il est possible d'effectuer une infinité d'étapes d'exploration, ce qui signifie que l'algorithme ne se terminera pas. Afin de garantir l'arrêt, Rudolph limite le nombre d'étapes d'exploration. Après avoir effectué des étapes d'exploration, il est alors possible de décider de la subsomption (par rapport au modèle donné) entre toutes les descriptions de concepts $\mathcal{FLE}$ jusqu'à la profondeur du rôle (i) en utilisant simplement les bases d'implications obtenues à la suite des étapes d'exploration.

En outre, il caractérise également les cas où un nombre fini d'étapes sont suffisantes pour obtenir des informations complètes pour décider de la subsomption entre les descriptions de concepts $\mathcal{FLE}$ et la profondeur du rôle.

Autres approches

Bazin et Ganascia [16] adaptent les algorithmes de l'ACF classiques afin de créer des ensembles de définitions de concepts à partir des descriptions d'objet. Dans cet esprit, nous avons proposé, dans le cadre de cette thèse, une approche pour générer des GCIs (General Concept Inclusion) à partir de descriptions d'objets dans les logiques de description $\mathcal{EL}$ [2]. Cette approche considère tous les concepts jusqu'à la profondeur de rôle maximale au lieu d'utiliser une phase d'apprentissage différente pour chaque profondeur et permet d'obtenir une base minimale de GCIs. Cette approche est présentée en détail dans la section 7.2.1.

On peut remarquer que toutes les approches mentionnées ci-dessus considèrent les implications d'attributs limitées à leur forme conjonctives. Comme conséquence évidente, ces approches proposées ne permettent pas d'extraire des GCIs disjonctives. Cette limitation est une conséquence de l'utilisation de l'opérateur de dérivation classique de Galois, à savoir l'opérateur de suffisance qui génère des concepts formels dont la **sémantique est exclusivement conjonctive**.

4.7 Conclusion

Dans ce chapitre nous avons présenté les notions essentielles des LDs, telles que les TBox et les ABox.

Nous avons essentiellement présenté un état de l'art exhaustif sur les travaux qui mettent a profit l'utilisation de l'ACF pour les LDs et réciproquement. Nous avons classé ces travaux en deux catégories : la première catégorie concerne les travaux qui enrichissent l'ACF en empruntant des propriétés plus complexes semblables à celles des constructeurs de concept dans les LDs ; la deuxième catégorie concerne les travaux qui emploient les méthodes de l'ACF pour résoudre certains problèmes rencontrés dans la représentation de connaissances basée sur les LDs.

Nous avons porté un intérêt particulier sur la deuxième catégorie parce qu'elle constitue l'objet de notre travail de recherche. Nous avons constaté que toutes les approches qu'on trouve dans la littérature considèrent les implications d'attributs limitées à leur forme conjonctives. Comme conséquence évidente, ces approches proposées ne permettent pas d'extraire des GCIs disjonctives. Cette limitation est une conséquence de l'utilisation de l'opérateur de dérivation classique de Galois, à savoir l'opérateur de suffisance qui génère des concepts formels dont la sémantique est restreinte à une forme conjonctive.

Le prochain chapitre présente notre première contribution. Nous y proposons de considérer d'autres opérateurs de dérivations afin d'obtenir des implications d'attributs disjonctives, qui seront utilisées dans le chapitre 7 afin de générer des GCIs disjonctives.

CHAPITRE 5

Caractérisation de la base minimale d'implications d'attributs disjonctives

Sommaire

5.1 Introduction

En tirant parti de l'utilisation de paires d'opérateurs possibilistes dans l'ACF (opérateurs présentés dans la section 3.5), nous proposons dans ce chapitre une approche bien fondée pour extraire les implications d'attributs dites disjonctives. Nous qualifions cette approche de bien fondée car elle repose sur la caractérisation et l'obtention préalable d'une base minimale.

Notre approche repose, de manière originale, sur l'utilisation des opérateurs de dérivation définis dans la théorie des possibilités qui sont l'opérateur de possibilité $(.)^{\Pi}$ et l'opérateur de nécessité $(.)^{N}$. Rappelons que toutes les approches existantes utilisent classiquement l'opérateur de Galois $(.)^{\Delta}$. Nous donnons d'abord une nouvelle caractérisation de la composition asymétrique des opérateurs de dérivation de la nécessité-possibilité

$((.)^{\mathrm{N}\circ\Pi})$ "d'ouvert-fermé". Il en résulte un nouveau type de concepts formels (NΠ-paires). Les concepts formels organisés dans une hiérarchie (c'est-à-dire un ordre partiel), appelée NΠ-treillis, peuvent ensuite être extraits à partir d'un contexte formel. Nous exploitons le NΠ-treillis pour extraire des implications d'attributs disjonctives, à savoir des formules de la forme $a_1 \vee ... \vee a_n \longmapsto b_1 \vee ... \vee b_m$. En profitant de la structure algébrique de l'ensemble de tous les NΠ-paires obtenues en utilisant la composition asymétrique $((.)^{\mathrm{N}\circ\Pi})$, l'idée consiste à générer une base minimale d'implications d'attributs disjonctives. La minimalité est prouvée au moyen de "Pseudo-NΠ-Fermé".

5.2 Les NΠ-paires : ouvert-fermé

Nous commençons d'abord par donner les définitions d'opérateur de fermeture et d'ouverture. Ces définitions seront largement utilisées dans la suite de ce document.

Définition 31. *Un opérateur de fermeture défini sur un ensemble $\mathcal{U}$ est une fonction $\Psi : 2^{\mathcal{U}} \longrightarrow 2^{\mathcal{U}}$ satisfaisant $\forall\ U, V \in 2^{\mathcal{U}}$:*
- $U \subseteq \Psi(U)$
- $U \subseteq V \Longrightarrow \Psi(U) \subseteq \Psi(V)$
- $\Psi(\Psi(U)) = \Psi(U)$

Définition 32. *Un opérateur d'ouverture défini sur un ensemble $\mathcal{U}$ est une fonction $\Phi : 2^{\mathcal{U}} \longrightarrow 2^{\mathcal{U}}$ satisfaisant $\forall\ U, V \in 2^{\mathcal{U}}$:*
- $\Phi(U) \subseteq U$
- $U \subseteq V \Longrightarrow \Phi(U) \subseteq \Phi(V)$
- $\Phi(\Phi(U)) = \Phi(U)$

Désignons par "NΠ-paire", une paire formelle $\langle X, A\rangle$ tel que $X = A^{\Pi}$ et $A = X^{\mathrm{N}}$, où X (resp. A) sera appelé NΠ-extension (resp. NΠ-intension). Il peut être remarqué que les deux éléments X et A possèdent des propriétés topologiques duales. En effet, l'ensemble X est un ouvert tandis que A est un fermé donnant ainsi naissance à une paire "ouvert-fermé".

Désignons dans la suite de ce document, par $\mathfrak{B}_{\mathrm{N}\Pi}$ l'ensemble de toutes les NΠ-paires. Désignons aussi par $\mathfrak{B}_{\mathrm{N}\Pi}(Ext)$ (resp. $\mathfrak{B}_{\mathrm{N}\Pi}(Int)$) l'ensemble de toutes les NΠ-extensions (resp. NΠ-intensions). Considérant l'ensemble $\overline{\mathfrak{B}}_{\mathrm{N}\Pi}(Int) = \{\overline{A} \subseteq \mathcal{P} \mid A \in \mathfrak{B}_{\mathrm{N}\Pi}(Int)\}$. Par exemple, $(\{x_1, x_2, x_4, x_5\}, \{a_1, a_4, a_5\}) \in \mathfrak{B}_{\mathrm{N}\Pi}$ de $\mathcal{K}_s$, tandis que $\{a_1, a_4, a_5\} \in \mathfrak{B}_{\mathrm{N}\Pi}(Int)$, $\{x_1, x_2, x_4, x_5\} \in \mathfrak{B}_{\mathrm{N}\Pi}(Ext)$ et $\{a_2, a_3\} \in \overline{\mathfrak{B}}_{\mathrm{N}\Pi}(Int)$.

Rappelons que $x\overline{\mathcal{R}}a$ désigne que l'objet x ne satisfait pas l'attribut a. En se basant sur l'hypothèse du monde fermé (CWA : Closed World Assumption), nous définissons le contexte formel complémentaire $\overline{\mathcal{K}}$ de $\mathcal{K}$ comme $\overline{\mathcal{K}} = (\mathcal{O}, \mathcal{P}, \overline{\mathcal{R}})$ où $\overline{\mathcal{R}} = \{(x, a) \in \mathcal{O} \times \mathcal{P} \mid (x, a) \notin \mathcal{R}\}$. Rappelons aussi que $\overline{X}$ est l'ensemble complémentaire de $X \subseteq \mathcal{O}$ (i.e., $\mathcal{O} \setminus X$).

A titre d'exemple, la table 5.1 illustre le contexte formel complémentaire $\overline{\mathcal{K}}_S$ de $\mathcal{K}_S$ (donné dans la table 2.2).

TABLE 5.1 – Contexte formel complémentaire $\overline{\mathcal{K}}_{\mathcal{S}}$ de $\mathcal{K}_{\mathcal{S}}$

$\overline{\mathcal{R}}$	a_1	a_2	a_3	a_4	a_5
x_1	×	×			
x_2		×	×	×	
x_3	×			×	×
x_4					×
x_5	×	×			
x_6	×	×		×	×

La proposition 4 donne en premier lieu une caractérisation des NΠ-paires tandis la proposition 5 établit la structure algébrique de l'ensemble $\mathfrak{B}_{N\Pi}$.

Proposition 4. *Soient $X \in 2^{\mathcal{O}}$ et $A \in 2^{\mathcal{P}}$, $\langle X, A\rangle$ est une* NΠ*-paire si et seulement si $\langle \overline{X}, A\rangle$ est un concept formel dans $\overline{\mathcal{R}}$.*

Preuve de la Proposition 4: Elle est prouvée en utilisant les propriétés P_1 et P_7.

$$
\begin{array}{lll}
 & (\overline{X}, A) \text{ est un concept formel dans } \overline{\mathcal{K}} & \\
\Longleftrightarrow & \overline{X} = (A)^{\Delta}_{\overline{\mathcal{K}}} \quad \text{et} \quad A = (\overline{X})^{\Delta}_{\overline{\mathcal{K}}} & \text{(par definition)} \\
\Longleftrightarrow & \overline{X} = \overline{A^{\Pi}} \quad \text{et} \quad A = \overline{((\overline{X})^{\Pi})} \text{ in } \mathcal{K} & \text{(en utilisant } P_1) \\
\Longleftrightarrow & \overline{\overline{X}} = \overline{\overline{A^{\Pi}}} \quad \text{et} \quad A = \overline{\overline{X}}^{\mathrm{N}} & \text{(en utilisant } P_7) \\
\Longleftrightarrow & X = A^{\Pi} \quad \text{et} \quad A = X^{\mathrm{N}} & \\
\Longleftrightarrow & (X, A) \text{ est une N}\Pi\text{-paire dans } \mathcal{K}. &
\end{array}
$$

□

5.3 Treillis des NΠ-paires

Considérons sur l'ensemble $\mathfrak{B}_{N\Pi}$ des NΠ-paires un ordre partiel (noté $\preceq$). Etant donné deux NΠ-paires $(X_1, A_1) \preceq (X_2, A_2)$, nous définissons cet ordre partiel comme suit :

$$(X_1, A_1) \preceq (X_2, A_2) \Leftrightarrow X_1 \subseteq X_2 \Leftrightarrow A_1 \subseteq A_2$$

Avec cet ordre partiel, il existe pour chaque couple de NΠ-paires $((X_1, A_1), (X_2, A_2))$ un plus petit majorant qui est unique et peut être défini par les objets $X_1 \cup X_2$ (on parle de supremum). Symétriquement il existe, un unique plus grand minorant qui peut être défini par les attributs $A_1 \cap A_2$ (on parle d'infimum).

L'ensemble $\mathfrak{B}_{\mathrm{N\Pi}}$, muni de la relation d'ordre partiel ($\preceq$), forme un treillis complet, appelé NΠ-treillis et désigné par $\mathfrak{L}_{\mathrm{N\Pi}}$. La proposition suivante donne l'infimum (plus grande borne inférieure) et le supremum (plus petite borne supérieure) pour un sous-ensemble donné de $\mathfrak{L}_{\mathrm{N\Pi}}$.

Proposition 5. *Le supremum (noté $\coprod$) et l'infimum (noté $\prod$) d'un sous-ensemble (X_j, A_j) (j un indice) de $\mathfrak{L}_{\mathrm{N\Pi}}$ sont donnés par :*

$$\coprod_{j\in J}(X_j, A_j) = \Big(\bigcup_{j\in J} X_j, ((\bigcup_{j\in J} A_j)^{\Pi})^{\mathrm{N}}\Big);$$

$$\prod_{j\in J}(X_j, A_j) = \Big(((\bigcap_{j\in J} X_j)^{\mathrm{N}})^{\Pi}, \bigcap_{j\in J} A_j\Big).$$

Preuve de la Proposition 5: Ce résultat peut être prouvé en utilisant la proposition 4, en plus du fait que $\langle\overline{X}, A\rangle$ soit un concept formel de $\mathcal{K}(O, \mathcal{P}, \overline{\mathcal{R}})$, alors :

$$\begin{array}{rll}
& \prod\limits_{j\in J}(\overline{X}_j, A_j) = & (\bigcap\limits_{j\in J}\overline{X}_j, (\bigcup\limits_{j\in J} A_j)^{\Delta\Delta}_{\overline{\mathcal{K}}}) \\
\Longleftrightarrow & \prod\limits_{j\in J}(\overline{X}_j, A_j) = & (\overline{X}_1 \cap \ldots \cap \overline{X}_j, \overline{((\bigcup\limits_{j\in J} A_j)^{\Pi})}^{\Delta}_{\overline{\mathcal{K}}}) \\
\Longleftrightarrow & \coprod\limits_{j\in J}(\overline{\overline{X}}_j, A_j) = & (\overline{\overline{X}_1 \cap \ldots \cap \overline{X}_j}, \overline{\overline{((\bigcup\limits_{j\in J} A_j)^{\Pi})}}^{\mathrm{N}}) \\
\Longleftrightarrow & \coprod\limits_{j\in J}(X_j, A_j) = & (\overline{\overline{X_1}} \cup \ldots \cup \overline{\overline{X_j}}, ((\bigcup\limits_{j\in J} A_j)^{\Pi})^{\mathrm{N}}) \\
\Longleftrightarrow & \coprod\limits_{j\in J}(X_j, A_j) = & (X_1 \cup \ldots \cup X_j, ((\bigcup\limits_{j\in J} A_j)^{\Pi})^{\mathrm{N}}) \\
\Longleftrightarrow & \coprod\limits_{j\in J}(X_j, A_j) = & (\bigcup\limits_{j\in J} X_j, ((\bigcup\limits_{j\in J} A_j)^{\Pi})^{\mathrm{N}})
\end{array}$$

La preuve de l'infimum est obtenue de manière similaire. □

Exemple 10. *La figure 5.1 illustre le treillis $\mathfrak{L}_{\mathrm{N\Pi}}$ correspondant au contexte formel donné dans le table 2.2. Etant donné deux* NΠ*-paires de la figure 5.1, il est aisé de retrouver l'infimum et supremum à partir du treillis $\mathfrak{L}_{\mathrm{N\Pi}}$. Pour l'infimum, il faut suivre les chemins descendants des nœuds correspondants dans le treillis $\mathfrak{L}_{\mathrm{N\Pi}}$. Il y a toujours un plus haut point où ces chemins se rencontrent, (c'est l'infimum). De même, pour deux* NΠ*-paires il y a toujours un plus bas nœud, qui peut être atteint à partir des deux* NΠ*-paires via des chemins ascendants (c'est le supremum des deux).*

Le supremum et l'infimum des deux NΠ*-paires peuvent également être déterminés en utilisant la proposition 5. Par exemple, le supremum et l'infimum de* $(\{x_1, x_4, x_5\}, \{a_4\})$, $(\{x_1, x_2, x_5\}, \{a_5\})$ *sont obtenus comme suit :*

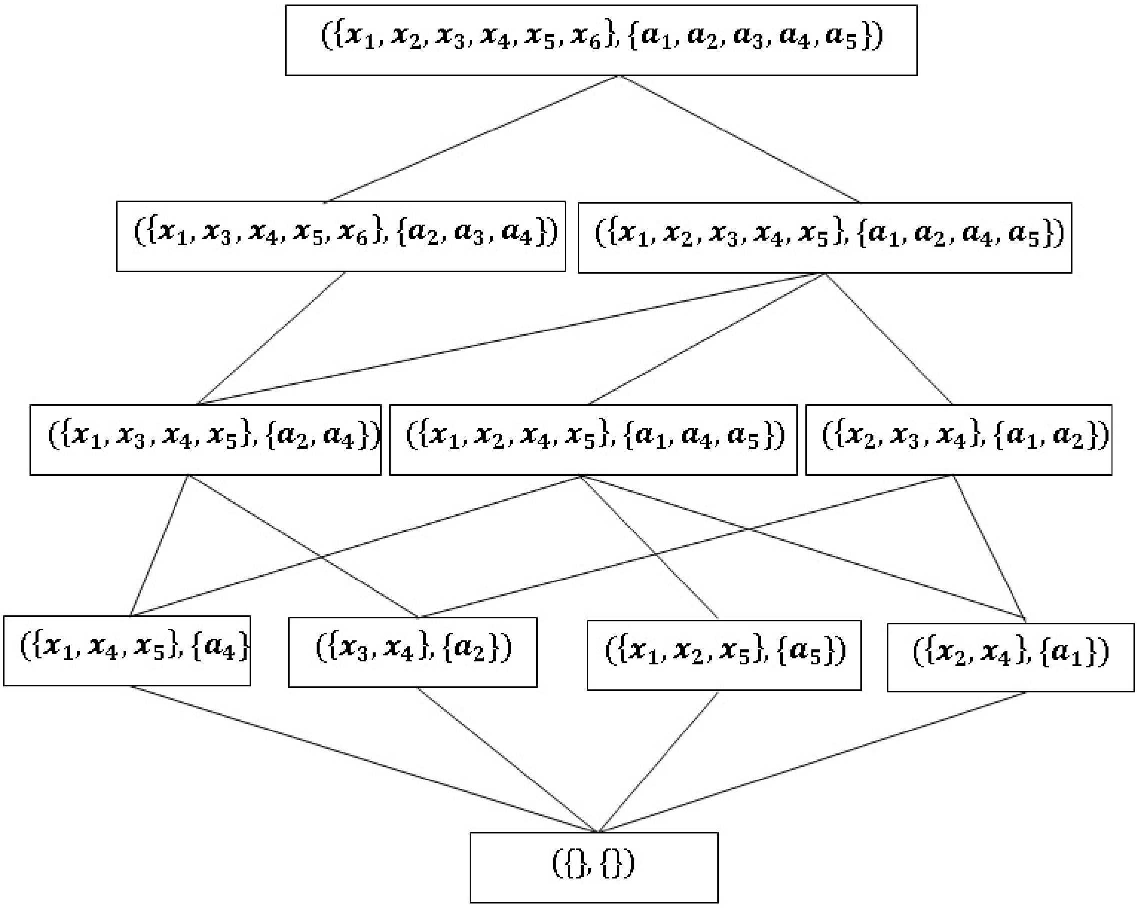

FIGURE 5.1 – Treillis $\mathfrak{L}_{\mathrm{N\Pi}}$ pour $\mathcal{K}_S$

$$\begin{aligned}
&\coprod\Big((\{x_1, x_4, x_5\}, \{a_4\}), (\{x_1, x_2, x_5\}, \{a_5\})\Big) = \\
&(\{\{x_1, x_4, x_5\} \cup \{x_1, x_2, x_5\}\}, \{\{\{a_4\} \cup \{a_5\}\}^{\Pi}\}^{\mathrm{N}}) = \\
&(\{x_1, x_2, x_4, x_5\}, \{\{a_4, a_5\}^{\Pi}\}^{\mathrm{N}}) = \\
&(\{x_1, x_2, x_4, x_5\}, \{a_1, a_4, a_5\}).
\end{aligned}$$

$$\begin{aligned}
&\prod\Big((\{x_1, x_4, x_5\}, \{a_4\}), (\{x_1, x_2, x_5\}, \{a_5\})\Big) = \\
&(\{\{\{x_1, x_4, x_5\} \cap \{x_1, x_2, x_5\}\}^{\mathrm{N}}\}^{\Pi}, \{\{a_4\} \cap \{a_5\}\}) = \\
&(\{\{x_1, x_5\}^{\mathrm{N}}\}^{\Pi}, \{\}) = \\
&(\{\}, \{\}).
\end{aligned}$$

□

Désignons maintenant par μ le mapping qui associe à chaque ensemble d'attributs $A \in 2^{\mathcal{P}}$, une NΠ-paire tel que :

$$\begin{array}{rccl}
\mu : & 2^{\mathcal{P}} & \rightarrow & \mathfrak{B}_{\mathrm{N\Pi}} \\
 & \mathrm{A} & \mapsto & \mu(A) = (A^{\Pi}, (A^{\Pi})^{\mathrm{N}})
\end{array}$$

La proposition suivante permet d'obtenir de manière explicite la valeur $\mu(A)$ pour un ensemble A d'attributs.

Proposition 6. *Soit $A \subseteq \mathcal{P}$, alors $\mu(A) = \coprod_{a \in A} \mu(\{a\})$*

Preuve de la Proposition 6: $A^{\Pi} = \bigcup_{a \in A} \{a\}^{\Pi}$ est obtenu directement par la définition de l'opérateur de possibilité, nous avons $\mu(A) = (A^{\Pi}, (A^{\Pi})^{\mathrm{N}}) \Leftrightarrow \mu(A) = (\bigcup_{a \in A} \{a\}^{\Pi}, (\bigcup_{a \in A} \{a\}^{\Pi})^{\mathrm{N}}) = \coprod_{a \in A} \mu(\{a\})$. □

5.4 Implications d'attributs disjonctives

La définition exprimant la satisfaction d'une implication d'attributs d'une manière intensionnelle a déjà été donnée (voir la définition 11). Dans ce qui suit, nous étendons cette définition aux implications d'attributs disjonctives de la forme $a_1 \vee ... \vee a_n \longmapsto b_1 \vee ... \vee b_m$ (désigné de manière équivalente par $\bigvee A \longmapsto \bigvee B$ avec $A = \{a_1, ..., a_n\}$, et $B = \{b_1, ..., b_m\}$).

Définition 33. *(Intension) Un sous-ensemble $M \subseteq \mathcal{P}$ satisfait une implication d'attributs disjonctive $\bigvee A \longmapsto \bigvee B$ (noté par $M \models \bigvee A \longmapsto \bigvee B$), si $B \nsubseteq M$ ou $A \subseteq M$. Nous disons que M satisfait un ensemble $\mathcal{D}$ d'implications d'attributs disjonctives (noté par $M \models \mathcal{D}$), si M satisfait chaque implication d'attributs disjonctive dans $\mathcal{D}$. Soit $M_i \in 2^{\mathcal{P}}, (i = 1..n)$, nous disons que $\bigvee A \longmapsto \bigvee B$ est valide dans l'ensemble $\{M_1, M_2, ..., M_n\}$ si $M_{i, i=1..n}$ satisfait $\bigvee A \longmapsto \bigvee B$.*

Dans le cadre d'un contexte formel, la satisfaction d'une implication d'attributs disjonctive est adaptée comme suit, en remplaçant les ensembles d'attributs par leurs extensions.

Définition 34. *(Extension) Soit $\mathcal{K} = (\mathcal{O}, \mathcal{P}, \mathcal{R})$ un contexte formel, $\mathcal{K} \models \bigvee A \longmapsto \bigvee B$ ssi $\forall x \in \mathcal{O}$, si $b_1 \notin \{x\}^{\Pi} \wedge ... \wedge b_m \notin \{x\}^{\Pi}$ alors $a_1 \notin \{x\}^{\Pi} \wedge ... \wedge a_n \notin \{x\}^{\Pi}$.*

Par exemple, le contexte formel $K_{\mathcal{S}}$ donné dans la table 2.2 satisfait l'implication d'attributs disjonctive $a_4 \longmapsto a_1 \vee a_5$ ($K_{\mathcal{S}} \models a_4 \longmapsto a_1 \vee a_5$), puisque tout objet qui ne satisfait ni a_1 ni a_5 ne satisfait pas a_4 non plus.

Il existe un moyen plus compact d'affirmer la satisfaction d'une implication d'attributs disjonctive basée sur l'opérateur de possibilité $(.)^{\Pi}$, Soit $\mathcal{K} = (\mathcal{O}, \mathcal{P}, \mathcal{R})$ un contexte formel. À savoir, $\mathcal{K} \models \bigvee A \longmapsto \bigvee B$ ssi pour chaque $x \in \mathcal{O}$, $B \nsubseteq \overline{\{x\}^{\Pi}}$ ou $A \subseteq \overline{\{x\}^{\Pi}}$. C'est parce que l'expression

$$\forall x \in \mathcal{O} \text{ si } b_1 \notin \{x\}^{\Pi} \wedge ... \wedge b_m \notin \{x\}^{\Pi} \text{ alors } a_1 \notin \{x\}^{\Pi} \wedge ... \wedge a_n \notin \{x\}^{\Pi}$$

peut être écrite comme équivalente à :

$$\forall x \in \mathcal{O} \text{ si } \{b_1, ..., b_m\} \subseteq \overline{\{x\}^{\Pi}} \text{ alors } \{a_1, ..., a_n\} \subseteq \overline{\{x\}^{\Pi}}.$$

Il est alors clair qu'un contexte formel $\mathcal{K}$ satisfait une implication d'attributs disjonctive $\bigvee A \longmapsto \bigvee B$ ssi chaque objet qui ne satisfait aucun attribut de B ne satisfait pas non plus aucun attribut de A. De façon équivalente, tout objet qui satisfait au moins un attribut dans A satisfait également au moins un attribut dans B. Une manière plus simple d'affirmer la satisfaction d'une implication d'attributs disjonctive basée sur l'opérateur de possibilité $(.)^{\Pi}$ est donné ci-après.

Proposition 7. *Soit un contexte formel $\mathcal{K} = (\mathcal{O}, \mathcal{P}, \mathcal{R})$ et $A, B \subseteq \mathcal{P}$, $\mathcal{K} \models \bigvee A \longmapsto \bigvee B$ ssi pour chaque $x \in \mathcal{O}$, $B \not\subseteq \overline{\{x\}^{\Pi}}$ ou $A \subseteq \overline{\{x\}^{\Pi}}$.*

Preuve de la Proposition 7: Par définition, un contexte formel $\mathcal{K} = (\mathcal{O}, \mathcal{P}, \mathcal{R})$ satisfait une implication d'attributs disjonctive $\bigvee A \longmapsto \bigvee B$ ssi $\forall x \in \mathcal{O}$:

$$\begin{aligned} & \text{Si } b_1 \not\subseteq \{x\}^{\Pi} \wedge ... \wedge b_m \not\subseteq \{x\}^{\Pi} \text{ alors } a_1 \not\subseteq \{x\}^{\Pi} \wedge ... \wedge a_n \not\subseteq \{x\}^{\Pi} \\ \Longleftrightarrow \quad & \text{Si } \{b_1, ..., b_m\} \subseteq \overline{\{x\}^{\Pi}} \text{ alors } \{a_1, ..., a_n\} \subseteq \overline{\{x\}^{\Pi}} \\ \Longleftrightarrow \quad & \text{Si } B \subseteq \overline{\{x\}^{\Pi}} \text{ alors } A \subseteq \overline{\{x\}^{\Pi}} \\ \Longleftrightarrow \quad & B \not\subseteq \overline{\{x\}^{\Pi}} \text{ ou } A \subseteq \overline{\{x\}^{\Pi}} \end{aligned}$$

□

A partir des opérateurs possibilistes, le théorème suivant établit un résultat important.

Théorème 1. *Soit un contexte formel $\mathcal{K}(\mathcal{O}, \mathcal{P}, \mathcal{R})$, $\mathcal{K} \models \bigvee A \longmapsto \bigvee B$ ssi $A \subseteq ((B)^{\Pi})^{\mathrm{N}}$.*

Preuve du Théorème 1:

1) Condition nécessaire

$$\begin{aligned} ((B)^{\Pi})^{\mathrm{N}} &= \Big(\bigcup_{x \in B^{\Pi}} \{x\}\Big)^{\mathrm{N}} \\ &= \overline{\Big(\overline{\bigcup_{x \in B^{\Pi}} \{x\}}\Big)^{\Pi}} \quad \text{(en utilisant la propriété } P_7\text{)} \\ &= \overline{\Big(\bigcup_{x \in \overline{B^{\Pi}}} \{x\}\Big)^{\Pi}} \\ &= \bigcap_{x \in \overline{B^{\Pi}}} \overline{\{x\}^{\Pi}} \quad \text{(en utilisant la loi de Morgan)} \end{aligned}$$

Par conséquent, par la propriété P_5 (i.e. $B \subseteq (B^{\Pi})^{\mathrm{N}}$), nous obtenons $B \subseteq \bigcap_{x \in \overline{B^{\Pi}}} \overline{\{x\}^{\Pi}}$.

Il s'ensuit que

$$\forall x \in \overline{B^{\Pi}} : B \subseteq \overline{\{x\}^{\Pi}} \qquad (a)$$

Voyons maintenant l'utilisation express de la condition nécessaire selon la proposition 7 :

$$\mathcal{K} \models \bigvee A \longmapsto \bigvee B \Longrightarrow \forall x \in \mathcal{O}, B \not\subseteq \overline{\{x\}^{\Pi}} \text{ ou } A \subseteq \overline{\{x\}^{\Pi}}$$

En particulier :

$$\mathcal{K} \models \bigvee A \longmapsto \bigvee B \Longrightarrow \forall x \in \overline{\{B\}^{\Pi}}, B \not\subseteq \overline{\{x\}^{\Pi}} \text{ ou } A \subseteq \overline{\{x\}^{\Pi}}$$

Selon le résultat précédent (a), il s'ensuit logiquement que :

$$\mathcal{K} \models \bigvee A \longmapsto \bigvee B \Longrightarrow \forall x \in \overline{\{B\}^{\Pi}}, A \subseteq \overline{\{x\}^{\Pi}}$$

Puisque pour chaque élément x de $\overline{\{B\}^{\Pi}}$, nous avons $A \subseteq \overline{\{x\}^{\Pi}}$, il est évident que $A \subseteq \bigcap_{x \in \overline{B^{\Pi}}} \overline{\{x\}^{\Pi}}$.

Selon le résultat précédent $(((B)^{\Pi})^{\mathrm{N}} = \bigcap_{x \in \overline{B^{\Pi}}} \overline{\{x\}^{\Pi}})$, enfin, nous obtenons

$$A \subseteq ((B)^{\Pi})^{\mathrm{N}}$$

2) Condition suffisante

Selon la proposition 7, nous devons prouver que : $A \subseteq (B)^{\Pi})^{\mathrm{N}} \Longrightarrow \forall x \in \mathcal{O} : B \not\subseteq \overline{\{x\}^{\Pi}} \;\; ou \;\; A \subseteq \overline{\{x\}^{\Pi}}$

Deux cas peuvent survenir $\forall x \in \mathcal{O}$:

a) $B \not\subseteq \overline{\{x\}^{\Pi}}$: dans ce cas, il ne reste rien à prouver.

b) $B \subseteq \overline{\{x\}^{\Pi}}$: en appliquant successivement les propriétés (P_3), (P_4) et (P_7), nous obtenons :

$$B \subseteq \overline{\{x\}^{\Pi}} \Longrightarrow B^{\Pi} \subseteq (\overline{\{x\}^{\Pi}})^{\Pi} \Longrightarrow (B^{\Pi})^{\mathrm{N}} \subseteq ((\overline{\{x\}^{\Pi}})^{\Pi})^{\mathrm{N}} \Longrightarrow (B^{\Pi})^{\mathrm{N}} \subseteq (((\overline{x})^{\mathrm{N}})^{\Pi})^{\mathrm{N}}$$

Nous avons $A \subseteq (B)^{\Pi})^{\mathrm{N}}$, nous obtenons : $A \subseteq (((\overline{x})^{\mathrm{N}})^{\Pi})^{\mathrm{N}}$.

En appliquant successivement les propriétés (P_4) et (P_7) sur la propriété (P_6)

$$(\{\overline{x}\}^{\mathrm{N}})^{\Pi} \subseteq \{\overline{x}\} \Longrightarrow ((\{\overline{x}\}^{\mathrm{N}})^{\Pi})^{\mathrm{N}} \subseteq \{\overline{x}\}^{\mathrm{N}} \Longrightarrow ((\{\overline{x}\}^{\mathrm{N}})^{\Pi})^{\mathrm{N}} \subseteq \overline{\{x\}^{\Pi}}$$

Ceci donne $A \subseteq \overline{\{x\}^{\Pi}}$.

□

Corollaire 1. *Si $\mathcal{K} \models \bigvee A \longmapsto \bigvee B$ alors $\bigvee A \longmapsto \bigvee B$ est valide dans $\overline{\mathfrak{B}}_{\mathrm{N}\Pi}(Int)$.*

Preuve du Corollaire 1: Selon le théorème 1, nous devons prouver que

$$A \subseteq ((B)^{\Pi})^{N} \Longrightarrow \bigvee A \longmapsto \bigvee B \;\; est \;\; valide \;\; dans \;\; \overline{\mathfrak{B}}_{\mathrm{N}\Pi}(Int)$$

Il est clair que, $\forall \overline{C} \in \overline{\mathfrak{B}}_{\mathrm{N}\Pi}(Int)$:

$$\begin{array}{lll}
 & B \not\subseteq C \;\; ou \;\; B \subseteq C & \\
\Longrightarrow & B \not\subseteq C \;\; ou \;\; (B)^{\Pi} \subseteq (C)^{\Pi} & \text{(en utilisant } P_3\text{)} \\
\Longrightarrow & B \not\subseteq C \;\; ou \;\; ((B)^{\Pi})^{N} \subseteq ((C)^{\Pi})^{N} & \text{(en utilisant } P_4\text{)} \\
\Longrightarrow & B \not\subseteq C \;\; ou \;\; A \subseteq ((C)^{\Pi})^{N} & \text{(puisque } A \subseteq ((B)^{\Pi})^{N}\text{)} \\
\Longrightarrow & B \not\subseteq C \;\; ou \;\; A \subseteq C & \text{(puisque } C \in \mathfrak{B}_{\mathrm{N}\Pi}(Int)\text{)} \\
\Longrightarrow & \overline{C} \;\; satisfait \;\; \bigvee A \longmapsto \bigvee B & \text{(par définition)}
\end{array}$$

Ainsi $\bigvee A \longmapsto \bigvee B$ est valide dans $\overline{\mathfrak{B}}_{\mathrm{N}\Pi}(Int)$ □

Dans ce qui suit, nous utilisant le contexte formel $\mathcal{K}^{\neg} = (\mathcal{O}, \mathcal{P}^{\neg}, \mathcal{R}^{\neg})$ qui est équivalent au contexte formel $\mathcal{K}(\mathcal{O}, \mathcal{P}, \mathcal{R})$ où les ensembles $\mathcal{P}^{\neg}$ et $\mathcal{R}^{\neg}$ sont définis comme suit :

$$\mathcal{P}^{\neg} = \{\neg a \mid a \in \mathcal{P}\}$$

$$\mathcal{R}^{\neg} = \{(x, \neg a) \mid (x, a) \notin \mathcal{R}\}$$

A titre d'exemple, la table 5.2 illustre le contexte formel $\mathcal{K}_S^{\neg}$ équivalent au contexte formel $\mathcal{K}_S$ (donné dans la table 2.2).

Les implications d'attributs conjonctives obtenues à partir d'un contexte formel $\mathcal{K}^{\neg}$ comportent donc des négations d'attributs. Aussi, par souci de clarté, si $A \subseteq \mathcal{P}$, on note $A^{\neg}$ l'ensemble $\{\neg a \mid a \in A\}$. Par exemple, si $A = \{a_1, a_4, a_5\}$, nous écrirons $A^{\neg} = \{\neg a_1, \neg a_4, \neg a_5\}$. Nous pouvons vérifier que $\{\neg a_1, \neg a_5\} \longmapsto \{\neg a_4\}$ depuis $\{\neg a_1, \neg a_5\}^{\Delta}_{\mathcal{K}_S^{\neg}} = \{\neg a_1\}^{\Delta}_{\mathcal{K}_S^{\neg}} \cap \{\neg a_5\}^{\Delta}_{\mathcal{K}_S^{\neg}} \subseteq \{\neg a_4\}^{\Delta}_{\mathcal{K}_S^{\neg}}$.

Table 5.2 – Contexte formel $\mathcal{K}_S^{\neg}$ équivalent à $\mathcal{K}_S$

$\mathcal{R}^{\neg}$	$\neg a_1$	$\neg a_2$	$\neg a_3$	$\neg a_4$	$\neg a_5$
x_1	×	×			
x_2		×	×	×	
x_3	×			×	×
x_4					×
x_5	×	×			
x_6	×	×		×	×

La proposition suivante établit un résultat utile.

Proposition 8. *Les quatre assertions suivantes sont équivalentes :*

- *L'implication d'attributs disjonctive $\bigvee A \longmapsto \bigvee B$ est valide dans le contexte formel $\mathcal{K}$.*
- *l'implication d'attributs (conjonctive) $B^{\neg} \longmapsto A^{\neg}$ est valide dans le contexte formel $\mathcal{K}^{\neg}$.*
- $A \subseteq (B^{\Pi})^{\mathrm{N}}$
- $A^{\Pi} \subseteq B^{\Pi}$

Preuve de la Proposition 8: Supposons que $B^{\neg} \longmapsto A^{\neg}$ ets valide dans $\mathcal{K}^{\neg}$. En termes logiques, cela signifie $\wedge_{b \in B} \neg b \rightarrow \wedge_{a \in A} \neg a$, qui est logiquement équivalent à $\vee_{a \in A} a \rightarrow \vee_{b \in B} b$. Maintenant,

$B^{\neg} \longmapsto A^{\neg}$ est valide dans $\mathcal{K}^{\neg}$ signifie $A \subseteq (B^{\Delta}_{\mathcal{K}^{\neg}})^{\Delta}_{\mathcal{K}^{\neg}}$, à savoir, $A \subseteq (\overline{B^{\Pi}})^{\Delta}_{\mathcal{K}^{\neg}}$ ssi $A \subseteq (B^{\Pi})^{\mathrm{N}}$. Pour le quatrième point : $A \subseteq ((B)^{\Pi})^{\mathrm{N}}$ ssi $(A)^{\Pi} \subseteq (((B)^{\Pi})^{\mathrm{N}}))^{\Pi}$ (en utilisant P_3) ssi $(A)^{\Pi} \subseteq (B)^{\Pi}$ (en utilisant P_8). □

Par exemple, $K_{\mathcal{S}} \models a_4 \longmapsto a_1 \vee a_5$ cela veut dire que $\{a_4\}^{\Pi} \subseteq \{a_1\}^{\Pi} \cup \{a_5\}^{\Pi}$. Il est intéressant de souligner que $(.)^{\Delta}$ et $(.)^{\Pi}$ jouent des rôles similaires pour extraire les règles conjonctives et disjonctives, respectivement. La condition pour que $A \longmapsto B$ soit une règle d'implication conjonctive pour un contexte formel est $A^{\Delta} \subseteq B^{\Delta}$, et c'est $A^{\Pi} \subseteq B^{\Pi}$ pour les règles disjonctives. Il peut également être montré directement comme suit en utilisant la propriété P_1 :

$$P_9 : B^{\Delta}_{\mathcal{K}} \subseteq A^{\Delta}_{\mathcal{K}} \Longleftrightarrow \overline{B^{\Pi}_{\overline{\overline{\mathcal{K}}}}} \subseteq \overline{A^{\Pi}_{\overline{\overline{\mathcal{K}}}}} \Longleftrightarrow \overline{B^{\Pi}_{\mathcal{K}}} \subseteq \overline{A^{\Pi}_{\mathcal{K}}} \Longleftrightarrow A^{\Pi}_{\mathcal{K}} \subseteq B^{\Pi}_{\mathcal{K}}$$

Notez que cette proposition repose sur une compréhension spécifique d'un contexte dans lequel un espace vide à la place (x, a) signifie que x *ne possède pas* l'attribut a. Faut-il l'entendre par ignorance, cette règle disjonctive ne serait pas significative pour le contexte.

Le corollaire suivant, suit directement la proposition8. Il établit un lien entre la satisfaction des implications d'attributs conjonctives et les implications d'attributs disjonctives.

Corollaire 2. *Soit $A, B, M \subseteq \mathcal{P}$. $M^{\neg} \models B^{\neg} \longmapsto A^{\neg}$ ssi $M \models \bigvee A \longmapsto \bigvee B$*

Preuve du Corollaire 2: $M^{\neg} \models B^{\neg} \longmapsto A^{\neg}$ est équivalent à : si $B^{\neg} \subseteq M^{\neg}$ alors $A^{\neg} \subseteq M^{\neg}$, qui est équivalent à si $B \subseteq M$ alors $A \subseteq M$, qui est précisément $M \models \bigvee A \longmapsto \bigvee B$. □

Les implications d'attributs disjonctives qui tiennent dans un contexte formel $\mathcal{K} = (\mathcal{O}, \mathcal{P}, \mathcal{R})$ peuvent être obtenues à partir du NΠ-treillis $\mathfrak{L}_{\mathrm{N}\Pi}$, comme le montre la proposition suivante.

Proposition 9. *Soit un contexte formel $\mathcal{K} = (\mathcal{O}, \mathcal{P}, \mathcal{R})$, $\mathcal{K} \models a \longmapsto \bigvee B$ ssi $(\{a\}^{\Pi}, (\{a\}^{\Pi})^{\mathrm{N}}) \leq (B^{\Pi}, (B^{\Pi})^{\mathrm{N}})$*

Preuve de la Proposition 9:

a) Condition nécessaire

$$\begin{array}{lll} & \mathcal{K} \models a \longmapsto \bigvee B & \\ \Rightarrow & \{a\}^{\Pi} \subseteq B^{\Pi} & \text{(en utilisant la proposition 8)} \\ \Rightarrow & (\{a\}^{\Pi})^{\mathrm{N}} \subseteq (B^{\Pi})^{\mathrm{N}} & \text{(en utilisant } P_4\text{)} \end{array}$$

Ce dernier est à son tour équivalent à $(\{a\}^{\Pi}, (\{a\}^{\Pi})^{\mathrm{N}}) \leq (B^{\Pi}, (B^{\Pi})^{\mathrm{N}})$ (i.e., $\mu(a) \leq \coprod_{b \in B} \mu(b)$) en utilisant la proposition 6.

b) Condition suffisante

$$\begin{array}{ll} & (\{a\}^{\Pi}, (\{a\}^{\Pi})^{\mathrm{N}}) \leq (B^{\Pi}, (B^{\Pi})^{\mathrm{N}}) \\ \Rightarrow & (\{a\}^{\Pi})^{\mathrm{N}} \subseteq (B^{\Pi})^{\mathrm{N}} \\ \Rightarrow & \{a\} \subseteq (\{a\}^{\Pi})^{\mathrm{N}} \subseteq (B^{\Pi})^{\mathrm{N}} \qquad \text{(en utilisant } P_5\text{)} \\ \Rightarrow & \{a\} \subseteq (B^{\Pi})^{\mathrm{N}} \end{array}$$

□

Cette proposition peut être utilisée pour extraire une implication d'attributs disjonctive de la forme $a \longmapsto \bigvee B$. En effet, il suffit de vérifier si la NΠ-paire associée à a est située au-dessus de la borne supérieure de toutes les NΠ-paires associées à b de B dans le NΠ-treillis $\mathfrak{L}_{\mathrm{N\Pi}}$. Par exemple, $a_5 \longmapsto \{a_1, a_4\}$ est une implication d'attributs disjonctive satisfaite par $\mathcal{K}_S$ de la table 2.2 puisque $(\{a_5\}^{\Pi}, (\{a_5\}^{\Pi})^{\mathrm{N}}) \leq (\{a_1, a_4\}^{\Pi}, (\{a_1, a_4\}^{\Pi})^{\mathrm{N}})$.

5.5 Base d'implications d'attributs disjonctives

L'ensemble des implications d'attributs disjonctives pour un contexte formel donné est souvent trop important. Ainsi, induire toutes ces implications devient une tâche fastidieuse. A cet effet, un plus petit ensemble d'implications (appelé base d'implications), à partir duquel toutes les autres implications du contexte formel suivent (c.à.d peuvent être dérivées) peut être d'intérêt à considérer.

Soit un contexte formel $\mathcal{K}(O, \mathcal{P}, \mathcal{R})$. Un ensemble d'implications est une base implicite pour $\mathcal{K}$ (noté $\mathcal{B}^{\mathcal{K}}$) si $\mathcal{B}^{\mathcal{K}}$ est sûr (sound) pour $\mathcal{K}$ (i.e. $\mathcal{K}$ satisfait toutes les implications de $\mathcal{B}^{\mathcal{K}}$) et, $\mathcal{B}^{\mathcal{K}}$ est complet (complete) pour $\mathcal{K}$ (i. e. chaque implication satisfaite par $\mathcal{K}$ suit (c.à.d peut être dérivée de) $\mathcal{B}^{\mathcal{K}}$). Ce qui suit définit la condition à partir de laquelle une implication d'attributs disjonctive suit un ensemble d'implications d'attributs disjonctives.

Définition 35. *Soit L un ensemble d'implications d'attributs disjonctives, une implication $\bigvee A \longmapsto \bigvee B$ suit L si :*

$$\forall M \in 2^{\mathcal{P}} : (M \textit{ satisfait } L) \quad \longrightarrow \quad (M \textit{ satisfait } \bigvee A \longmapsto \bigvee B)$$

Notons par $\delta(L)$ l'ensemble d'éléments de $2^{\mathcal{P}}$ t.q. $\delta(L) = \{M \subseteq \mathcal{P} \mid M \text{ satisfait } L\}$ et notons par $Imp(\mathcal{K})$ l'ensemble de toutes les implications d'attributs disjonctives du contexte formel $\mathcal{K}(O, \mathcal{P}, \mathcal{R})$. Le lemme suivant caractérise l'ensemble $Imp(\mathcal{K})$ comme l'ensemble de toutes les NΠ-paires.

Lemme 4. *Soit un contexte formel $\mathcal{K}(O, \mathcal{P}, \mathcal{R})$, nous avons :*

$$\delta(Imp(\mathcal{K})) = \overline{\mathfrak{B}}_{\mathrm{N\Pi}}(Int)$$

Preuve du Lemme 4: Afin de prouver ce résultat, nous devons établir :
a) $\overline{\mathfrak{B}}_{\mathrm{N\Pi}}(Int) \subseteq \delta(Imp(\mathcal{K}))$

Du corollaire 1, on peut remarquer que $\forall M \in \overline{\mathfrak{B}}_{\mathrm{N\Pi}}(Int)$: M satisfait toutes les implications d'attributs disjonctives de $Imp(\mathcal{K})$.
Puisque

$$\delta(Imp(\mathcal{K})) = \{M \mid M \text{ satisfait toutes les implications d'attributs disjonctives de } Imp(\mathcal{K})\}.$$

Il est évident que

$$\overline{\mathfrak{B}}_{\mathrm{N\Pi}}(Int) \subseteq \delta(Imp(\mathcal{K}))$$

b) $\delta(Imp(\mathcal{K})) \subseteq \overline{\mathfrak{B}}_{\mathrm{N\Pi}}(Int)$
Soit $\overline{M} \in \delta(Imp(\mathcal{K}))$. Il est clair que :

$$(M)^{\Pi})^{\mathrm{N}} \subseteq (M)^{\Pi})^{\mathrm{N}}$$

$$\begin{array}{lll}
\Longrightarrow & \mathcal{K} \models \bigvee((M)^{\Pi})^{\mathrm{N}} \longmapsto \bigvee M & \text{(par théorème 1)} \\
\Longrightarrow & \overline{M} \;\; satisfait \;\; \bigvee((M)^{\Pi})^{\mathrm{N}} \longmapsto \bigvee M & \text{(condition initiale)} \\
\Longrightarrow & M \nsubseteq M \text{ ou } (M)^{\Pi})^{\mathrm{N}} \subseteq M & \text{(par définition)} \\
\Longrightarrow & (M)^{\Pi})^{\mathrm{N}} \subseteq M &
\end{array}$$

Puisque $M \subseteq (M)^{\Pi})^{\mathrm{N}}$ (par propriété (P_5)). Nous avons enfin $M = (M)^{\Pi})^{\mathrm{N}}$ (i.e.$M \in \mathfrak{B}_{\mathrm{N\Pi}}(Int)$). Ainsi

$$\overline{M} \in \overline{\mathfrak{B}}_{\mathrm{N\Pi}}(Int).$$

□

Le lemme suivant indique que, pour prouver que l'ensemble L d'implications d'attributs disjonctives est complet, on doit montrer que chaque ensemble qui satisfait L est le complément d'un NΠ-intension.

Lemme 5. *Un ensemble L d'implications d'attributs disjonctives de $\mathcal{K}(\mathcal{O},\mathcal{P},\mathcal{R})$ est complet si et seulement si*

$$\delta(L) \subseteq \overline{\mathfrak{B}}_{\mathrm{N\Pi}}(Int)$$

Preuve du Lemme 5: L est un ensemble complet d'implications d'attributs disjonctives alors :

$$\begin{array}{ll}
 & \forall \bigvee A \longmapsto \bigvee B \in Imp(\mathcal{K}) : \bigvee A \longmapsto \bigvee B \text{ suit } L \text{ (par définition)} \\
\Longleftrightarrow & \forall \bigvee A \longmapsto \bigvee B \in Imp(\mathcal{K}), \forall M \in 2^{P} : (M \;\; satisfait \;\; L) \longrightarrow (M \;\; satisfait \;\; \bigvee A \longmapsto \bigvee B) \\
\Longleftrightarrow & \forall M \in 2^{P} : (M \;\; satisfait \;\; L) \longrightarrow (M \;\; satisfait \;\; Imp(\mathcal{K})) \\
\Longleftrightarrow & \delta(L) \subseteq \delta(Imp(\mathcal{K}))
\end{array}$$

Voyons maintenant par le lemme 4 que $\delta(Imp(\mathcal{K})) = \overline{\mathfrak{B}}_{\mathrm{N\Pi}}(Int)$. □

Pour un contexte formel donné $\mathcal{K}(\mathcal{O},\mathcal{P},\mathcal{R})$, $\psi^{\mathcal{K}}$ désigne l'ensemble d'implications d'attributs disjonctives défini comme suit :

$$\psi^{\mathcal{K}} = \{\bigvee((A)^{\Pi})^{\mathrm{N}} \longmapsto \bigvee A\}.$$

Théorème 2. *L'ensemble $\psi^{\mathcal{K}}$ est une base, i.e. il est sûr et complet.*

Preuve du Théorème 2:

a) $\psi^{\mathcal{K}}$ est sûr :

$\forall M \in 2^{\mathcal{P}} : \mathcal{K} \models \bigvee((\mathrm{M})^{\Pi})^{\mathrm{N}} \longmapsto \bigvee \mathrm{M}$ puisque la condition dans le théorème 1 est évidemment satisfaite (i.e. $((M)^{\Pi})^{\mathrm{N}} \subseteq ((M)^{\Pi})^{\mathrm{N}}$).

b) $\psi^{\mathcal{K}}$ est complet :

Soit $\overline{M} \in \delta(\psi^{\mathcal{K}})$. Il est clair que :

$$\begin{aligned} & \overline{M}\ \ satisfait\ \ \bigvee((\mathrm{M})^{\Pi})^{\mathrm{N}} \longmapsto \bigvee \mathrm{M} \quad \text{(dû à la définition de } \psi^{\mathcal{K}}) \\ \Longrightarrow\ & M \not\subseteq M \text{ ou } ((\mathrm{M})^{\Pi})^{\mathrm{N}} \subseteq \mathrm{M} \\ \Longrightarrow\ & ((\mathrm{M})^{\Pi})^{\mathrm{N}} \subseteq \mathrm{M} \end{aligned}$$

Puisque $M \subseteq ((\mathrm{M})^{\Pi})^{\mathrm{N}}$ (par propriété P_5). Nous obtenons $M = ((\mathrm{M})^{\Pi})^{\mathrm{N}}$ (i.e. $M \in \mathfrak{B}_{\mathrm{N}\Pi}(Int)$). Ainsi $\overline{M} \in \overline{\mathfrak{B}}_{\mathrm{N}\Pi}(Int)$. Nous avons enfin $\delta(\psi^{\mathcal{K}}) \subseteq \overline{\mathfrak{B}}_{\mathrm{N}\Pi}(Int)$. Ainsi $\psi^{\mathcal{K}}$ est complet (selon le lemme 5). □

5.6 Base minimale d'implications d'attributs disjonctives

Dans la section qui précède, nous avons vu qu'une base d'implication est un moyen plus facile de représenter toutes les implications. Cependant, il peut exister plusieurs bases d'implication. De toute évidence, la caractérisation de la plus petite (minimale) base d'implication est une question importante. Nous proposons dans cette section, de caractériser la base minimale d'implications d'attributs disjonctives. Pour cela, nous définissons d'abord la notion de base minimale.

Définition 36. *(Base minimale) Une base minimale d'implications d'attributs disjonctives est la plus petite base à partir de laquelle toutes les autres implications d'attributs disjonctives peuvent être générées.*

Définissons aussi la notion de "Pseudo-NΠ-Fermé" comme suit :

Définition 37. *(Pseudo-*NΠ*-Fermé). Soit $\mathcal{K}(\mathcal{O},\mathcal{P},\mathcal{R})$ un contexte formel. Un ensemble $A \subseteq \mathcal{P}$ est appelé Pseudo-*NΠ*-Fermé de $\mathcal{K}$ si :*

1) $A \neq ((A)^{\Pi})^{\mathrm{N}}$ et,

*2) pour chaque Pseudo-*NΠ*-Fermé $Q \subset A$: $((Q)^{\Pi})^{\mathrm{N}} \subseteq A$.*

Étant donné un contexte formel $\mathcal{K}(\mathcal{O},\mathcal{P},\mathcal{R})$, désignons maintenant par $Base_{\mathrm{N}\Pi}$, l'ensemble d'implications d'attributs disjonctives défini comme suit :

$$Base_{\mathrm{N}\Pi} = \{\bigvee((A)^{\Pi})^{\mathrm{N}} \longmapsto \bigvee A \mid A \text{ est un Pseudo-N}\Pi\text{-Fermé de } \mathcal{K}\}$$

Le théorème suivant établit que l'application de l'opérateur de fermeture $((.)^{\Pi})^{\mathrm{N}}$ à un Pseudo-NΠ-Fermé permet d'obtenir une base d'implications d'attributs disjonctives.

Théorème 3. *L'ensemble $Base_{\mathrm{N}\Pi}$ est une base, i.e. il est sûr et complet.*

Preuve du Théorème 3: Afin de prouver ce résultat, nous devons établir :
a) $Base_{\mathrm{N}\Pi}$ est sûr :
Il est clair que $Base_{\mathrm{N}\Pi} \subseteq \psi^{\mathcal{K}}$ et $\psi^{\mathcal{K}}$ est sûr, ainsi que b) $Base_{\mathrm{N}\Pi}$ est complet :
Afin de montrer que $Base_{\mathrm{N}\Pi}$ est complet, nous devons montrer que chaque $\overline{M} \in \delta(Base_{\mathrm{N}\Pi})$ est un élélment de $\overline{\mathfrak{B}}_{\mathrm{N}\Pi}(Int)$ (i.e. $M \in \mathfrak{B}_{\mathrm{N}\Pi}(Int)$) (par lemme 5). Soit Q un Pseudo-NΠ-Fermé. Soit $\overline{M} \in \delta(Base_{\mathrm{N}\Pi})$ alors :

$$\begin{array}{rl} & \overline{M} \text{ satisfait } \bigvee((Q)^{\Pi})^{N} \longmapsto \bigvee Q \\ \Longrightarrow & Q \subseteq M \longrightarrow ((Q)^{\Pi})^{N} \subseteq M \\ \Longrightarrow & Q \subset M \longrightarrow ((Q)^{\Pi})^{N} \subseteq M \end{array}$$

Si nous supposons que $M \neq ((M)^{\Pi})^{N}$, M lui-même satisfait la définition d'un Pseudo-NΠ-Fermé. Par conséquent, $\bigvee((M)^{\Pi})^{N} \longmapsto \bigvee M \in Base_{\mathrm{N}\Pi}$ mais M ne satisfait pas $\bigvee((M)^{\Pi})^{N} \longmapsto \bigvee M$, ce qui donne une contradiction. Donc, $M = ((M)^{\Pi})^{N}$ alors $\overline{M} \in \overline{\mathfrak{B}}_{\mathrm{N}\Pi}(Int)$ (i.e. $M \in \mathfrak{B}_{\mathrm{N}\Pi}(Int)$). □

Dans le reste de cette section, on va montrer l'utilité de la base $Base_{\mathrm{N}\Pi}$. En effet, il sera prouvé par le théorème 4 que $Base_{\mathrm{N}\Pi}$ est une base minimale. Les deux lemmes suivants sont donnés pour obtenir cet important résultat.

Lemme 6. *Soit P une NΠ-intension et Q un Pseudo-NΠ-Fermé, avec $P \not\subseteq Q$ et $Q \not\subseteq P$, alors $\overline{P \cap Q} \in \overline{\mathfrak{B}}_{\mathrm{N}\Pi}(Int)$.*

Preuve du Lemme 6: Soit $(\bigvee((A)^{\Pi})^{\mathrm{N}} \longmapsto \bigvee A) \in Base_{\mathrm{N}\Pi}$.
D'une part, puisque P est une NΠ-intension (i.e. $P \in \mathfrak{B}_{\mathrm{N}\Pi}(Int)$), selon le lemme 4, nous avons :

$$\begin{array}{rl} & \overline{P} \text{ satisfait } \bigvee((A)^{\Pi})^{\mathrm{N}} \longmapsto \bigvee A \\ \Longleftrightarrow & A \subseteq P \longrightarrow (A)^{\Pi})^{\mathrm{N}} \subseteq P \quad \text{(a)} \end{array}$$

D'autre part, puisque Q est un Pseudo-NΠ-Fermé, nous avons :

$$\begin{array}{rl} & Q \neq (Q)^{\Pi})^{\mathrm{N}} \text{ et} \\ & A \subset Q \text{ alors } ((A)^{\Pi})^{\mathrm{N}} \subseteq Q \quad \text{(b)} \\ \Longrightarrow & \overline{Q} \text{ satisfait } \bigvee((A)^{\Pi})^{\mathrm{N}} \longmapsto \bigvee A \text{ (à l'exception de } \overline{A} \text{ lui-même)} \end{array}$$

De (a), (b) et la condition initiale (i.e. $P \not\subseteq Q$ et $Q \not\subseteq P$), on a :

$$\begin{array}{rl} & A \subset P \cap Q \text{ alors } ((A)^{\Pi})^{\mathrm{N}} \subseteq P \cap Q \\ \Longleftrightarrow & \overline{P \cap Q} \text{ satisfait } Base_{\mathrm{N}\Pi} \backslash \{\bigvee((P)^{\Pi})^{\mathrm{N}} \longmapsto \bigvee P, \bigvee((Q)^{\Pi})^{\mathrm{N}} \longmapsto \bigvee Q\} \end{array}$$

Depuis $P \nsubseteq Q \Longrightarrow P \nsubseteq P \cap Q \Longrightarrow \overline{P \cap Q}$ satisfait $((P)^{\Pi})^{\mathrm{N}} \longmapsto \bigvee P$ on raison de la symétrie, nous obtenons $\overline{P \cap Q}$ satisfait $((Q)^{\Pi})^{\mathrm{N}} \longmapsto \bigvee Q$. Ainsi, $\overline{P \cap Q}$ satisfait $Base_{\mathrm{N}\Pi}$. Rappelons que l'ensemble $Base_{\mathrm{N}\Pi}$ est complet, donc, $\delta(Base_{\mathrm{N}\Pi}) \subseteq \overline{\mathfrak{B}}_{\mathrm{N}\Pi}(Int)$. Cela montre que $\overline{P \cap Q} \in \overline{\mathfrak{B}}_{\mathrm{N}\Pi}(Int)$ (par lemme 5) □

Le lemme suivant montre, entre autres, qu'il ne peut y avoir aucun ensemble complet qui présente moins d'implication d'attributs disjonctives qu'il y a de Pseudo-NΠ-Fermé.

Lemme 7. *Pour chaque Pseudo-*NΠ*-Fermé P, chaque base $\mathcal{B}^{\mathcal{K}}$ d'implications d'attributs disjonctives contient une implication $\bigvee A \longmapsto \bigvee B$ t.q.* $((B)^{\Pi})^{\mathrm{N}} = ((P)^{\Pi})^{\mathrm{N}}$.

Preuve du Lemme 7: soit P un Pseudo-NΠ-Fermé alors :

$$
\begin{array}{lll}
 & P \neq ((P)^{\Pi})^{\mathrm{N}} & \\
\Longrightarrow & P \notin \mathfrak{B}_{\mathrm{N}\Pi}(Int) & \\
\Longrightarrow & \overline{P} \notin \overline{\mathfrak{B}}_{\mathrm{N}\Pi}(Int) & \\
\Longrightarrow & \overline{P} \text{ ne satisfait pas } \mathcal{B}^{\mathcal{K}} & \text{(par lemme 5)} \\
\Longrightarrow & \exists \bigvee A \longmapsto \bigvee B \in \mathcal{B}^{\mathcal{K}} : \overline{P} \text{ ne satisfait pas } \bigvee A \longmapsto \bigvee B & \\
\Longrightarrow & A \nsubseteq P \text{ et} \quad \text{(a)} & \\
 & B \subseteq P \quad \text{(b)} &
\end{array}
$$

Par le théorème 1 nous avons : $A \subseteq ((B)^{\Pi})^{\mathrm{N}}$ (c)
De (a) et (c) nous obtenons : $((B)^{\Pi})^{\mathrm{N}} \nsubseteq P$ (d)
Si $\overline{((B)^{\Pi})^{\mathrm{N}} \cap P} \subseteq \overline{\mathfrak{B}}_{\mathrm{N}\Pi}(Int)$ alors :

$$
\begin{array}{ll}
 & \overline{((B)^{\Pi})^{\mathrm{N}} \cap P} \text{ satisfait } \mathcal{B}^{\mathcal{K}} \\
\Longrightarrow & \overline{((B)^{\Pi})^{\mathrm{N}} \cap P} \text{ satisfait } \bigvee A \longmapsto \bigvee B \\
\Longrightarrow & A \subseteq (((B)^{\Pi})^{\mathrm{N}} \cap P) \text{ ou } B \nsubseteq (((B)^{\Pi})^{\mathrm{N}} \cap P)
\end{array}
$$

Cette déclaration est fausse (par (a) et (b)). Par conséquent $\overline{((B)^{\Pi})^{\mathrm{N}} \cap P} \nsubseteq \overline{\mathfrak{B}}_{\mathrm{N}\Pi}(Int)$ alors :

$$
\begin{array}{lll}
 & ((B)^{\Pi})^{\mathrm{N}} \subseteq P \text{ ou } P \subseteq ((B)^{\Pi})^{\mathrm{N}} & \text{(en utilisant lemme 6)} \\
\Longrightarrow & P \subseteq ((B)^{\Pi})^{\mathrm{N}} & \text{(par (d))} \\
\Longrightarrow & ((P)^{\Pi})^{\mathrm{N}} \subseteq ((((B)^{\Pi})^{\mathrm{N}})^{\Pi})^{\Pi} & \text{(en utilisant } (P_3) \text{ et } (P_4)) \\
\Longrightarrow & ((P)^{\Pi})^{\mathrm{N}} \subseteq ((B)^{\Pi})^{\mathrm{N}} &
\end{array}
$$

De (b) : $B \subseteq P$, en appliquant successivement les propriétés (P_3) et (P_4), nous obtenon : $((B)^{\Pi})^{\mathrm{N}} \subseteq ((P)^{\Pi})^{\mathrm{N}}$. Enfin $((B)^{\Pi})^{\mathrm{N}} = ((P)^{\Pi})^{\mathrm{N}}$. □

Théorème 4. *$Base_{\mathrm{N}\Pi}$ est une base minimale de $\mathcal{K}$.*

Preuve du Théorème 4: Soit$\mathcal{B}^{\mathcal{K}}$ une base d'implications d'attributs disjonctives. Notons qu'il suffit de montrer que

$$|Base_{\mathrm{N}\Pi}| \leq |\mathcal{B}^{\mathcal{K}}|$$

Il est clair qu'il existe une bijection entre la base minimale $Base_{\mathrm{N\Pi}}$ et l'ensemble de tous les Pseudo-NΠ-Fermé. En outre, chaque base $\mathcal{B}^{\mathcal{K}}$ d'implications d'attributs disjonctives contient obligatoire une implication $\bigvee A \longmapsto \bigvee B$ avec $((B)^{\Pi})^{\mathrm{N}} = ((P)^{\Pi})^{\mathrm{N}}$ pour chaque Pseudo-NΠ-Fermé P par lemme 7. D'où, $Base_{\mathrm{N\Pi}}$ est une base minimale d'implications d'attributs disjonctives. □

Corollaire 3. *L'ensemble $Base_{\mathrm{N\Pi}}$ est une base non redondante.*

Pour plus de commodité, les implications d'attributs disjonctives seront exprimées, dans le reste, de la forme $\bigvee(((A)^{\Pi})^{\mathrm{N}} \setminus A) \to \bigvee A$ plutôt que $\bigvee((A)^{\Pi})^{\mathrm{N}} \to \bigvee A$.

Exemple 11. *La base minimale d'implications d'attributs disjonctives du contexte formel $\mathcal{K}_S$ de la table 2.2 est :*

$Base_{\mathrm{N\Pi}}^{\mathcal{K}_S} = \{$
$a_2 \vee a_4 \longmapsto a_3,$
$a_5 \longmapsto a_1 \vee a_4,$
$a_4 \longmapsto a_1 \vee a_5,$
$a_1 \longmapsto a_4 \vee a_5,$
$a_1 \vee a_4 \longmapsto a_2 \vee a_5\ \}$

5.7 Conclusion

Nous avons présenté dans ce chapitre notre première contribution qui consiste en la caractérisation d'une base minimale d'implications d'attributs disjonctives.

Il s'avère que l'ACF est souvent amenée à considérer des réalités concrètes (mesures, observations, jugements, etc...) où peuvent apparaître des notions d'imprécision, d'incertitude et d'incomplétude. De pareilles réalités résultent des contextes formels de diverses natures. Il semble donc utile de voir ce que deviennent les implications résultantes pour chaque type de contexte formel. Dans le chapitre suivant, nous traitons les contextes formels incomplets.

Chapitre 6
Implications d'attributs dans un contexte incomplet

Sommaire

6.1 Introduction

Rappelons que la théorie de l'analyse de concepts formels a été développée par Wille [66], dans un cadre où la relation considérée est exclusivement Booléenne (i.e. un objet possède totalement l'attribut ou ne le possède pas du tout). De plus, cette relation est complètement renseignée, c'est à dire qu'il est toujours connu si un objet possède ou ne possède pas la propriété correspondante.

Il s'avère que ce cadre est souvent trop restrictif pour représenter certaines réalités. En effet, L'ACF est amenée à considérer des contextes de diverses natures, modélisant des réalités concrètes (exemple : mesures, observations, jugements, etc...) où peuvent apparaître des questions de gradualité, d'incertitude, d'imprécision, d'incomplétude, etc... La table 6.1 illustre différentes extensions d'un contexte formel qui peuvent ainsi s'avérer naturelles.

TABLE 6.1 – Contexte formel : Généralisations

	Jeune	*Anglais*	*Marié*
Ali	1	0.9	×
Safia	0.7	]0,1]	?
Nassime	0.6	0	
Nahla	1	[0.2,0.4]	(0.7 ; 0.0)

A titre d'exemple, la première colonne de la table 6.1 indique le degré de satisfaction graduelle de la propriété *jeune*. La seconde colonne indique le niveau de maîtrise de l'*Anglais*. Ce niveau peut être connu de manière précise ou seulement apprécié sous forme d'intervalles. Par exemple, on sait seulement qu'il est faux que *Safia* ne parle pas du tout *Anglais*. La plupart des approches actuelles traitent essentiellement ce type de contexte. Depuis le premier papier de Burusco et Fuentes-Gonzalez [26], différentes variantes reposant sur une algèbre résiduée [18] ont été proposées (voir [20] pour une vue d'ensemble). Dans [44], une approche utilisant une algèbre plus générale est proposé pour des contextes en présence d'intervalles.

La troisième colonne de la table 6.1 illustre la présence d'incomplétude : alors que *Ali* est marié et *Nassime* ne l'est pas, rien ne peut être affirmé pour *Safia*. Tandis que la paire (0.7 ;0.0) exprime que la certitude que *Nahla* ne soit pas mariée est de 0.0, et de 0.7 qu'elle le soit. Remarquons que l'entrée " ?" corresponds à (0 ;0). Dans cette optique, Burmeister et Holzer [25] ont considéré des contextes formels Booléens comportant une troisième valeur qui permet de modéliser l'ignorance totale sur la satisfaction de la propriété par l'objet.

Comme on l'a précisé, L'ACF est amenée à considérer des contextes de diverses natures, il semble donc utile de voir ce que deviennent les implications résultant de chaque contexte formel. Dans ce chapitre, nous allons traiter le cas des contextes formels incomplets (c'est à dire correspondant à la troisième colonne). De manière originale, nous allons définir et caractériser de nouvelles formes d'implications d'attributs, à savoir les implications possibles et certaines que ce soit dans le cas conjonctif ou dans le cas disjonctif. Le cadre théorique retenu pour modéliser l'incomplétude est celui de la théorie des possibilités.

6.2 Contexte incomplet

Il est largement admis que, dans de nombreux domaines, les connaissances peuvent être incomplètes. Il semble donc important de distinguer entre le cas où l'on sait qu'un objet ne possède pas un attribut et le cas où on ne sait pas si l'objet possède l'attribut ou non. C'est une distinction que les contextes formels habituels ne peuvent pas faire. Il semble donc utile de voir ce que deviennent les implications résultant d'un contexte formel incomplet. Le cas de l'ignorance partielle dans des contextes a été considéré par

Obiedkov [94] en utilisant la logique modale et par Burmeister et Holzer [25] en utilisant la logique à trois valeurs (Kleene three–valued logic). Ils ont proposé d'introduire une troisième valeur, notée " ?", dans un contexte formel, ce qui conduit à la notion de contexte incomplet, parfois aussi appelé contexte formel à trois valeurs. Dans ce qui suit, nous utilisons la théorie des possibilités pour modéliser des informations incomplètes.

Un contexte incomplet est formellement représenté par $\mathbf{K}^? = (\mathcal{O}, \mathcal{P}, \{+, -, ?\}, \mathbf{R})$ où $\mathcal{O}$ est un ensemble d'objets, $\mathcal{P}$ un ensemble d'attributs, "+", "-", " ?" sont les trois entrées possibles du contexte incomplet, et $\mathbf{R}$ est une relation ternaire $\mathbf{R} \subseteq \mathcal{O} \times \mathcal{P} \times \{+, -, ?\}$. L'interprétation de la relation $\mathbf{R}$ est comme suit. Soit $x \in \mathcal{O}$ et $a \in \mathcal{P}$:

- $(x, a, +) \in \mathbf{R}$: il est connu que l'objet x vérifie l'attribut a ;
- $(x, a, -) \in \mathbf{R}$: il est connu que l'objet x ne vérifie pas l'attribut a ;
- $(x, a, ?) \in \mathbf{R}$: on ne sait pas, si l'objet x vérifie ou ne vérifie pas l'attribut a.

Exemple 12. *soit* $\mathbf{K}^?$ *un contexte formel incomplet.*

$\mathbf{K}^?$	a_1	a_2	a_3
x_1	$+$	$+$	$+$
x_2	$-$	$+$	$?$
x_3	$+$	$?$	$-$

TABLE 6.2 – Contexte formel incomplet $\mathbf{K}^?$.

Un contexte formel incomplet peut être considéré comme la famille de tous les contextes formels standards obtenus en remplaçant les entrées inconnues $(x, a, ?)$ par celles connues $((x, a, +)$ ou $(x, a, -))$. Considérant les deux cas extrêmes où toutes les entrées inconnues $(x, a, ?)$ sont remplacées par $(x, a, -)$, et le cas où toutes ces entrées inconnues $(x, a, ?)$ sont remplacées par $(x, a, +)$, cela donne naissance aux complétions inférieures et supérieures respectivement [38, 52].

De cette manière, deux contextes formels classiques (Booléen), notés $\mathcal{K}_*$ et $\mathcal{K}^*$ peuvent être obtenus respectivement suite aux deux remplacements. De manière plus formelle :

- $\mathcal{K}_* = (\mathcal{O}, \mathcal{P}, \mathcal{R}_*)$ est un contexte formel booléen tel que $\mathcal{R}_* = \{(x, a) \mid (x, a, +) \in \mathbf{R}\}$ où les entrées " ?" sont remplacées par $-$, interprétant le manque de connaissance sur (x, a) comme le fait que x ne possède pas l'attribut a.
- $\mathcal{K}^* = (\mathcal{O}, \mathcal{P}, \mathcal{R}^*)$ est un contexte formel booléen tel que $\mathcal{R}^* = \{(x, a) \mid (x, a, +) \in \mathbf{R}$ ou $(x, a, ?) \in \mathbf{R}\}$ où les entrées " ?" sont remplacées par $+$, interprétant le manque de connaissance sur (x, a) comme le fait que x possède l'attribut a.

Il existe exactement 2^n contextes formels obtenus en remplaçant chaque " ?" par "+" ou "-" (où n est le nombre de " ?" dans le contexte formel incomplet). Ils sont appelés *contextes possibles*. Si $\mathcal{K}_1$ et $\mathcal{K}_2$ sont des contextes possibles obtenus à partir de $\mathbf{K}^?$. On définit un ordre entre eux comme suit : $\mathcal{K}_1 \prec \mathcal{K}_2$ si $\mathcal{R}_1 \subset \mathcal{R}_2$, i.e., Il y a plus de "+" dans $\mathcal{R}_2$ que dans $\mathcal{R}_1$. Ensuite, il est facile de vérifier que :

Lemme 8. *$\mathcal{K}_1 \prec \mathcal{K}_2$ implique que pour tout sous-ensemble A d'attributs, $A^{\Delta}_{\mathcal{K}_1} \subseteq A^{\Delta}_{\mathcal{K}_2}$*

Preuve du Lemme 8: Il est évident que puisque tous les "+" dans $\mathcal{R}_1$ sont des "+" dans $\mathcal{R}_2$ il ne peut pas y avoir moins d'objets satisfaisant tous les attributs dans A pour $\mathcal{K}_2$ que pour $\mathcal{K}_1$. □

Il est clair que le contexte minimal pour $\prec$ est $\mathcal{K}_*$ et que le contexte maximal est $\mathcal{K}^*$.

Exemple 12 (suite) Les contextes formels de table 6.3 illustrent tous les contextes formels possibles obtenus à partir du contexte formel incomplet $\mathbf{K}^?$ donné dans la table 6.7.

$\mathcal{K}_*$	a_1	a_2	a_3	$\mathcal{K}^*$	a_1	a_2	a_3	$\mathcal{K}_1$	a_1	a_2	a_3	$\mathcal{K}_2$	a_1	a_2	a_3
x_1	+	+	+	x_1	+	+	+	x_1	+	+	+	x_1	+	+	+
x_2	−	+	−	x_2	−	+	+	x_2	−	+	−	x_2	−	+	+
x_3	+	−	−	x_3	+	+	−	x_3	+	+	−	x_3	+	−	−

TABLE 6.3 – Tous les contextes formels possibles de $\mathbf{K}^?$.

Dans le cas d'un contexte formel incomplet $\mathbf{K}^?$, l'implication d'attributs $a_3 \longmapsto a_2$ est valide dans le contexte formel $\mathcal{K}_*$, $\mathcal{K}^*$, $\mathcal{K}_1$ et $\mathcal{K}_2$; cette implication d'attributs est valide indépendamment de ce que peuvent représenter les points d'interrogation. Considérant l'implication d'attributs $a_1 \longmapsto a_2$, elle n'est valable que dans $\mathcal{K}^*$ et $\mathcal{K}_1$: elle dépend de la valeur avec laquelle les points d'interrogation seront remplacés.

Notez que, dans la terminologie de la théorie des possibilités, $A^{\Delta}_{\mathcal{K}_*} = \{x \mid A \subseteq \mathcal{R}_*(x)\}$ est l'ensemble des objets possédant certainement tous les attributs de A tandis que $A^{\Delta}_{\mathcal{K}^*} = \{x \mid A \subseteq \mathcal{R}^*(x)\}$ est l'ensemble des objets pouvant posséder tous les attributs de A.

6.3 Implications d'attributs (conjonctives) possibles et certaines dans un contexte incomplet

Les implications d'attributs (conjonctives) obtenues à partir du contexte formel incomplet sont soit des implications d'attributs certaines, soit des implications d'attributs possibles.

Définition 38. *Une implication d'attributs est dite* certaine *dans un contexte incomplet $\mathbf{K}^?$ si et seulement si elle est valide dans* chaque *contexte formel $\mathcal{K}$ tel que $\mathcal{K}_* \preceq \mathcal{K} \preceq \mathcal{K}*$.*

Cette définition peut sembler difficile à vérifier en utilisant une énumération explicite des contextes formels possibles. Le théorème suivant fournit une solution à ce problème.

Théorème 5. *$A \longmapsto B$ est une implication d'attributs certaine dans $\mathbf{K}^?$ ssi $A^{\Delta}_{\mathcal{K}^*} \subseteq B^{\Delta}_{\mathcal{K}_*}$*

Preuve du Théorème 5:

a) *Condition nécessaire*

Soit $A \longmapsto B$ une implication d'attributs certaine dans $\mathbf{K}^?$ et on suppose que $A^{\Delta}_{\mathbf{K}^*} \nsubseteq B^{\Delta}_{\mathbf{K}_*}$.
$A^{\Delta}_{\mathbf{K}^*} \nsubseteq B^{\Delta}_{\mathbf{K}_*} \Longrightarrow \exists x \in \mathcal{O} \mid x \in A^{\Delta}_{\mathbf{K}^*}$ et $x \notin B^{\Delta}_{\mathbf{K}_*} \Longrightarrow \exists$ un contexte formel $\mathcal{K}_j(\mathcal{O}, \mathcal{P}, \mathcal{R}_j)$ tel que $\mathcal{R}_j = \mathcal{R}_* \cup Q$ où $Q = \{(x, a) \in \mathcal{R}^* \mid a \in A \wedge (x, a, ?) \in \mathbf{R}\}$
ainsi, $x \in A^{\Delta}_{\mathcal{K}_j}$ et $x \notin B^{\Delta}_{\mathcal{K}_j} \Longrightarrow A^{\Delta}_{\mathcal{K}_j} \nsubseteq B^{\Delta}_{\mathcal{K}_j} \Longrightarrow A \longmapsto B$ n'est pas valide dans $\mathcal{K}_j \Longrightarrow A \longmapsto B$ n'est pas une implication d'attributs certaine.

b) *Condition suffisante*

Si $A^{\Delta}_{\mathcal{K}^*} \subseteq B^{\Delta}_{\mathcal{K}_*}$ alors $A \longmapsto B$ est une implication d'attributs certaine dans $\mathbf{K}^?$. En effet, pour tout contexte formel possible $\mathcal{K}_j$ il est valide que $A^{\Delta}_{\mathcal{K}_j} \subseteq A^{\Delta}_{\mathcal{K}^*} \subseteq B^{\Delta}_{\mathcal{K}_*} \subseteq B^{\Delta}_{\mathcal{K}_j}$ due au lemme 8. Par conséquent $A \longmapsto B$ est une implication d'attributs certaine dans $\mathcal{K}_j$ Pour tous les contextes formels possibles $\mathbf{K}^?$. □

Remarque 1. *Une preuve directe de la condition nécessaire peut également être donnée. Notez que $A \longmapsto B$ est une implication d'attributs certaine si et seulement si $A \longmapsto b$ est une implication d'attributs certaine pour tout $b \in B \setminus A$. Dans ce qui suit nous pouvons se limiter aux implications d'attributs certaines $A \longmapsto b$. Cela signifie que pour tout contexte $\mathcal{K}$ compatible avec $\mathbf{K}^?, A^{\Delta}_{\mathcal{K}} \subseteq \{b\}^{\Delta}_{\mathcal{K}}$ est valide. Considérons l'ensemble des objets tels que $O_A = \{x \in \mathcal{O} : \forall a \in A, (x, a, -) \notin \mathbf{R}\}$. Ainsi, les lignes restreintes à A du contexte incomplet correspondant à O_A ne contiennent que "+" ou " ?". L'inclusion $A^{\Delta}_{\mathcal{K}} \subseteq \{b\}^{\Delta}_{\mathcal{K}}, \forall \mathcal{K}$ compatible avec $\mathbf{K}^?$ signifie que si $x \in O_A$ alors $(x, b, +) \in \mathbf{R}$. donc nous pouvons voir que $A^{\Delta}_{\mathcal{K}^*} = O_A$, tandis que $\{b\}^{\Delta}_{\mathcal{K}_*} \supseteq O_A$.*

Un autre problème est de déterminer toutes les implications d'attributs possibles :

Définition 39. *Une implication d'attributs est dite* possible *dans un contexte incomplet $\mathbf{K}^?$ ssi elle est valide* dans au moins *un contexte formel $\mathcal{K}$ tel que $\mathcal{K}_* \preceq \mathcal{K} \preceq \mathcal{K}*$.*

Il est clair qu'une implication d'attributs certaine est également possible dans le sens de cette définition. Évidemment, l'inverse ne tient pas. Encore une fois, vérifier si une implication d'attributs est possible semble exiger une énumération des contextes possibles. Le théorème suivant facilite cette détermination.

Théorème 6. *$A \longmapsto B$ est une implication d'attributs possible dans $\mathbf{K}^?$ ssi $A^{\Delta}_{\mathcal{K}_*} \subseteq B^{\Delta}_{\mathcal{K}^*}$*

Preuve du Théorème 6:

a) *Condition nécessaire :*

$A \longmapsto B$ est une implication d'attributs possible dans $\mathbf{K}^?$ signifie qu'il existe un contexte formel

$\mathcal{K}_j$ tel que $\mathcal{K}_* \leq \mathcal{K}_j \leq \mathcal{K}^*$ et $A^{\Delta}_{\mathbf{K}_j} \subseteq B^{\Delta}_{\mathbf{K}^j}$. (a)
Il est clair que $A^{\Delta}_{\mathcal{K}_*} \subseteq A^{\Delta}_{\mathbf{K}_j}$ et $B^{\Delta}_{\mathbf{K}_j} \subseteq B^{\Delta}_{\mathcal{K}^*}$ par le lemme 8.
Mettre ces inclusions ensemble : $A^{\Delta}_{\mathcal{K}_*} \subseteq A^{\Delta}_{\mathbf{K}_j} \subseteq B^{\Delta}_{\mathbf{K}_j} \subseteq B^{\Delta}_{\mathcal{K}^*}$, enfin $A^{\Delta}_{\mathcal{K}_*} \subseteq B^{\Delta}_{\mathcal{K}^*}$.

b)*Condition suffisante*
on suppose que $A^{\Delta}_{\mathcal{K}_*} \subseteq B^{\Delta}_{\mathcal{K}^*}$. Deux cas sont facilement donnés avec :
$B \subseteq A$: alors $A \longmapsto B$ est clairement une implication d'attributs valide dans tout les contextes possibles ;
$B \cap A \neq \emptyset$ et $B \nsubseteq A$: alors $A \longmapsto B$ ets une implication valide ssi $A \longmapsto B \setminus A$ est une implication d'attributs valide dans le même contexte.
Donc nous n'avons qu'à étudier le cas où A et B sont disjoints. Définir le contexte formel $\mathcal{K}_{AB}$ comme suit : Remplacer tout (?) Dans les colonnes de A par "-", et tous (?) dans les colonnes de B par "+". Alors, il est clair que $A^{\Delta}_{\mathbf{K}_*} = A^{\Delta}_{\mathbf{K}_{AB}}$ et $B^{\Delta}_{\mathbf{K}^*} = B^{\Delta}_{\mathbf{K}_{AB}}$. Par conséquent $A^{\Delta}_{\mathbf{K}_{AB}} \subseteq B^{\Delta}_{\mathbf{K}_{AB}}$, donc $A \longmapsto B$ une implication d'attributs valide dans $\mathcal{K}_{AB}$, et puisque $\mathcal{K}_* \leq \mathcal{K}_{AB} \leq \mathcal{K}^*$, $A \longmapsto B$ est une implication d'attributs possible dans $\mathbf{K}^?$. □

La section suivante considère également les implications d'attributs disjonctives, présentées dans le chapitre 5, pour des contextes formels incomplets.

6.4 Implications d'attributs disjonctives possibles et certaines dans un contexte incomplet

Comme dans le cas des implications d'attributs conjonctives, nous distinguons les implications d'attributs disjonctives certaines et les implications d'attributs disjonctives possibles.

Notons que $(A)^{\Pi}_{\mathcal{K}_*}$ est l'ensemble des objets ayant certainement au moins un attribut dans A et $(A)^{\Pi}_{\mathcal{K}^*}$ est l'ensemble des objets pouvant avoir au moins un attribut dans A. Alors, $\overline{(A)^{\Pi}_{\mathcal{K}_*}}$ est l'ensemble des objets qui n'ont certainement jamais d'attribut dans A et $\overline{(A)^{\Pi}_{\mathcal{K}^*}}$ est l'ensemble des objets qui ont tous leurs attributs à l'exception de A.

On note par $\overline{\mathbf{K}^?} = (\mathcal{O}, \mathcal{P}^{\neg}, \{+, -, ?\}, \overline{\mathbf{R}})$ le complémentaire du contexte formel incomplet $\mathbf{K}^?$. Il est défini en utilisant la négation de Kleene, à savoir $(x, a, +) \in \overline{\mathbf{R}}$ ssi $(x, a, -) \in \mathbf{R}$, $(x, a, -) \in \overline{\mathbf{R}}$ ssi $(x, a, +) \in \mathbf{R}$, et $(x, a, ?) \in \overline{\mathbf{R}}$ ssi $(x, a, ?) \in \mathbf{R}$. Il est facile de voir que le lower completion de $\overline{\mathbf{K}^?}$ est $\overline{\mathcal{K}^*}$, le complémentaire de la complétion haute (upper completion) de $\mathbf{K}^?$, et la complétion haute de $\overline{\mathbf{K}^?}$ et $\overline{\mathcal{K}_*}$, le complémentaire de la complétion basse (lower completion) de $\mathbf{K}^?$. Nous obtenons deux autres résultats :

Théorème 7. *$\bigvee A \longmapsto \bigvee B$ est une implication d'attributs disjonctive certaine ssi $A^{\Pi}_{\mathcal{K}^*} \subseteq B^{\Pi}_{\mathcal{K}_*}$*

Preuve du Théorème 7: D'une part, par le théorème 5, $B^{\neg} \longmapsto A^{\neg}$ est une implication d'attributs certaine dans $\overline{\mathbf{K}^?}$ ssi $B^{\Delta}_{\overline{\mathcal{K}_*}} \subseteq A^{\Delta}_{\overline{\mathcal{K}^*}}$, puisque $\overline{\mathcal{K}_*}$ et $\overline{\mathcal{K}^*}$ sont respectivement la haute complétion et la basse complétion (the upper and the lower completions) dans $\overline{\mathbf{K}^?}$; cela signifie également que $\bigvee A \longmapsto \bigvee B$ est une implication d'attributs disjonctive certaine dans $\mathbf{K}^?$.

D'autre part, nous avons : $B^{\Delta}_{\overline{\mathcal{K}_*}} \subseteq A^{\Delta}_{\overline{\mathcal{K}^*}}$ ssi $A^{\Pi}_{\mathcal{K}^*} \subseteq B^{\Pi}_{\mathcal{K}_*}$ (d'après la proposition 8 et P_9). Il en découle que, $\bigvee A \longmapsto \bigvee B$ est une implication d'attributs disjonctive certaine ssi $A^{\Pi}_{\mathcal{K}^*} \subseteq B^{\Pi}_{\mathcal{K}_*}$. □

Théorème 8. *$\bigvee A \longmapsto \bigvee B$ est une implication d'attributs disjonctive possible ssi $A^{\Pi}_{\mathcal{K}_*} \subseteq B^{\Pi}_{\mathcal{K}^*}$*

Preuve du Théorème 8: D'une part, par le théorème 6, $B^{\neg} \longmapsto A^{\neg}$ est une implication d'attributs disjonctive possible ssi $B^{\Delta}_{\overline{\mathcal{K}^*}} \subseteq A^{\Delta}_{\overline{\mathcal{K}_*}}$. Cela signifie également que $\bigvee A \longmapsto \bigvee$ est une implication d'attributs disjonctive possible dans $\mathbf{K}^?$.

D'autre part, $B^{\Delta}_{\overline{\mathcal{K}^*}} \subseteq A^{\Delta}_{\overline{\mathcal{K}_*}}$ est équivalent $A^{\Pi}_{\mathcal{K}_*} \subseteq B^{\Pi}_{\mathcal{K}^*}$ (d'après la proposition 8 et P_9). Par conséquent, nous obtenons que $\bigvee A \longmapsto \bigvee B$ est une implication d'attributs disjonctive possible ssi $A^{\Pi}_{\mathcal{K}_*} \subseteq B^{\Pi}_{\mathcal{K}^*}$.

□

6.5 Implications induites à partir de contextes incertains

Il peut être constaté que ni l'ACF classique ni son extension floue ne sont équipées pour représenter des situations d'incertitude partielle. On peut donc penser à l'introduction de gradations d'incertitude en changeant les croix et les blancs dans la table en degrés de probabilité, ou par des degrés de possibilité ou de nécessité. Dans le cas probabiliste, un nombre doit évaluer la probabilité qu'une propriété considérée détient pour un objet donné (son complément à 1 correspondant à la probabilité qu'il ne tient pas). Cependant, cela suppose une connaissance précise des valeurs de probabilité, ce qui n'est pas vraiment approprié si nous devons modéliser l'état d'ignorance complète. C'est pourquoi nous étudions l'utilisation du paramètre possibiliste dans ce qui suit. Dans le cadre de la possibilité, les croisements peuvent être remplacés par des degrés de nécessité positifs pour exprimer une certaine certitude qu'un objet satisfait une propriété. Les points blancs pourraient être définis par des degrés de possibilité inférieurs à 1.

Dans ce cas qui concerne l'incertitude, les attributs peuvent rester non flous, mais la certitude de la satisfaction (ou de la non satisfaction) de l'attribut par un objet n'est pas complète. Dans la suite, on ne considère que la représentation possibiliste en termes de (borne inférieure) de nécessité. C'est-à-dire que si on est α-certain que x satisfait l'attribut a, cela signifie qu'il y a une possibilité au plus égale à $1 - \alpha$ que x ne l'ait pas. On a alors une stratification d'un contexte formel en plusieurs contextes, dont chacun correspond à un niveau de certitude, de la même manière qu'une base de connaissances possibiliste peut être vue comme la stratification de plusieurs bases de connaissances classiques plus ou moins certaines. Dans pareil cas, le nombre d'objets possédant (ou ne possédant pas) un attribut croit lorsque le niveau de certitude décroit.

Dans un contexte formel incertain les cases sont remplies par des paires (α,β) de degré de nécessité. Cela veut dire que (α) désigne la nécessité que l'objet vérifie l'attribut, et (β) désigne la nécessité que l'objet ne vérifie pas l'attribut. Nous avons donc $\min(\alpha,\beta) = 0$ [49].

Exemple 13. *Soit $\mathcal{K}_{(u,v)}$ un contexte formel incertain.*

$\mathcal{K}_{(u,v)}$	a_1	a_2	a_3	a_4
x_1	(1,0)	(0.7,0)	(0.8,0)	(0,0)
x_2	(1,0)	(0.9,0)	(1,0)	(1,0)
x_3	(0,0.5)	(0,0.3)	(1,0)	(0,0.5)
x_4	(0,0.7)	(0.7,0)	(0,1)	(1,0)

TABLE 6.4 – Contexte formel incertain $\mathcal{K}_{(u,v)}$.

Les paires (1,0) et (0,1) correspondent aux situations complètement informées où on sait avec certitude que l'objet vérifie l'attribut (i.e. +), ou ne vérifie pas l'attribut (i.e. -), respectivement. La paire (0,0) reflète l'ignorance totale (i.e. ?), tandis que les paires (α,β) s.t. $1 > \max(\alpha,\beta) > 0$ correspondent aux situations d'ignorance partielle.

Considérons une paire de seuils (u, v) avec $u > 0$ et $v > 0$. $\mathcal{K}_{(u,v)}$ est un contexte formel obtenu en remplaçant toutes les entrées de la forme :

$(\alpha, 0)$ tel que $\alpha \geq u$ par (+) ;
$(\alpha, 0)$ tel que $\alpha < u$ par (?) ;
$(0,\beta)$ tel que $\beta \geq v$ par (-) ;
$(0,\beta)$ tel que $\beta < v$ par (?).

Exemple 13 (suite) On considérons une paire de seuils $(0.7, 0.6)$, on obtient le contexte formel suivant :

$\mathcal{K}_{(0.7,0.6)}$	a_1	a_2	a_3	a_4
x_1	+	+	+	?
x_2	+	+	+	+
x_3	?	-	+	?
x_4	-	+	-	+

TABLE 6.5 – Contexte formel $\mathcal{K}_{(0.7,0.6)}$.

Le contexte formel classique $(\mathcal{K}_{(u,v)})_*$ est obtenu en remplaçant par des (+) les paires $(\alpha, 0)$ tel que $\alpha \geq u$ et tout le reste par des (-).

$(\mathcal{K}_{(0.7,0.6)})_*$	a_1	a_2	a_3	a_4
x_1	+	+	+	-
x_2	+	+	+	+
x_3	-	-	+	-
x_4	-	+	-	+

TABLE 6.6 – Contexte formel $(\mathcal{K}_{(0.7,0.6)})_*$.

Exemple 13 (suite)

Le contexte formel classique $(\mathcal{K}_{(u,v)})^*$ est obtenu en remplaçant par des (-) les paires $(0, \beta)$ tel que $\beta \geq v$ et tout le reste par des (+).

Exemple 13 (suite)

$(\mathcal{K}_{(0.7,0.6)})^*$	a_1	a_2	a_3	a_4
x_1	+	+	+	+
x_2	+	+	+	+
x_3	+	-	+	+
x_4	-	+	-	+

TABLE 6.7 – Contexte formel $(\mathcal{K}_{(0.7,0.6)})^*$.

Observons alors que $(\mathcal{K}_{(u,v)})_*$ ne dépend pas de v, et augmente quand u diminue. $(\mathcal{K}_{(u,v)})^*$ ne dépend pas de u, et augmente quand v augmente. Rappelons-nous que $A^{\Delta}_{\mathcal{K}}$ augmente si $\mathcal{K}$ augmente (au sens de l'inclusion). Donc, $A^{\Delta}_{(\mathcal{K}_{(u,v)})^*}$ augmente quand v augmente. $B^{\Delta}_{(\mathcal{K}_{(u,v)})_*}$ diminue quand u augmente.

Une implication d'attributs $A \longmapsto B$ est donc d'autant plus certaine qu'il existe un u grand et un v grand tel que $A^{\Delta}_{(\mathcal{K}_{(u,v)})^*} \subseteq B^{\Delta}_{(\mathcal{K}_{(u,v)})_*}$. Donc le degré de certitude cert($A \longmapsto B$) de l'implication d'attributs $A \longmapsto B$ est égal à la valeur maximale w tel que $A^{\Delta}_{(\mathcal{K}_{(w,w)})^*} \subseteq B^{\Delta}_{(\mathcal{K}_{(w,w)})_*}$. En particulier, cert($A \longmapsto B$) = 1 ssi $A^{\Delta}_{(\mathcal{K}_{(1,1)})^*} \subseteq B^{\Delta}_{(\mathcal{K}_{(1,1)})_*}$, c'est-à-dire que les implications d'attributs certaine sont calculées avec la partie la plus certaines des données. De même un degré de possibilité est attaché aux implications d'attributs telles que $A^{\Delta}_{(\mathcal{K}_{(u,v)})_*} \subseteq B^{\Delta}_{(\mathcal{K}_{(u,v)})^*}$ qui est d'autant plus grand que u et v sont grands.

Nous considérons également les implications d'attributs disjonctives dans un contexte formel incertain. Il peut être constaté que $\overline{(\mathcal{K}_{(u,v)})_*}$ ne dépend pas de v, et augmente quand u augmente, et $\overline{(\mathcal{K}_{(u,v)})^*}$ ne dépend pas de u, et augmente quand v diminue. Rappelons que l'implication d'attributs disjonctive $\bigvee A \longmapsto \bigvee B$ est valide dans un contexte formel $\mathcal{K}$ si et seulement si l'implication d'attributs $B \longmapsto A$ est valide dans $\overline{\mathcal{K}}$. Par conséquent, le degré de certitude cert($B \longmapsto A$) est égal à la valeur maximale w tel que

$B^{\Delta}_{\overline{(\mathcal{K}_{(w,w)})_*}} \subseteq A^{\Delta}_{\overline{(\mathcal{K}_{(w,w)})^*}}$, équivalent à $\overline{B^{\Pi}_{(\mathcal{K}_{(w,w)})_*}} \subseteq \overline{A^{\Pi}_{(\mathcal{K}_{(w,w)})^*}}$, ce qui est s'écrit de façon équivalente : $A^{\Pi}_{(\mathcal{K}_{(w,w)})^*} \subseteq B^{\Pi}_{(\mathcal{K}_{(w,w)})_*}$. De même un degré de possibilité est attaché à une implication d'attributs de telle sorte que $A^{\Pi}_{(\mathcal{K}_{(u,v)})^*} \subseteq B^{\Pi}_{(\mathcal{K}_{(u,v)})_*}$, qui est d'autant plus grand que u et v sont grands.

6.6 Conclusion

Dans ce chapitre nous avons présenté notre seconde contribution qui consiste à caractériser de nouvelles formes d'implications d'attributs rencontrées dans des contextes formels en présence d'incomplétude, à savoir des implications possibles et certaines Nous avons aussi proposé une méthode qui permet de générer ces implications à partir de ces contextes formels. Enfin, nous avons initié une discussion pour une modélisation des contextes incertains.

Dans le chapitre suivant, nous allons présenter notre troisième contribution qui apporte une proposition sur l'utilisation conjointe de l'ACF et des LDs. celle-ci consiste à appliquer nos résultats théoriques à des logiques de description (LDs) où nous allons considérer le cas d'extraction des GCIs dans des environnements complets où on tient compte que de l'hypothèse du monde clos et le cas d'extraction des GCIs dans des environnements incomplets où on tient compte de l'hypothèse du monde ouvert.

CHAPITRE 7

Application aux logiques de description

Sommaire

7.1 Introduction

Dans le chapitre 4, nous avons présenté les logiques de description et les différents travaux qui utilisent l'ACF pour répondre à certaines problématiques de représentation de connaissances rencontrées dans les logiques de description.

Nous avons vu qu'un certain nombre de travaux sont concernées par l'enrichissement (l'apprentissage) d'une TBox à partir d'une ABox en utilisant des connaissances qui sont sous forme d'implications d'attributs, au moyen des méthodes de l'ACF. Par exemple, Baader et al. [15] utilisent l'analyse de concepts formels pour compléter les bases de connaissances de la logique de description. Dans cet esprit, nous avons déjà proposé une approche pour générer des GCIs à partir de descriptions d'objets dans les logiques de description $\mathcal{EL}$ [2]. Notre approche considère tous les concepts jusqu'à la profondeur de rôle maximale au lieu d'utiliser une phase d'apprentissage différente pour chaque profondeur et permet d'obtenir une base minimale de GCIs. Cette approche est présentée en détail dans la sous-section 7.2.1.

On peut remarquer que toutes les approches mentionnées ci-dessus considèrent les implications d'attributs limitées à leur forme conjonctives. Comme conséquence évidente, toutes ces approches proposées ne permettent pas d'extraire des GCIs disjonctives. Cette limitation est une conséquence de l'utilisation de l'opérateur de dérivation classique de Galois, à savoir l'opérateur de suffisance qui génère des concepts formels dans une sémantique conjonctive.

A ce titre, nous proposons dans la sous-section 7.2.2 d'enrichir la composante terminologique (TBox) d'une logique de description $\mathcal{ELU}$ en termes de GCIs en exploitant sa composante assertionnelle, et ce, en utilisant les opérateurs asymétriques de l'ACF, en particulier la composition asymétrique des opérateurs de nécessité et de possibilité étudiés dans le chapitre 5. Nous avons présenté cette approche dans [5].

Un dernier constat établi dans le cadre de cette thèse est que les approches qui sont concernées par l'apprentissage d'une TBox à partir d'une ABox [2, 8, 11, 14–16, 64, 102] utilisent l'exploration d'attributs qui est l'un des outils de l'ACF. Il suit le modèle typique, où les implications sont successivement présentées à l'expert, qui peut les accepter ou les réfuter. Si l'implication est acceptée, elle est ajoutée dans la TBox. Si l'implication est réfutée, l'expert va donner un contre-exemple, qui est ajouté à la ABox. Ces approches supposent que l'expert va toujours répondre aux questions et à la fin de l'exploration d'attributs une base de connaissance complète est obtenue. Mais dans certain cas, même l'expert du domaine peut ne pas répondre à certaines questions, et en conséquence, les connaissances obtenues après le processus d'exploration d'attributs peuvent encore être incomplètes.

Aussi, nous proposons en dernier lieu [5] l'apprentissage des GCIs à partir d'une ABox en passant par un contexte formel incomplet ce qui entraîne des GCIs plausibles et certaines aussi bien pour le cas conjonctif que pour le cas disjonctif. Cette contribution est présentée dans la section 7.3.

7.2 Extraction des GCIs dans des environnements complets

Rappelons que l'objectif de ce chapitre et de générer une TBox à partir d'une ABox en passant par une représentation intermédiaire qui est un contexte formel. La définition de ce contexte formel sera différente dans chacune des sous-sections qui vont suivre parce que chaque sous-section répond à un problème bien défini.

7.2.1 Extraction des GCIs en logique $\mathcal{EL}$

L'objectif de cette sous-section est de proposer une approche qui permet d'apprendre des inclusions générales de concepts (GCIs) à partir des descriptions d'objets données comme assertions en logique $\mathcal{EL}$, c'est-à-dire en considérant les rôles avec le constructeur existentiel, qui considère tous les concepts jusqu'à la profondeur de rôle maximale au lieu d'utiliser une phase d'apprentissage différente pour chaque profondeur et permet d'obtenir une base minimale de GCIs.

7.2.1.1 Fondements théoriques

Nous allons utiliser dans la suite de ce chapitre les notations et abréviations suivantes :

N_O : ensemble des objets du domaine d'interprétation ($\triangle^{\mathcal{I}}$)

N_C : ensemble des prédicats unaires nommés noms de concepts

N_R : ensemble des prédicats binaires appelés noms de rôles

τ : mappage qui affecte pour chaque objet une description basée sur des concepts et des rôles dans $\mathcal{EL}$

$concepts(C_i)$: sous-ensemble de N_C qui se trouve dans la description d'objet C_i

$roles(C_i)$: sous-ensemble de N_R qui se trouve dans la description d'objet C_i

$restrict_r(C_i)$: indique la description de concept se produisant dans une restriction existentielle sur le rôle r de C_i

Pour un sous-ensemble non vide $\{a_1, \ldots, a_k\}$ de noms de concepts, $\bigsqcap A$ désigne la conjonction $a_1 \sqcap \cdots \sqcap a_k$ de concepts. Pour un sous-ensemble non vide $\{r_1, \ldots, r_s\}$ des noms de rôles, nous désignons la conjonction $r_1 \sqcap \cdots \sqcap r_s$ par $\bigsqcap R$. Une description d'objet basée sur des concepts et des rôles peut maintenant être abrégée comme suit :

$$C_i = \tau(O_i) = \bigsqcap_{A \in concepts(C_i)} A \sqcap \bigsqcap_{r \in roles(C_i)} \bigsqcap_{E \in restrict_r(C_i)} \exists r.(E)$$

Exemple 14. *Soit C :Man⊓ Father⊓hasChild(Man⊓ Father) la description du concept object Bob (i.e. $C = \tau(Bob)$). Ainsi, $concepts(C)$={$Man, Father$}, $roles(C)$={$hasChild$}, et $restrict_{hasChild}(C)$ ={$Man \sqcap Father$}.*

L'analogie évidente entre une description d'objet C_i (liée au paradigme des LDs) et une ligne d'un contexte formel (liée à la théorie de l'ACF) nous permet d'obtenir la représentation intermédiaire sous la forme du contexte formel $K' = (N_O, N_C \cup N_R, \mathcal{R})$ dont l'ensemble d'objets est N_O, dont les attributs sont les noms de concepts et les noms de rôles $N_C \cup N_R$. $\mathcal{R}$ est décrit comme un sous ensemble du produit cartésien ($\mathcal{R} \subseteq N_O \times (N_C \cup N_R)$).

L'algorithme 4 permet d'obtenir la relation $\mathcal{R}$ du contexte formel K'.

Algorithme 4. Génère la relation $\mathcal{R}$ du contexte formel K'

les entrées : N_O, N_C, N_R
debut
1 : $\mathcal{R} := \{\emptyset\}$;
2 : **pour** chaque $o \in N_O$ **faire**
3 : **pour** chaque $A \in concept(\tau(o))$ **faire**
4 : $\mathcal{R} := \{(o, A)\} \cup \mathcal{R}$;
5 : **fin pour** ;
6 : **pour** chaque $r \in roles(\tau(o))$ **faire**
7 : $\mathcal{R} := \{(o, r)\} \cup \mathcal{R}$;
8 : **fin pour** ;
9 : **fin pour**
fin

Ayant ainsi obtenu un nouveau contexte formel, nous proposons de définir la base de Duquenne-Guigues $DG_{K'}$ de ce contexte formel. Rappelons d'abord qu'une telle base est non seulement non redondante, mais a aussi une cardinalité minimale [73].

Définition 40. *La base de Duquenne-Guigues $DG_{K'}$ de toutes les implications d'attributs est donnée comme suit :*
$DG_{K'}$ $=\{\ \{A_1, ..., A_m, r_1, ..., r_n\ \} \longmapsto \{A_{m+1}, ..., A_p, r_{n+1}, ..., r_k\} \mid \{A_1, ..., A_m, r_1, ..., r_n\}$ *est un pseudo-fermé* $\wedge$ $\{A_{m+1}, ..., A_p, r_{n+1}, ..., r_k\} = \{A_1, ..., A_m, r_1, ..., r_n\}^{\Delta\Delta}\}$

Où $A_1, ..., A_m, A_{m+1}, ..., A_p \in N_C$ et $r_1, ..., r_n, r_{n+1}, ..., r_k \in N_R$.
$\{\{A_1, ..., A_m, r_1, ..., r_n\ \} \longmapsto \{A_{m+1}, ..., A_p, r_{n+1}, ..., r_k\}$ est la forme générale de chaque implication $\in DG_{K'}$ mais deux sous-ensembles ($\phi(DG_{K'})$, $\psi(DG_{K'})$) peuvent être distingués et sont formalisés dans les deux définitions suivantes.

Définition 41. *Nous appelons* $\phi(DG_{K'}) = \{\ (\{m_1, ..., m_k\} \longmapsto \{m_{k+1}, ..., m_p\}) \in DG_{K'} \mid m \in \{m_1, ..., m_k, m_{k+1}, ..., m_p\} \longmapsto m \in N_C\}$ *l'ensemble d'implications de* $DG_{K'}$ *qui n'ont aucun nom de rôle (r) ni dans la partie prémisse ni dans la partie conclusion de l'implication, i.e., l'implication de* $\phi(DG_{K'})$ *est de la forme* $\{A_1, ..., A_m\} \longmapsto \{A_1, ..., A_m\}$.

Définition 42. *Nous appelons* $\psi(DG_{K'}) = \{\ (\{m_1, ..., m_k\} \longmapsto \{m_{k+1}, ..., m_p\}) \in DG_{K'} \mid \exists m \in \{m_1, ..., m_k, m_{k+1}, ..., m_p\}\ \wedge\ m \in N_R\}$ *l'ensemble d'implications de* $DG_{K'}$ *avec les implications qui ont au moins un nom de rôle r appartenant à la prémisse ou à la conclusion.*

Il y'a lieu de faire remarquer que notre approche ainsi proposée permet d'induire des implications d'attributs *imp* contenant des rôles (i.e. $imp \in \phi(DG_{K'}) \cup \psi(DG_{K'})$).

À cette fin, nous devons avoir le type de chaque rôle. Étant donné $imp_i \in \psi(DG_{K'})$, $\psi(DG^*_{K'})$ dénote l'ensemble de toutes les implications de $\psi(DG_{K'})$, tel que un type est assigné pour chaque rôle de chaque implication (exemple : si $\exists r.C$ est une description de concept avec la restriction existentielle, alors r est un nom de rôle et C est le type de r). Il en résulte une implication de $\psi(DG^*_{K'})$ qui est de la forme :
$Imp_i =$

$$\{\{A_1, ..., A_m, \exists r_1.(restrict^*_{r_1}(Imp_i)), ..., \exists r_n.(restrict^*_{r_n}(Imp_i))\} \longrightarrow \{A_{m+1}, ..., A_p, \exists r_{n+1}.(restrict^*_{r_{n+1}}(Imp_i)), ..., \exists r_k.(restrict^*_{r_k}(Imp_i))\}.$$

où $restrict^*_r(imp_i)$ indique le type du rôle $r \in \{m_1, ..., m_k, m_{k+1}, ..., m_p\}$.

Les deux propositions suivantes établissent le cadre théorique qui permet d'obtenir dans un premier temps, l'ensemble $\psi(DG^*_{K'})$, puis nous montrons dans un deuxième temps qu'à toute implication $A \longmapsto B$ de $\psi(DG^*_{K'})$ correspond une GCI (conjonctive) de la forme $\sqcap A \sqsubseteq \sqcap B$.

Proposition 10. *Supposons que Imp_i =$\{\{m_1, ..., m_k\} \longmapsto \{m_{k+1}, ..., m_p\}\}$ est une implication de $\psi(DG_{K'})$, que r est le nom de rôle de $\{m_1, ..., m_k, m_{k+1}, ..., m_p\}$, et que $\{O_1, ..., O_s\}$ est l'ensemble des objets correspondant à $\{m_1, ..., m_k, m_{k+1}, ..., m_p\}'$, alors pour chaque r : $restrict^*_r(Imp_i) = lcs(\bigsqcap resctict_r(\tau(O_1)), ..., \bigsqcap resctict_r(\tau(O_l)))$*

Il est montré dans [12] que le lcs (plus petit commun subsumeur) de deux descriptions de concepts $\mathcal{EL}$ (ou plus) existe toujours et il peut être calculé en temps polynomial.

Proposition 11. *Si $A \longmapsto B$ est une implication d'attributs valide dans $\psi(DG^*_{K'})$, alors $\sqcap A \sqsubseteq \sqcap B$.*

Preuve de la Proposition 11: Supposons que la relation $\sqcap A \sqsubseteq \sqcap B$ n'est pas valide,c'est le cas où si et seulement s'il existe un objet $g \in (\sqcap A)^{\mathcal{I}}$ et $g \notin (\sqcap B)^{\mathcal{I}}$, c'est équivalent à $g \in m^{\mathcal{I}}$ pour tous $m \in A$, et il existe $p \in B$ tel que $g \notin p^{\mathcal{I}}$ (en utilisant la sémantique de l'opérateur de conjonction). Par définition, $g \in A^{\Delta}$, et $g \notin B^{\Delta}$ alors $A^{\Delta} \nsubseteq B^{\Delta}$, Cela montre que l'implication $A \longmapsto B$ n'est pas valide. Évidemment, toutes les conclusions que nous avons faites sont réversibles. □

7.2.1.2 Apprentissage des GCIs (conjonctives)

L'algorithme 5 génère une TBox avec un nombre minimal de GCIs (General Concept Inclusion). Les autres GCIs seront trouvées par le mécanisme d'inférence en LD. Notre apprentissage prend comme entrée $\psi(DG_{K'})$, $\phi(DG_{K'})$ et K'.

L'algorithme 5 est donné comme suit :

Algorithme 5. Apprentissage d'une base minimale de GCIs

les entrées : $\psi(DG_{K'})$,$\phi(DG_{K'})$, K'
debut

$\psi(DG^*_{K'}) := \{\emptyset\}$;
pour chaque $Imp = \{\{m_1, ..., m_k\} \longmapsto \{m_{k+1}, ..., m_p\}\} \in \psi(DG_{K'})$ **faire**
  $Imp^* := Imp$;
  $\{O_1, ..., O_s\}$ est l'ensemble d'objets, correspondant à $\{m_1, ..., m_k, m_{k+1}, ..., m_p\}^{\Delta}$;
  pour chaque $r \in \{m_1, ..., m_k, m_{k+1}, ..., m_p\}$ **faire**
    $restrict^*_r(Imp) := lcs(\bigsqcap resctict_r(\tau(O_1)), ..., \bigsqcap resctict_r(\tau(O_s)))$;
    remplacer r par $\exists r.restrict^*_r(Imp)$ dans Imp^*;
  fin pour ;
  $\psi(DG^*_{K'}) := \psi(DG^*_{K'}) \cup \{Imp^*\}$;
fin pour
pour chaque $Imp^* = (A \longmapsto B) \in (\psi(DG^*_{K'}) \cup \phi(DG_{K'}))$ **faire**
  Ajouter la GCI $(\sqcap A \sqsubseteq \sqcap B)$ à la TBox ;
fin pour
fin

Exemple 15. *Illustrons notre proposition dans l'exemple suivant :*
N_C *={Man, Woman, Father, Mother, Parent, Aunt, Uncle},* N_r *={hasBrother, hasSister} et* N_O *={Hocine, Amine, Idir, Lili, Nora, Ali, Aziz, Sonia, Lilia, Lyes, Samir, Ania, Lina}.*

Nous considérons l'ensemble des objets suivant décrits dans les descriptions de concepts :
Hocine : Man ⊓ Father ⊓ Parent
Amine : Man
Idir : Man ⊓ Father ⊓ Parent
Lila : Woman ⊓ Mother ⊓ Parent
Nora : Woman
Ali : Man ⊓ ∃ hasBrother.(Aunt ⊓ Woman ⊓ Mother ⊓ Parent) ⊓ Uncle
Aziz : Man ⊓ ∃ hasSister.(Man ⊓ Father ⊓ Parent) ⊓ Uncle
Sonia : Woman ⊓ ∃ hasSister.(Mother ⊓ Parent ⊓ Woman) ⊓ Aunt
Lilia : Woman ⊓ ∃ hasBrother.(Man ⊓ Father ⊓ Parent) ⊓ Aunt
Lyes : Man ⊓ ∃ hasSister.(Woman ⊓ Mother ⊓ Parent) ⊓ Uncle
Samir : Man ⊓ ∃ hasBrother.(Uncle ⊓ Father ⊓ Parent) ⊓ Uncle
Ania : Woman ⊓ ∃ hasSiste.(Mother ⊓ Aunt ⊓ Parent ⊓ Woman) ⊓ Aunt
Lina : Woman ⊓ ∃ hasBrother.(Man ⊓ Parent ⊓ Uncle) ⊓ Aunt

Dans un premier temps, nous générons le contexte formel $K' = (N_O, N_C \cup N_R, \mathcal{R})$ *en utilisant l'algorithme 4 :*

N_O :={ Hocine, Amine, Idir, Lila, Nora, Ali, Aziz, Sonia, Lilia, Lyes, Samir, Ania, Lina }

$N_C \cup N_R$:={Man, Woman, Father, Mother, Parent, Aunt, Uncle, hasSister, hasBrother}

$\mathcal{R}$:={(Hocine, Man), (Hocine, Father), (Hocine, Parent), (Amine, Man), (Idir, Man), (Idir, Father), (Idir, Parent), (Lila, Woman), (Lila, Mother), (Lila, Parent), (Nora, Moman), (Ali, Man), (Ali, hasSister), (Ali,Uncle), (Aziz, Man), (Aziz, hasBrother), (Aziz, Uncle), (Sonia, Woman), (Sonia, hasSister), (Sonia, Aunt), (Lilia, Woman), (Lilia, hasBrother), (Lilia, Aunt), (Lyes, Man), (Lyes, hasSister), (Lyes, Uncle), (Samir, Man), (Samir, hasBrother), (Samir, Uncle), (Ania, Woman),(Ania, hasSister),(Ania, Aunt), (Lina, Woman), (Lina, hasBrother), (Lina, Aunt)}

Les implications de la base de Duquenne-Guigues résultant du contexte formel $K' = (N_O, N_C \cup N_R, \mathcal{R})$ sont les suivantes :

$\phi(DG_{K'})$ ={{Father} $\longmapsto$ {Man, Parent}, {Mother} $\longmapsto$ {Woman, Parent}, {Man, Parent} $\longmapsto$ {Father}, {Woman, Parent} $\longmapsto$ {Mother}, {Aunt} $\longmapsto$ {Woman}, {Uncle} $\longmapsto$ {Man}}

$\psi(DG_{K'})$ ={{Woman, hasSister} $\longmapsto$ {Aunt}, {Woman, hasBrother} $\longmapsto$ {Aunt}, {Man, hasSister} $\longmapsto$ {Uncle}, {Man, hasBrother} $\longmapsto$ {Uncle}}

Maintenant, nous calculons $restrict^*_r(Imp_i)$ pour chaque rôle de chaque implication $\in \psi(DG_{K'})$ en utilisant l'algorithme 5
- Imp_1 : {Woman, hasSister} $\longmapsto$ {Aunt}
{Sonia, Ania} = {Woman, hasSister, Aunt}$^\Delta$ (utiliser le contexte formel K')

$resctict^*_{hasSister}(imp_1) = lcs(\bigsqcap resctict_{hasSister}(\tau(Sonia)), \bigsqcap resctict_{hasSister}(\tau(Ania)))$

$resctict^*_{hasSister}(imp_1) = lcs((Mother \sqcap Parent \sqcap Woman), (Mother \sqcap Aunt \sqcap Parent \sqcap Woman))$

$resctict^*_{hasSister}(imp_1) = (Parent)$

l'implication {woman, $\exists$ hasSister.(Parent)} $\longmapsto$ {Aunt} est ajoutée à $\psi(DG^*_{K'})$

Toutes les autres implications $\in \psi(DG_{K'})$ sont calculées de la même manière que nous avons calculé $resctict^*_{hasSister}(imp_1)$ puis obtenir :

$\psi(DG^*_{K'})$ ={ {woman, $\exists$ hasSister.(Parent)} $\longmapsto$ {Aunt}, {woman, $\exists$ hasBrother.(Parent)} $\longmapsto$ {Aunt}, {man, $\exists$ hasSister.(Parent)} $\longmapsto$ {Uncle},{man, $\exists$ hasBrother.(Parent)} $\longmapsto$

{Uncle}}.

À la fin de l'algorithme 5, les GCIs suivantes ont été trouvées :

$Father \sqsubseteq Man \sqcap Parent$
$Mother \sqsubseteq Woman \sqcap Parent$
$Man \sqcap Parent \sqsubseteq Father$
$Woman \sqcap Parent \sqsubseteq Mother$
$Aunt \sqsubseteq Woman$
$Uncle \sqsubseteq Man$
$Woman \sqcap \exists\ hasSister.(Parent) \sqsubseteq Aunt$
$Woman \sqcap \exists\ hasBrother.(Parent) \sqsubseteq Aunt$
$Man \sqcap \exists\ hasSister.(Parent) \sqsubseteq Uncle$
$Man \sqcap \exists\ hasBrother.(Parent) \sqsubseteq Uncle$

7.2.2 Extraction des GCIs en logique $\mathcal{ELU}$

Après avoir présenté d'utiles résultats théoriques dans le chapitre 5 qui nous ont permi de caractériser et de construire une base minimale d'implications disjonctives entre attributs, nous nous intéressons maintenant à l'application de ces résultats, dans le cadre des logiques de description. Plus précisément, nous montrerons comment ces résultats peuvent être appliqués afin d'extraire les GCIs qui sont sous forme disjonctive.

Pour cela, nous utilisons une représentation intermédiaire qui consiste en un contexte formel $K'' = (N_O, N_C, \mathcal{R})$ où l'ensemble des objets est N_O, l'ensemble des attributs correspond à l'ensemble des noms de concepts N_C et $\mathcal{R} = \{(x, a) \,|\, x \in a^{\mathcal{I}} \wedge a \in N_C\}$

Nous proposons maintenant de déterminer toutes les relations de subsomption entre disjonctions des noms de concept (également appelé GCI disjonctive dénotés par $GCI_{\sqcup}$). qui sont sous la forme $a_1 \sqcup ... \sqcup a_n \sqsubseteq b_1 \sqcup ... \sqcup b_m$ tel que $a_i, b_j \in N_C$ (de manière équivalente désigné par $\bigsqcup A \sqsubseteq \bigsqcup B$) tel que $A = \{a_1, ..., a_n\}$ et $B = \{b_1, ..., b_m\}$.

Le lemme suivant met en évidence une correspondance entre l'opérateur de possibilité $(.)^{\Pi}$ dans le contexte formel $\mathcal{K}''$ et l'interprétation $\mathcal{I}$.

Lemme 9. *Soit $\mathcal{K}''$ le contexte formel induit de $\mathcal{I}$ et $A \subseteq N_C$, alors $A^{\Pi} = (\bigsqcup A)^{\mathcal{I}}$*

Preuve du Lemme 9: Par définition de l'opérateur de possibilité $(.)^{\Pi}$

$$\begin{aligned} A^{\Pi} &= \{x \in \mathcal{O} \mid \exists a \in A, x\mathcal{R}a\} \\ &= \bigcup_{a \in A} \{x \in \mathcal{O} \mid x\mathcal{R}a\} \\ &= \bigcup_{a \in A} a^{\Pi} \end{aligned}$$

et par définition de l'interprétation de la disjonction de concepts

$$\begin{aligned}(\bigsqcup A)^{\mathcal{I}} &= \{x \in \Delta^{\mathcal{I}} \mid \exists a \in A, x \in (a)^{\mathcal{I}}\} \\ &= \bigcup_{a \in A} (a)^{\mathcal{I}}\end{aligned}$$

Il est facile de voir que $(a)^{\mathcal{I}} = a^{\Pi}$, alors $A^{\Pi} = (\bigsqcup A)^{\mathcal{I}}$. □

Étant donné une ABox (connaissance assertionnelle) et sa représentation intermédiaire (contexte formel) $\mathcal{K}''$, Il sera maintenant établi que pour chaque implication d'attributs disjonctive correspond une GCI disjonctive (connaissances intensionnelle). En tant que résultat important, le théorème suivant donne la possibilité d'induire des GCIs disjonctives à partir d'implications d'attributs disjonctives.

Théorème 9. *Soit $\mathcal{K}''$ un contexte formel induit d'une ABox $\mathcal{A}$. Soit $A, B \in 2^{\mathcal{P}'}$. Si $\mathcal{K}'' \models \bigvee A \longmapsto \bigvee B$ alors la $GCI_{\sqcup}$ $(\sqcup A \sqsubseteq \sqcup B)$ est valide dans l'interprétation $\mathcal{I}$.*

Preuve du Théorème 9: Soit $\mathcal{K}'' \models \bigvee A \longmapsto \bigvee B$ alors :

$$\begin{array}{lll} & A \subseteq ((B)^{\Pi})^{\mathrm{N}} & \\ \Longrightarrow & (A)^{\Pi} \subseteq (((B)^{\Pi})^{\mathrm{N}}))^{\Pi} & \text{(en utilisant la propriété } P_3\text{)} \\ \Longrightarrow & (A)^{\Pi} \subseteq (B)^{\Pi} & \text{(en utilisant la propriété } P_8\text{)} \\ \Longrightarrow & (\sqcup A)^{\mathcal{I}} \subseteq (\sqcup B)^{\mathcal{I}} & \text{(par le lemme 9)} \\ \Longrightarrow & \sqcup A \sqsubseteq \sqcup B \text{ est valide dans l'interprétation } \mathcal{I} & \end{array}$$

□

Pour chaque interprétation $\mathcal{I}$, on peut définir son contexte formel induit $\mathcal{K}''$ et obtenir un ensemble de $GCI_{\sqcup}$ $(\mathcal{T}_{\sqcup})$ à partir d'une base minimale d'implications d'attributs disjonctives ($Base_{\mathrm{N}\Pi}$) en utilisant le théorème 9. Il devient évident que chaque $GCI_{\sqcup} \in \mathcal{T}_{\sqcup}$ est valide dans $\mathcal{I}$. Nous obtenons immédiatement la proposition suivante.

Proposition 12. *Linterprétation $\mathcal{I}$ est un modèle de $\mathcal{T}_{\sqcup}$.*

La logique de description $\mathcal{ELU}$ est l'extension de la logique de description $\mathcal{EL}$ avec l'opérateur de disjonction. Dans [9] les auteurs montrent que $\mathcal{T}_{\mathcal{ELU}}$ est une $\mathcal{T}_{\mathcal{EL}}$ étendu avec une $GCI_{\sqcup}$ de la forme $a_1 \sqsubseteq b_1 \sqcup ... \sqcup b_m$ avec $m \geq 2$, Il en résulte que notre approche permet d'étendre la TBox $\mathcal{T}_{\mathcal{EL}}$ à la TBox $\mathcal{T}_{\mathcal{ELU}}$ donné dans la logique de description $\mathcal{ELU}$.

7.3 Extraction des GCIs dans des environnements incomplets

Dans le paradigme des Logiques de Description, l'interprétation des connaissances respecte l'hypothèse du monde ouvert. Ainsi, et contrairement à la sémantique des bases de données traditionnelles, l'absence d'information représente l'ignorance plutôt qu'une information négative. Pour enrichir la comparaison avec les bases de données, répondre à une requête pour un système bâti sur les LDs nécessite d'effectuer un raisonnement logique souvent plus complexe qu'une simple recherche pour vérifier la présence d'une information, car le système doit souvent considérer plusieurs interprétations possibles, une conséquence de l'hypothèse du monde ouvert.

C'est dans cet esprit que nous mettons en évidence dans cette section la pertinence et surtout l'utilité des résultats obtenus dans le chapitre 6, c'est à dire les résultats concernant les implications d'attributs dans des contextes incomplets. En effet, nous montrons ainsi que la notion d'incomplétude peut permettre d'apporter une solution au problème fondamental résultant de l'utilisation conjointe de deux paradigmes (en l'occurrence l'ACF et les LDs) reposant respectivement sur des hypothèses fondamentales différentes (en l'occurrence l'hypothèse du monde clos-CWA et l'hypothèse du monde ouver-OWA). Nous illustrons ceci à travers l'exemple suivant :

Exemple 16. *Nous considérons l'ensemble des assertions $\mathcal{A}$:*

$\mathcal{A}$ *:=*$\Big\{$*Man(John),Man(Peter), ¬Man(Maria), ¬Man(Clara) Woman(Maria), Woman(Clara), ¬Woman(John), ¬Woman(Peter), ¬Father(John), ¬Father(Maria),¬Father(Clara), Mother(Clara), ¬Mother(John), ¬Mother(Maria), ¬Mother(Peter), Parent(Clara), Parent(Peter), ¬Parent(John)*$\Big\}$.

Nous pouvons vérifier que $N_O = \{John, Peter, Maria, Clara\}, N_C = \{Man, Woman, Father\}$.

Nous considérons également l'interprétation $\mathcal{I} =< \{John, Maria, Peter, Clara\}, (.)^{\mathcal{I}} >$ *avec*

$Man^{\mathcal{I}}$ = *{John, Peter}*
$Woman^{\mathcal{I}}$ = *{Maria, Clara}*
$(\neg Father)^{\mathcal{I}}$ = *{John, Maria, Clara}*
$Mother^{\mathcal{I}}$ = *{Clara}*
$Parent^{\mathcal{I}}$ = *{Peter, Clara}*
$(\neg Man)^{\mathcal{I}}$ = *{Maria, Clara}*
$(\neg Woman)^{\mathcal{I}}$ = *{John, Peter}*
$(\neg Mother)^{\mathcal{I}}$ = *{John, Maria, Peter}*
$(\neg Parent)^{\mathcal{I}}$ = *{John}*

Il a été montré [2,16,102] que l'apprentissage des GCIs à partir d'une ABox nécessite, dans une première étape, de générer un contexte formel à partir d'une ABox afin d'extraire toutes les implications d'attributs. Dans une deuxième étape, une correspondance formelle est établie entre une implication d'attributs donnée et une GCI relative.

Cependant, il est important de se rappeler que l'hypothèse du monde fermée CWA (un fait dont nous ne savons rien est supposé être faux) est souvent associé à des contextes formels, où des espaces blancs sont censés exprimer la fausseté. Alors que l'hypothèse du monde ouvert (un fait dont nous ne savons rien peut être vrai ou faux) est la compréhension naturelle d'une ABox. Cette divergence nous empêche d'avoir un mappage simple entre les logiques de description et l'ACF.

Pour palier à ce type de problèmes, nous proposons une représentation intermédiaire, à savoir un contexte formel qui doit considérer l'hypothèse du monde ouvert, et une logique de description où la négation explicite est autorisée. Afin de tenir compte de cette hypothèse pour apprendre les GCIs correctes, nous proposons de considérer des contextes formels portant sur l'ignorance partielle.

Considérons le contexte formel incomplet $\mathbf{K}^? = (N_O, N_C, \{+, -, ?\}, \mathbf{R})$ dont l'ensemble d'objets est N_O, dont les attributs sont les noms des concepts N_C, et $\mathbf{R}$ est défini comme suit

- $(x, a, +) \in \mathbf{R}$ si $a(x)$ est dans la ABox ;
- $(x, a, -) \in \mathbf{R}$ si $\neg a(x)$ est dans la ABox ;
- $(x, a, ?) \in \mathbf{R}$ sinon.

Le contexte formel incomplet $\mathbf{K}^?$ induit à partir de l'ensemble $\mathcal{A}$ des assertions et l'interprétation $\mathcal{I}$ donnée dans l'exemple 16 est affichée dans la table 7.1.

R	Man	Woman	Father	Mother	Parent
John	+	−	−	−	−
Maria	−	+	−	−	?
Peter	+	−	?	−	+
Clara	−	+	−	+	+

TABLE 7.1 – Contexte formel incomplet $\mathbf{K}^?$ pour la ABox de l'exemple 16

Comme déjà vu dans le chapitre 6, traiter des contextes formels incomplets entraîne des implications d'attribut possibles et certaines (dans la sémantique disjonctive et conjonctive). Étant donné que ces types d'implications sont à la base du processus d'apprentissage, différents cas peuvent survenir selon que l'implication d'attributs correspondante est possible ou certaine. Il est bien convenu dans la littérature que la subsomption des concepts (GCI) est universellement vérifiée (à savoir dans toutes les interprétations). Par conséquent, nous considérons qu'une implication d'attributs certaine, qui est vérifiée dans tous les contextes formels, correspond à une GCI dans sa définition stricte. Nous considérons aussi qu'une implication d'attributs possible, qui est vérifiée dans au moins un contexte formel, correspond à une GCI plausible.

7.3.1 Extraction des GCIs conjonctives

Le processus d'apprentissage tire parti des résultats précédents afin de déterminer toutes les relations de subsomption entre les conjonctions de noms de concepts (également appelés GCIs conjonctifs et désignés par $GCI_{\sqcap}$), qui sont sous la forme $a_1 \sqcap ... \sqcap a_n \sqsubseteq b_1 \sqcap ... \sqcap b_m$ tel que $a_i, b_j \in N_C$ (de manière équivalente désigné par $\bigsqcap A \sqsubseteq \bigsqcap B$) tel que $A = \{a_1, ..., a_n\}$ et $B = \{b_1, ..., b_m\}$.

soit $\mathcal{K}_j = (N_O, N_C, \mathcal{R}_j)$ un contexte formel booléenne possible du contexte formel incomplet $\mathbf{K}^? = (N_O, N_C, \{+, -, ?\}, \mathbf{R})$ (obtenu de l'interprétation $\mathcal{I}$). Il est clair qu'il existe une interprétation possible $\mathcal{I}_j$ qui correspond à $\mathcal{K}_j$ (la ABox associée à $\mathcal{K}_j$ contient strictement celui associé à $\mathbf{K}^?$).

Le lemme suivant met en évidence une correspondance entre les opérateurs de dérivation de suffisance $(.)^\Delta$ dans le contexte formel $\mathcal{K}_j$ et l'interprétation $\mathcal{I}_j$.

Lemme 10. *Soit $A \subseteq N_C$, alors $A^\Delta_{\mathcal{K}_j} = (\bigsqcap A)^{\mathcal{I}_j}$*

Preuve du Lemme 10: Par définition, l'opérateur de dérivation de suffisance $(.)^\Delta$

$$A^\Delta_{\mathcal{K}_j} = \{x \in N_O \mid \forall a \in A, x\mathcal{R}_j a\} = \bigcap_{a \in A} \{a\}^\Delta_{\mathcal{K}_j}$$

et par définition de l'interprétation de conjonction de concepts

$$(\bigsqcap A)^{\mathcal{I}_j} = \{x \in \Delta^{\mathcal{I}} \mid \forall a \in A, x \in (a)^{\mathcal{I}_j}\} = \bigcap_{a \in A} (a)^{\mathcal{I}_j}$$

Il est facile de voir que $(a)^{\mathcal{I}_j} = \{a\}^\Delta_{\mathcal{K}_j}$, alors $A^\Delta_{\mathcal{K}_j} = (\bigsqcap A)^{\mathcal{I}_j}$. □

Le théorème suivant donne une méthode pour induire des GCIs conjonctives (c'est-à-dire des GCIs valides dans chaque interprétation $\mathcal{I}_j$) d'une implication d'attributs conjonctive certaine pour lequel une caractérisation simple est donnée dans le théorème 5.

Théorème 10. *$\bigsqcap A \sqsubseteq \bigsqcap B$ est une GCI conjonctive ssi $A \longmapsto B$ est une implication d'attributs certaine dans $\mathbf{K}^?$.*

Preuve du Théorème 10: $A \longmapsto B$ est une implication d'attributs certaine dans $\mathbf{K}^? \iff A \longmapsto B$ est valide dans tous $\mathcal{K}_j \iff A^\Delta_{\mathcal{K}_j} \subseteq B^\Delta_{\mathcal{K}_j}$ dans tous $\mathcal{K}_j$ (par définition) $\iff (\bigsqcap A)^{\mathcal{I}_j} \subseteq (\bigsqcap B)^{\mathcal{I}_j}$ dans toutes les interprétations possibles $\mathcal{I}_j$ (par le lemme 10) $\iff \bigsqcap A \sqsubseteq \bigsqcap B$ est valide dans toute interprétation $\mathcal{I}_j \iff \bigsqcap A \sqsubseteq \bigsqcap B$ est une GCI conjonctive. □

Ainsi, il est facile d'obtenir un ensemble $\mathcal{B}_{GCI_{\sqcap}}$ de GCIs conjonctives de l'interprétation $\mathcal{I}$, donée dans l'exemple 16, en utilisant le théorème 5 et le théorème 10 :

$$\mathcal{B}_{GCI_{\sqcap}} = \{Mother \sqsubseteq Woman \sqcap Parent, Woman \sqcap Parent \sqsubseteq Mother\}$$

En revanche, lorsque l'implication d'attributs est possible, à savoir vérifiée dans un contexte formel, on peut considérer que la GCI correspondante est plausible tel que défini cidessous [99] :

Définition 43. *Une GCI est dite être plausible si elle est valide dans au moins une interprétation possible $\mathcal{I}_j$, où une interprétation possible correspond à un contexte formel possible $\mathcal{K}_J$.*

Le théorème suivant donne une méthode pour induire des GCIs conjonctives plausibles à partir d'une implication d'attributs possible pour laquelle une caractérisation simple est donnée dans le théorème 6.

Théorème 11. *$\bigsqcap A \sqsubseteq \bigsqcap B$ est une GCI conjonctive plausible ssi $A \longmapsto B$ est une implication d'attributs possible dans $\mathbf{K}^?$.*

Preuve du Théorème 11: $A \longmapsto B$ est une implication d'attributs possible dans $\mathbf{K}^? \Longleftrightarrow A \longmapsto B$ est valide dans au moins un contexte formel booléen possible $\mathcal{K}_j \Longleftrightarrow A^{\Delta}_{\mathcal{K}_j} \subseteq B^{\Delta}_{\mathcal{K}_j}$ (par définition) $\Longleftrightarrow (\bigsqcap A)^{\mathcal{I}_j} \subseteq (\bigsqcap B)^{\mathcal{I}_j}$ (par le lemme 10) $\Longleftrightarrow \bigsqcap A \sqsubseteq \bigsqcap B$ est valide dans $\mathcal{I}_j \Longleftrightarrow \bigsqcap A \sqsubseteq \bigsqcap B$ est une GCI conjonctive plausible. □

Notez que la prise en compte d'une GCI plausible est risquée car elle peut être en conflit avec d'autres, ce qui peut endommager la cohérence de la TBox. Donc, ce type d'élément de connaissance doit être traité avec précaution et rejeté s'il est en contradiction avec d'autres déjà acceptés par exemple.

7.3.2 Extraction des GCIs disjonctives

Cette section tente de suggérer que les résultats obtenus ci-dessus, concernant les implications d'attributs disjonctives et les contextes incomplets, peuvent être pertinents pour l'application de l'ACF aux logiques de description.

Nous pouvons faire une étude similaire afin de déterminer toutes les relations de subsomption entre les noms de concept disjonctives (également appelé GCI disjonctive et noté par $GCI_{\sqcup}$). Les GCIs disjonctives ($GCI_{\sqcup}$) sont de la forme $a_1 \sqcup ... \sqcup a_n \sqsubseteq b_1 \sqcup ... \sqcup b_m$ tel que a_i, $b_j \in N_C$. (équivalemment noté par $\bigsqcup A \sqsubseteq \bigsqcup B$) tel que $A = \{a_1, ..., a_n\}$ et $B = \{b_1, ..., b_m\}$.

Le lemme suivant met en évidence une correspondance entre les opérateurs de dérivation de possibilité $(.)^{\Pi}$ dans le contexte formel possible $\mathcal{K}_j$ compatible avec le contexte formel incomplet $\mathbf{K}^?$ et l'interprétation $\mathcal{I}_j$.

Lemme 11. *Soit $A \subseteq N_C$, alors $A^{\Pi}_{\mathcal{K}_j} = (\bigsqcup A)^{\mathcal{I}_j}$*

Preuve du Lemme 11: Par définition de l'opérateur de dérivation de possibilité $(.)^{\Pi}$

$$A^{\Pi}_{\mathcal{K}_j} = \{x \in N_O \mid \exists a \in A, x\mathcal{R}_j a\} = \bigcup_{a \in A} \{a\}^{\Pi}_{\mathcal{K}_j}$$

et par définition de l'interprétation de disjonction de concepts

$$(\bigsqcup A)^{\mathcal{I}_j} = \{x \in \Delta^{\mathcal{I}} \mid \exists a \in A, x \in (a)^{\mathcal{I}_j}\} = \bigcup_{a \in A} (a)^{\mathcal{I}_j}$$

Il est facile de voir que $(a)^{\mathcal{I}_j} = \{a\}^{\Pi}_{\mathcal{K}_j}$, alors $A^{\Pi}_{\mathcal{K}_j} = (\bigsqcup A)^{\mathcal{I}_j}$. □

Étant donné une GCI disjonctive ($GCI_{\sqcup}$) est valide dans chaque interprétation $\mathcal{I}_j$, il sera maintenant établi que pour chaque implication d'attributs disjonctive certaine correspond une GCI disjonctive (Connaissances intensionnelle).

Théorème 12. *$\bigsqcup A \sqsubseteq \bigsqcup B$ est une GCI disjonctive ssi $\bigvee A \longmapsto \bigvee B$ est une implication d'attributs disjonctive certaine dans $\mathbf{K}^?$.*

Preuve du Théorème 12: $\bigvee A \longmapsto \bigvee B$ est une implication d'attributs disjonctive certaine dans $\mathbf{K}^? \Longleftrightarrow \bigvee A \longmapsto \bigvee B$ est valide dans tout $\mathcal{K}_j \Longleftrightarrow A^{\Pi}_{\mathcal{K}_j} \subseteq B^{\Pi}_{\mathcal{K}_j}$ in all $\mathcal{K}_j$ (par la proposition 8) $\Longleftrightarrow$ $(\bigsqcup A)^{\mathcal{I}_j} \subseteq (\bigsqcup B)^{\mathcal{I}_j}$ dans toutes les interprétations possibles $\mathcal{I}_j$ (par le lemme 11) $\Longleftrightarrow \bigsqcup A \sqsubseteq \bigsqcup B$ est valide dans toute interprétation $\mathcal{I}_j \Longleftrightarrow \bigsqcup A \sqsubseteq \bigsqcup B$ est une GCI disjonctive. □

Ainsi, il est facile d'induire un ensemble $\mathcal{D}_{GCI_{\sqcup}}$ de GCIs disjonctifs de l'interprétation $\mathcal{I}$, donnée dans l'exemple 16, en utilisant le théorème 7 et le théorème 12,

$$\mathcal{D}_{GCI_{\sqcup}} = \{Father \sqsubseteq Man, Mother \sqsubseteq Woman, Father \sqcup Mother \sqsubseteq Parent, Parent \sqsubseteq Father \sqcup Mother\}$$

Une GCI disjonctive ($GCI_{\sqcup}$) est dite plausible si elle est valable dans au moins une interprétation possible $\mathcal{I}_j$. Le théorème suivant donne une méthode pour induire des GCIs disjonctives plausibles à partir d'une implication d'attributs disjonctive possible pour laquelle une simple caractérisation est donnée dans le théorème 8.

Théorème 13. *$\bigsqcup A \sqsubseteq \bigsqcup B$ est une GCI disjonctive plausible ssi $\bigvee A \longmapsto \bigvee B$ est une implication d'attributs disjonctive possible dans $\mathbf{K}^?$.*

Preuve du Théorème 13: $\bigvee A \longmapsto \bigvee B$ est une implication d'attributs disjonctive possible dans $\mathbf{K}^? \Longleftrightarrow \bigvee A \longmapsto \bigvee B$ est valide dans au moins un contexte formel booléen possible $\mathcal{K}_j \Longleftrightarrow$ $A^{\Pi}_{\mathcal{K}_j} \subseteq B^{\Pi}_{\mathcal{K}_j}$ (par la proposition 8) $\Longleftrightarrow (\bigsqcup A)^{\mathcal{I}_j} \subseteq (\bigsqcup B)^{\mathcal{I}_j}$ (par le lemme 11) $\Longleftrightarrow \bigsqcup A \sqsubseteq \bigsqcup B$ est valide dans $\mathcal{I}_j \Longleftrightarrow \bigsqcup A \sqsubseteq \bigsqcup B$ est une GCI disjonctive plausible. □

Les résultats obtenus dans cette section indiquent clairement qu'il est possible d'étendre les techniques d'induction basées sur l'ACF dans les logiques de description à l'extraction des règles disjonctives et une gestion plus explicite de l'incomplétude des informations. C'est un sujet pour recherche plus poussé.

7.4 Conclusion

Dans ce chapitre nous avons proposé en premier lieu une approche qui tient compte des rôles et de la restriction existentielle afin d'extraire des GCIs exprimées en logique de description $\mathcal{EL}$. Ensuite on a enrichi la composante terminologique (TBox) d'une logique de description $\mathcal{ELU}$ en utilisant les opérateurs asymétriques de l'ACF, en particulier la composition asymétrique des opérateurs de nécessité et de possibilité étudiés dans le chapitre 5 et cela dans des environnements complets où on tient compte que de l'hypothèse du monde clos.

En second lieu, nous avons proposé une approche qui traite le cas où même l'expert du domaine peut ne pas répondre à certaines questions, et en conséquence, les connaissances obtenues après le processus d'exploration d'attributs peuvent encore être incomplètes. Ce qui nécessite de considérer des environnements incomplets où on tient compte de l'hypothèse du monde ouvert. Nous avons appris des GCIs à partir d'une ABox en passant par un contexte formel incomplet ce qui a entraîné des GCIs plausibles et des GCIs certaines aussi bien pour le cas conjonctif que pour le cas disjonctif.

CONCLUSION

Sommaire

I Résumé conclusif

Le présent travail de thèse se situe à la frontière de trois thématiques de recherche à savoir : l'analyse de concepts formels, les logiques de description et les informations incomplètes. Nous avons consacré les trois premiers chapitres de cette thèse à un état de l'art des domaines de recherche dans lesquels se situe notre travail ; les trois derniers chapitres présentent nos différentes contributions.

Dans cette thèse, nous avons proposé d'élargir la sémantique sous-jacente des implications d'attributs en considérant les "implications d'attributs disjonctives". Cette proposition se base sur l'utilisation d'opérateurs de dérivation autre que l'opérateur de Galois classiquement utilisé, à savoir les opérateurs de possibilités et de nécessité.

Notre première contribution est fondée sur une interprétation possibiliste de la théorie de l'ACF. Dans cet esprit, nous avons originalement proposé une nouvelle composition asymétrique ($(.)^{\mathrm{N}\circ\Pi}$) de l'opérateur de possibilité $(.)^{\Pi}$ et de l'opérateur de nécessité $(.)^{\mathrm{N}}$. Il en résulte un nouveau type de concepts formels qui sont des paires "d'ouvert-fermé" appelés NΠ-paires. Ensuite nous avons généré le NΠ-treillis. Nous avons exploité le NΠ-treillis pour extraire des implications d'attributs disjonctives, à savoir des formules de la forme $a_1 \vee ... \vee a_n \longmapsto b_1 \vee ... \vee b_m$. En profitant de la structure algébrique de l'ensemble de toutes les NΠ-paires obtenues en utilisant la composition asymétrique ($(.)^{\mathrm{N}\circ\Pi}$), nous avons réussi à caractériser et obtenir une base minimale d'implications d'attributs disjonctives. La minimalité a été théoriquement prouvée au moyen de "Pseudo-NΠ-Fermé".

Notre seconde contribution concernait les contextes formels en présence d'incomplétude. En effet, il est largement admis que dans certaines applications concrètes (réelles), les contextes formels peuvent comporter des informations incomplètes. Dans le cas d'un contexte formel incomplet, nous devons distinguer entre le cas de ne pas savoir si un objet possède un attribut ou non, et le cas (exclusif) où il est bien connu que l'objet possède ou ne possède pas cet attribut. Nous avons traité les implications extraites de ces contextes formels incomplets en définissant les “implications certaines” qui tiennent dans tous les mondes possibles compatibles avec l'information incomplète et les “implications possibles” qui tiennent dans au moins une situation compatible avec l'information incomplète, et cela pour la sémantique conjonctive et disjonctive.

Nous avons traité aussi le cas d'un contexte formel incertain, les cases sont remplies par des paires (α,β) où α est le degré de nécessité que l'objet ait l'attribut et β est le degré de nécessité que l'objet n'ait pas l'attribut avec $min(\alpha,\beta) = 0$. Nous avons proposé une approche qui permet d'obtenir, à partir d'un contexte formel incertain, des implications d'attributs à la fois pour la sémantique conjonctive et disjonctive, en calculant pour chaque implication son degré de certitude.

Notre troisième contribution apporte une proposition sur l'utilisation conjointe de l'ACF et des LDs. celle-ci consiste à appliquer nos résultats théoriques à des logiques de description où nous avons considéré deux cas :

-Cas 1 : extraction des GCIs dans des environnements complets où l'on tient compte que de l'hypothèse du monde clos. Nous avons proposé une approche qui tient compte des rôles et de la restriction existentielle afin d'extraire des GCIs exprimées en logique de description $\mathcal{EL}$. Nous avons proposé une autre approche qui permet de générer des GCIs disjonctive afin d'enrichir la base terminologique exprimée en logique de description $\mathcal{EL}$ et avoir des GCIs exprimées en logique de description $\mathcal{ELU}$.

-Cas 2 : extraction des GCIs dans des environnements incomplets où l'on tient compte de l'hypothèse du monde ouvert. Dans ce cas, notre approche repose sur des hypothèses différentes des travaux trouvés jusque là dans la littérature. Nous avons supposé que l'expert ne peut pas répondre à toutes les questions et, en conséquence, les connaissances obtenues après le processus d'exploration peuvent encore être incomplètes, ce qui conduit à deux types de GCIs, les GCIs certaines et les GCIs plausibles.

II Perspectives

Ces résultats ouvrent la voie à de nouvelles recherches :

1. Nous nous sommes concentrés uniquement sur la composition hybride des opérateurs $(.)^{N\Pi}$. Une recherche plus poussée devrait porter sur l'étude de la composition

d'autres opérateurs, tels que $(.)^{\Pi\Delta}$, $(.)^{\nabla\Delta}$, etc.

2. Les contextes incomplets peuvent être affinés en introduisant des qualités de certitude selon lesquelles un objet satisfait un attribut ou ne le satisfait pas, en utilisant la version graduée de la théorie des possibilités. Cette représentation a déjà été introduite dans [38]. Les étapes préliminaires visant à attribuer des grades de probabilité et de certitude aux implications d'attributs extraites de ces contextes incomplets généralisés peuvent être trouvées dans [3].

3. En ce qui concerne les aspects algorithmiques, il est clair que les mêmes méthodes de l'ACF peuvent être utilisées pour dériver des bases de règles conjonctives et disjonctives puisque les règles disjonctives d'un contexte sont en correspondance un à un avec les règles conjonctives induites par le contexte complémentaire. En ce qui concerne les contextes incomplets, il a été montré qu'une partie de la complexité peut être apprivoisée en n'utilisant que deux contextes formels (the upper and the lower) au lieu de tous les contextes compatibles avec les informations disponibles. Néanmoins, la conception d'algorithmes efficaces pour l'induction de règles possibles et certaines nécessite des efforts supplémentaires au-delà des méthodes existantes.

4. Le problème du calcul des implications, c'est-à-dire des régularités de la forme "quand il y a A, il y a B", dans des ensembles de données composés d'objets décrits par des attributs. Calculer toutes les implications peut être vu comme l'énumération d'ensembles d'attributs appelés pseudo-intensions. Nous savons que ces pseudo-intensions ne peuvent pas être énumérées avec un délai polynomial dans l'ordre lectique mais aucun résultat n'existe, à l'heure actuelle, pour d'autres ordres. Bien que certains algorithmes existants n'énumèrent pas forcément dans l'ordre lectique, aucun n'a un délai polynomial. Cette absence de connaissances sur les autres ordres autorise toujours l'existence d'un algorithme avec délai polynomial et le trouver serait une avancée utile et significative. Malheureusement, les algorithmes actuels ne nous autorisent pas à choisir l'ordre d'énumération, ce qui complique considérablement et inutilement l'étude de l'influence de l'ordre dans la complexité. C'est donc pour aller vers une meilleure compréhension du rôle de l'ordre dans l'énumération des pseudo-intensions qu'il faut proposer des algorithmes qui peuvent réaliser cette énumération dans n'importe quel ordre qui respecte la relation d'inclusion.

Bibliographie

[1] R. Fuentes Gonzales A. Burusco. Contexts with multiple weighted values. In *International J. uncertainty, fuzziness and knowledge based systems*, 2001.

[2] Z. Ait-Yakoub and Y. Djouadi. Generating GCIs axioms from objects descriptions in $\mathcal{EL}$ -description logics. In *Modeling Approaches and Algorithms for Advanced Computer Applications (CIIA 2013)*, volume 488 of *Studies in Computational Intelligence*, pages 75–84. Springer International Publishing, 2013.

[3] Z. Ait-Yakoub, Y. Djouadi, D. Dubois, and H. Prade. From a possibility theory view of formal concept analysis to the possibilistic handling of incomplete and uncertain contexts. In *CEUR Proc. Workshop FCA for AI*, The Hague, 2016.

[4] Zina Ait-Yakoub, Yassine Djouadi, Didier Dubois, and Henri Prade. Concepts formels incomplets ou incertains : vers une généralisation possibiliste de l'analyse formelle de concepts (regular paper). In *Rencontres Francophones sur la Logique Floue et ses Applications (LFA), La Rochelle, 02/11/2016-04/11/2016*, pages 203–210, http ://www.cepadues.com/, 2016. Cépaduès Editions.

[5] Zina Ait-Yakoub, Yassine Djouadi, Didier Dubois, and Henri Prade. Asymmetric composition of possibilistic operators in formal concept analysis. application to the extraction of attribute implications from incomplete contexts. *International Journal of Intelligent Systems*, 32(12) :1285–1311, 2017.

[6] Bassem Alsahwa. *Representation of an environment by a multi-sensor system : fusion and scene interpretation*. Theses, Télécom Bretagne ; Université de Rennes 1, December 2014.

[7] Zainab Assaghir, Mehdi Kaytoue, and Henri Prade. A possibility theory-oriented discussion of conceptual pattern structures (regular paper). In Amol Deshpande and Anthony Hunter, editors, *International Conference on Scalable Uncertainty Management (SUM), Toulouse, France, 27/09/2010-29/09/2010*, number 6379 in LNAI, pages 70–83, http ://www.springerlink.com, 2010. Springer.

[8] F. Baader. Computing a minimal representation of the subsumption lattice of all conjunctions of concepts defined in a terminology. In *Proceedings of the International Symposium on Knowledge Retrieval, Use, and Storage for Efficiency, KRUSE 95*, pages 168–178, Santa Cruz, USA, 1995.

[9] F. Baader, S. Brandt, and C. Lutz. Pushing the el envelope. LTCS-Report LTCS-05-01, Chair for Automata Theory, Institute for Theoretical Computer Science, Dresden University of Technology, Germany, 2005.

[10] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P.F. Patel-Schneider, editors. *The Description Logic Handbook : Theory, Implementation, and Applications*. Cambridge University Press, 2003.

[11] F. Baader and F. Distel. Exploring finite models in the description logic ELgfp. In S. Ferre and S. Rudolph, editors, *Proceedings of the 7th International Conference on Formal Concept Analysis, (ICFCA)*, volume 5548 of *Lecture Notes in Artificial Intelligence*, pages 146–161. Springer Verlag, 2009.

[12] F. Baader, R. Kuesters, and R. Molitor. Computing least common subsumers in description logics with existential restrictions. Technical report, 1999.

[13] F. Baader and R. Molitor. Building and structuring description logic knowledge bases using least common subsumers and concept analysis. In B. Ganter and G. Mineau, editors, *Conceptual Structures : Logical, Linguistic, and Computational Issues – Proceedings of the 8th International Conference on Conceptual Structures (ICCS 2000)*, volume 1867 of *Lecture Notes in Artificial Intelligence*, pages 290–303. Springer Verlag, 2000.

[14] F. Baader and B. Sertkaya. Applying formal concept analysis to description logics. In P. Eklund, editor, *Proceedings of the 2nd International Conference on Formal Concept Analysis (ICFCA)*, volume 2961 of *Lecture Notes in Artificial Intelligence*, pages 261–286. Springer, 2004.

[15] Franz Baader, Bernhard Ganter, Ulrike Sattler, and Baris Sertkaya. Completing description logic knowledge bases using formal concept analysis. LTCS-Report LTCS-06-02, Chair for Automata Theory, Institute for Theoretical Computer Science, Dresden University of Technology, Germany, 2006. See http ://lat.inf.tu-dresden.de/research/reports.html.

[16] A. Bazin and J. Ganascia. Completing terminological axioms with formal concept analysis. In *International Conference on Formal Concept Analysis (ICFCA)*, pages 30–39, 2012.

[17] Olivier Bedel, Sébastien Ferré, Olivier Ridoux, and Erwan Quesseveur. GEOLIS : A Logical Information System for Geographical Data. *Revue Internationale de Géomatique*, 17(3-4) :371–390, 2008.

[18] R Belohlavek. Fuzzy galois connections. *Math. Logic Quart*, 45 :497–504, 1999.

[19] R. Belohlavek. Fuzzy relational systems : Foundations and principles. In *Kluwer academic/ plunium publisher, New York*, 2002.

[20] R. Belohlávek and V. Vychodil. What is a fuzzy concept lattice ? In *CLA 2005, Proceedings of the 3rd International Workshop*, volume 162, pages 34–45, 2005.

[21] Garrett Birkhoff. Lattice theory. In *Colloquium Publications*, volume 25. Amer. Math. Soc., 3. edition, 1967.

[22] J.-P. Bordat. Calcul pratique du treillis de galois d'une correspondance. In *Mathématiques et Sciences Humaines*, pages 31–47, 1986.

[23] B. Bouchon. *La logique floue et ses applications*. Addison Wesley, New York, 1995.

[24] Ronald J. Brachman and James G. Schmolze. An overview of the kl-one knowledge representation system. *Cognitive Science*, 9(2) :171–216, 1985.

[25] P. Burmeister and R. Holzer. Treating incomplete knowledge in formal concept analysis. In B. Ganter, G. Stumme, and R. Wille, editors, *Formal Concept Analysis*, pages 114–126. Springer, 2005.

[26] Fuentes-González Ramón Burusco Juandeaburre, Ana. The study of the l-fuzzy concept lattice. *Mathware and Soft Computing*, 1(3) :209–218, 1994.

[27] C. Carpineto and G. Romano. Galois : An order-theoretic approach to conceptual clustering. In *Proceedings of the 10th International Conference on Machine Learning (ICML'90)*, pages 33–40, July 1993.

[28] C. Carpineto and G. Romano. *Concept Data Analysis : Theory and Applications*. John Wiley & Sons, Chichester, England ; Hoboken, NJ, 2004.

[29] Claudio Carpineto and Giovanni Romano. A lattice conceptual clustering system and its application to browsing retrieval. In *Machine Learning*, pages 95–122, 1996.

[30] Claudio Carpineto and Giovanni Romano. Exploiting the potential of concept lattices for information retrieval with credo. 10(8) :985–1013, 2004.

[31] M. Chein. Algorithme de recherche des sous-matrices premières d'une matrice. In *Bull. Math. Soc. Sci. Math. R.S. Roumanie*, pages 21–25, 1969.

[32] O. Colot. *Systèmes de perception d'informations incertaines - Application au diagnostic médical*. 2000.

[33] A. Coulet, M. Smail-Tabbone, A. Napoli, and M.-D. Devignes. Role assertion analysis : a proposed method for ontology refinement through assertion learning. In Amedeo Cesta and Nikos Fakotakis, editors, *STAIRS, Proceedings of the Fourth Starting AI Researchers' Symposium*, volume 179 of *Frontiers in Artificial Intelligence and Applications*, pages 47–58. IOS Press, 2008.

[34] A. Coulet, M. Smail-Tabbone, A. Napoli, and M.D. Devignes. Ontology refinement through role assertion analysis : Example in pharmacogenomics. In Franz Baader, Carsten Lutz, and Boris Motik, editors, *Description Logics*, volume 353 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2008.

[35] Frithjof Dau and Julia Klinger. From formal concept analysis to contextual logic. In Bernhard Ganter, Gerd Stumme, and Rudolf Wille, editors, *Formal Concept Analysis*, volume 3626 of *Lecture Notes in Computer Science*, pages 81–100. Springer, 2005.

[36] F. Distel. *Learning Description Logic Knowledge Bases from Data Using Methods from Formal Concept Analysis*. PhD thesis, TU Dresden, Germany, 2011.

[37] Y. Djouadi, D. Dubois, and H. Prade. Possibility theory and formal concept analysis : Context decomposition and uncertainty handling. In *Computational Intelligence for Knowledge-Based Systems Design*, volume 6178 of *LNCS-LNAI*, pages 260–269. Springer, 2010.

[38] Y. Djouadi, D. Dubois, and H. Prade. Graduality, uncertainty and typicality in formal concept analysis. In C. Cornelis et al., editor, *35 Years of Fuzzy Set Theory - Celebratory Volume Dedicated to the Retirement of Etienne E. Kerre*, pages 127–147. Springer, 2011.

[39] Y. Djouadi and H. Prade. Possibility-theoretic extension of derivation operators in formal concept analysis over fuzzy lattices. *Fuzzy Optimization and Decision Making*, 10(1–4) :287–309, 2011.

[40] Yassine Djouadi. Extended galois derivation operators for information retrieval based on fuzzy formal concept lattice. In *Proceedings of the 5th International Conference on Scalable Uncertainty Management*, SUM'11, pages 346–358, Berlin, Heidelberg, 2011. Springer-Verlag.

[41] Yassine Djouadi. Généralisation des opérateurs de dérivation de galois en recherche d'information basée sur l'analyse formelle de concepts. In *CORIA*, pages 373–386, 2012.

[42] Yassine Djouadi, Didier Dubois, and Henri Prade. Différentes extensions floues de l'analyse formelle de concepts / Different fuzzy extensions of formal concept analysis (regular paper). In *Rencontres Francophones sur la Logique Floue et*

ses Applications (LFA), Annecy (France), 05/11/2009-06/11/2009, pages 141–148. Cépaduès Editions, 2009.

[43] Yassine Djouadi, Didier Dubois, and Henri Prade. On the possible meanings of degrees when making formal concept analysis fuzzy (regular paper). In Pedro Burillo, Humberto Bustince, Bernard De Baets, and János Fodor, editors, *EUROFUSE Workshop Preference Modelling and Decision Analysis, Pampelune (Espagne), 16/09/2009-18/09/2009*, pages 253–258, http ://www.unavarra.es/, 2009. Universidad Pública de Navarra.

[44] Yassine Djouadi and Henri Prade. Interval-valued fuzzy formal concept analysis. In *ISMIS '09 : Proc. of the 18th International Symposium on Foundations of Intelligent Systems*, pages 592–601, Berlin, Heidelberg, 2009. Springer-Verlag.

[45] F. M. Donini, M. Lenzerini, D. Nardi, and A. Schaerf. Reasoning in description logics, 1997.

[46] William F. Dowling and Jean H. Gallier. Linear-time algorithms for testing the satisfiability of propositional horn formulae. *J. Log. Program.*, 1(3) :267–284, 1984.

[47] D. Dubois, F. Dupin de Saint-Cyr, and H. Prade. A possibility-theoretic view of formal concept analysis. *Fundamenta. Informaticae*, 75(1–4) :195–213, 2007.

[48] D. Dubois and H. Prade. *Fuzzy sets and systems - Theory and applications*. Academic press, New York, 1980.

[49] D. Dubois and H. Prade. *Possibility Theory*. Plenum Press, New York, N.Y., 1988.

[50] D. Dubois and H. Prade. An overview of the asymmetric bipolar representation of positive and negative information in possibility theory. *Fuzzy Sets and Systems*, 160(10) :1355–1366, 2009.

[51] D. Dubois and H. Prade. Possibility theory and formal concept analysis in information systems. In *In : Proc. IFSA'09, International Fuzzy Systems Association World Congress, Lisbon, Portugal*, pages 1021–1026, 2009.

[52] D. Dubois and H. Prade. Formal concept analysis from the standpoint of possibility theory. In *Formal Concept Analysis - 13th International Conference, ICFCA 2015, Nerja, Spain, June 23-26, 2015, Proceedings*, volume 9113 of *LNCS*, pages 21–38. Springer, 2015.

[53] Didier Dubois and Henri Prade. *Théorie des Possibilités. Applications à la Représentation des Connaissances en Informatique*. Collection Méthode + Programmes, Masson, Paris, 2 edition, 1985. Seconde édition revue et augmentée, Masson, Paris, 1987.

[54] Didier Dubois and Henri Prade. *Possibility Theory : An Approach to Computerized Processing of Uncertainty (traduction revue et augmentée de "Théorie des Possibilités").* Plenum Press, New York, 1988.

[55] Jon Ducrou and Peter W. Eklund. Searchsleuth : The conceptual neighbourhood of an web query. In *CLA*, volume 331 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2007.

[56] Düntsch, I., and G. Gediga. Modal-style Operators in Qualitative Data Analysis. In *Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM'02)*, pages 155–162, 2002.

[57] I. Düntsch and G. Gediga. Approximation operators in qualitative data analysis. In Harri de Swart, Ewa Orłowska, Gunther Schmidt, and Marc Roubens, editors, *Theory and Application of Relational Structures as Knowledge Instruments*, volume 2929 of *Lecture Notes in Computer Science*, pages 214–230. Springer–Verlag, Heidelberg, 2003.

[58] I. Düntsch and E. Orlowska. Mixing modal and sufficiency operators. *Bulletin of the Section of Logic, Polish Academy of Sciences*, 28(2) :99–106, 1999.

[59] I. Düntsch and E. Orlowska. Beyond modalities : sufficiency and mixed algebras. In A Szalas E. Orlowska, editor, *Relational Methods for Computer Sciences Applications*, pages 263–285. Springer, 2001.

[60] Justin Fagnan, Reihaneh Rabbany, Mansoureh Takaffoli, Eric Verbeek, and Osmar R Zaïane. Community dynamics : event and role analysis in social network analysis. In *International Conference on Advanced Data Mining and Applications*, pages 85–97. Springer International Publishing, 2014.

[61] S. Ferré. *Systèmes d'information logiques : un paradigme logico-contextuel pour interroger, naviguer et apprendre*. PhD thesis, Université de Rennes 1, October 2002. Accessible en ligne à l'adresse http ://www.irisa.fr/bibli/publi/theses/theses02.html.

[62] S. Ferre, O. Ridoux, and B. Sigonneau. Arbitrary relations in formal concept analysis and logical information systems. In *Conceptual Structures : Common Semantics for Sharing Knowledge, 13th International Conference on Conceptual Structures, ICCS 2005, Kassel, Germany, July 17-22, 2005, Proceedings*, pages 166–180, 2005.

[63] B. Ganter. Two basic algorithms in concept analysis. FB4–Preprint 831, TH Darmstadt, 1984.

[64] B. Ganter. Attribute exploration with background knowledge. *Theoretical Computer Science (TCS)*, 217(2) :215–233, 1999.

[65] B. Ganter and K. Reuter. Finding all closed sets : A general approach. In *Order, 8(3)*, pages 283–290, 1991.

[66] B. Ganter and R. Wille. *Formal Concept Analysis : Mathematical Foundations*. Springer-Verlag, Berlin, Heidelberg, 1999.

[67] Bernhard Ganter and Sergei O. Kuznetsov. Pattern structures and their projections. In *Proceedings of the 9th International Conference on Conceptual Structures : Broadening the Base*, ICCS '01, pages 129–142, London, UK, UK, 2001. Springer-Verlag.

[68] G. Gargov, S. Passy, and T. Tinchev. Modal environment for Boolean speculation. In D. Skordev, editor, *Mathematical Logic and Applications*, pages 253–263. Plenum Press, New-York, N.Y., 1987.

[69] Alain Gély. *A Generic Algorithm for Generating Closed Sets of a Binary Relation*, pages 223–234. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005.

[70] George Georgescu and Andrei Popescu. Non-dual fuzzy connections. *Arch. Math. Log.*, 43(8) :1009–1039, 2004.

[71] Robert Godin, Rokia Missaoui, and Alain April. Experimental comparison of navigation in a galois lattice with conventional information retrieval methods. *International Journal of Man-Machine Studies*, 38(5) :747 – 767, 1993.

[72] A. Guenoche. Construction du treillis de galois d'une relation binaire. In *Mathmatiques, Informatique et Sciences Humaines, 28ème année(109)*, pages 41–53, 1990.

[73] J. L. Guigues and V. Duquenne. Familles minimales d'implications informatives résultant d'un tableau de données binaires. *Mathématiques et Sciences Humaines*, 95 :5–18, 1986.

[74] A. Jaoua. Conceptual structured browsing : Applications for information retrieval, document summarization and meta-search engine design. In *5th International Conference ICFCA*. Clermont-Ferrand, France, 2007.

[75] B. Jónsson and A. Tarski. Boolean algebras with operators i. *American Journal of Mathematics*, 73 :891–939, 1951.

[76] Bjoern Koester. *Conceptual Knowledge Retrieval with FooCA : Improving Web Search Engine Results with Contexts and Concept Hierarchies*, pages 176–190. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[77] S. Kuznetsov. A fast algorithm for computing all intersections of objects in a finite semi-lattice. In *Automatic Documentation and Mathematical Linguistics*, pages 11–21, 1993.

[78] S. O. Kuznetsov and S. A. Obiedkov. Comparing Performance of Algorithms for Generating Concept Lattices. In *Journal of Experimental and Theoretical Artificial Intelligence*, pages 189–216, 2002.

[79] Sergei O. Kuznetsov and Sergei Obiedkov. Counting pseudo-intents and #p-completeness. In *Proceedings of the 4th International Conference on Formal Concept Analysis*, ICFCA'06, pages 306–308, Berlin, Heidelberg, 2006. Springer-Verlag.

[80] C. C. Latiri, S. Elloumi, J.P. Chevallet, and A. Jaoua. Extension of fuzzy galois connection for information retrieval using a fuzzy quantifier, 2003.

[81] Fritz Lehmann and Rudolf Wille. A triadic approach to formal concept analysis. In *Proceedings of the Third International Conference on Conceptual Structures : Applications, Implementation and Theory*, ICCS '95, pages 32–43, London, UK, UK, 1995. Springer-Verlag.

[82] M. Luxemburger. Implications partielles dans un contexte. *Mathématiques et Sciences Humaines*, 113 :35–55, 1991.

[83] M.-H. Masson. *Apports de la théorie des possibilités et des fonctions de croyance à l'analyse de données imprécises*. Hdr, UTC, December 2005.

[84] Jesús Medina, Manuel Ojeda-Aciego, and Jorge Ruiz-Calviòo. Formal concept analysis via multi-adjoint concept lattices. *Fuzzy Sets Syst.*, 160(2) :130–144, January 2009.

[85] R. Missaoui, L. Nourine, and Y. Renaud. Generating positive and negative exact rules using formal concept analysis : Problems and solutions. In *Formal Concept Analysis, 6th International Conference, ICFCA 2008, Montreal, Canada, February 25-28, 2008, Proceedings*, pages 169–181, 2008.

[86] R. Missaoui, L. Nourine, and Yoan Renaud. Computing implications with negation from a formal context. *Fundam. Inform.*, 115(4) :357–375, 2012.

[87] Emmanuel Nauer and Yannick Toussaint. Dynamical modification of context for an iterative and interactive information retrieval process on the web. In *Proceedings of the Fifth International Conference on Concept Lattices and Their Applications, CLA 2007, Montpellier, France, October 24-26, 2007*, 2007.

[88] Bernhard Nebel. Reasoning and revision in hybrid representation systems. In *of Lecture Notes in Artificial Intelligence*. Springer-Verlag, 1990.

[89] E. M. Nguifo and P. Njiwoua. Using Lattice-based Feamework as a Tool for Feature Extraction. In *Kluwer Academic Publishers*, pages 205–216, 1998.

[90] Engelbert Mephu Nguifo and Patrick Njiwoua. Treillis de concepts et classification supervisée. 24(4) :449–488, 2005.

[91] V.A. Niskanen. Introduction to imprecise reasoning, uncertainty, decision making and knowledge engineering. volume 1. Publications of the Society for Artificial Intelligence in Finland, 1989.

[92] E. M. Norris. An algorithm for computing the maximal rectangles in a binary relation. In *Revue Roumaine de Mathématiques Pures et Appliquées, 23(2)*, pages 243–250, 1978.

[93] L Nourine and O Raynaud. A fast algorithm for building lattices. *Information Processing Letters*, 71 :199–204, 1999.

[94] S. A. Obiedkov. Modal logic for evaluating formulas in incomplete contexts. In G. Angelova U. Priss, D. Corbett, editor, *Conceptual Structures : Integration and Interfaces*, volume 2393 of *LNCS*, pages 314–325. Springer, 2002.

[95] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. Efficient mining of association rules using closed itemset lattices. *Inf. Syst.*, 24(1) :25–46, 1999.

[96] S. Pollandt. Fuzzy begriffe. In *Springer-Verlag, Berlin/ Heidelberg*, 1997.

[97] S. Prediger. Terminologische merkmalslogik in der formalen begriffsanalyse. In Gerd Stumme and Rudolf Wille, editors, *Begriffliche Wissensverarbeitung : Methoden und Anwendungen*, Berlin–Heidelberg, 2000. Springer–Verlag.

[98] S. Prediger and G. Stumme. Theory-driven logical scaling : Conceptual information systems meet description logics. In E. Franconi and M. Kifer, editors, *Proc. 6th Int. Workshop Knowledge Representation meets Databases (KRDB'99)*, volume 21 of *CEUR Workshop Proceedings*, pages 46–49. CEUR-WS.org, 1999.

[99] Guilin Qi, Jeff Z. Pan, and Qiu Ji. A possibilistic extension of description logics. In Diego Calvanese, Enrico Franconi, Volker Haarslev, Domenico Lembo, Boris Motik, Anni-Yasmin Turhan, and Sergio Tessaris, editors, *Description Logics*, volume 250 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2007.

[100] V. Vychodil R. Belohlavek. What is a fuzzy concept lattice. In *Proc CLA'05, Olomouc Czech republic*, pages 34–45, 2005.

[101] M. H. Rouane, M. Huchard, A. Napoli, and P. Valtchev. A proposal for combining formal concept analysis and description logics for mining relational data. In Sergei O. Kuznetsov and Stefan Schmidt, editors, *"Proceedings of the 2nd International Conference on Formal Concept Analysis (ICFCA)"*, volume 4390 of *Lecture Notes in Computer Science*, pages 51–65. Springer, 2007.

[102] S. Rudolph. Exploring relational structures via FLE. In *Conceptual Structures at Work : 12th International Conference on Conceptual Structures. Volume 3127 of LNCS*. Springer, 2004.

[103] S. Rudolph. *Relational exploration : Combining Description Logics and Formal Concept Analysis for knowledge specification*. PhD thesis, FakultÄat Mathematik und Naturwissenschaften, TU Dresden, Germany, 2006.

[104] S. Rudolph. Acquiring generalized domain-range restrictions. In *Proceedings of the 6th International Conference on Formal Concept Analysis (ICFCA)*, volume 4933 of *LNAI*, pages 32–45. Springer, 2008.

[105] Sebastian Rudolph. An FCA method for the extensional exploration of relational data. In Bernhard Ganter and Aldo de Moor, editors, *Using Conceptual Structures, Contributions to ICCS 2003*, pages 197 – 210. Shaker, Aachen, July 2003.

[106] D. Dubois H. Prade S. Benfarhat, T. Denoeux. Représentations de l'incertitude en intelligence artificielle. *in : Pierre Marquis, O. Papini, H. Prade (Eds.),Panorama de l'Intelligence Artificielle, Cépaduès Editions, 3 2014, pp. 65-121*, volume (1).

[107] Ferroudja Seddoud. Construction du treillis de concepts formels pour des contextes à intervalle de vérité à partir d'implications résiduées. *mémoire de magister*, 2012.

[108] B. Sertkaya. *Formal Concept Analysis Methods for Descriptions Logics*. PhD thesis, Dresden university, 2008.

[109] Philippe Smets. *Imperfect Information : Imprecision and Uncertainty*, pages 225–254. Springer US, Boston, MA, 1997.

[110] Gerd Stumme and Alexander Maedche. Fca-merge : Bottom-up merging of ontologies. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume 1*, IJCAI'01, pages 225–230, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.

[111] Mansoureh Takaffoli, Reihaneh Rabbany, and Osmar R. Zaïane. Community evolution prediction in dynamic social networks. In *2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2014, Beijing, China, August 17-20, 2014*, pages 9–16, 2014.

[112] Rudolf Wille. Restructuring lattice theory : an approach based on hierarchies of concepts. In Ivan Rival, editor, *Ordered sets*, pages 445–470, Dordrecht–Boston, 1982. Reidel.

[113] Rudolf Wille. Concept lattices and conceptual knowledge systems. *Computers and Mathematics with Applications*, 23(6) :493 – 515, 1992.

[114] Rudolf Wille. Restructuring mathematical logic : an approach based on peirce's pragmatism. In Aldo Ursini and Paolo Agliano, editors, *Logic and algebra*, pages 267–281, New York, 1996. Marcel Dekker.

[115] William A. Woods and James G. Schmolze. The kl-one family. *Computers et Mathematics with Applications*, 23(2) :133 – 177, 1992.

[116] Sadok Ben Yahia and Ali Jaoua. Data mining and computational intelligence. chapter Discovering Knowledge from Fuzzy Concept Lattice, pages 167–190. Physica-Verlag GmbH, 2001.

[117] Yiyu Yao and Yaohua Chen. Rough set approximations in formal concept analysis. In James F. Peters and Andrzej Skowron, editors, *Transactions on Rough Sets V*, volume 4100 of *Lecture Notes in Computer Science*, pages 285–305. Springer, 2006.

[118] L. A. Zadeh. Fuzzy sets as the basis for a theory of possibility. *Fuzzy Sets and Systems*, 1 :3–28, 1978.

[119] L.A. Zadeh. Fuzzy sets. *Information and Control*, 8(3) :338 – 353, 1965.

[120] M. Zickwolff. *Rule Exploration : First Order Logic in Formal Concept Analysis*. PhD thesis, Technische Hochschule Darmstadt, 1991.

ANNEXE A

Liste des publications

Conférences nationales :

- Z. Ait-Yakoub and Y. Djouadi. Generating GCIs axioms from objects descriptions in EL-description logics. In Modeling Approaches and Algorithms for Advanced Computer Applications (CIIA 2013), volume 488 of Studies in Computational Intelligence, pages 75-84. Springer International Publishing, 2013.

Conférences internationales :

- Z. Ait-Yakoub, Y. Djouadi, D. Dubois, and H. Prade. From a possibility theory view of formal concept analysis to the possibilistic handling of incomplete and uncertain contexts. In CEUR Proc. Workshop FCA for AI, The Hague, 2016.
- Zina Ait-Yakoub, Yassine Djouadi, Didier Dubois, and Henri Prade. Concepts formels incomplets ou incertains : vers une généralisation possibiliste de l'analyse formelle de concepts (regular paper). In Rencontres Francophones sur la Logique Floue et ses Applications (LFA), La Rochelle, 02/11/2016-04/11/2016, pages 203-210, http ://www.cepadues.com/, 2016. Cépaduès Editions.

Articles de revue :

- Zina Ait-Yakoub, Yassine Djouadi, Didier Dubois, and Henri Prade. Asymmetric composition of possibilistic operators in formal concept analysis. application to the extraction of attribute implications from incomplete contexts. *International Journal of Intelligent Systems*, 32(12) :1285-1311, 2017.

Résumé

La théorie de l'analyse de concepts formels (ACF) dépend classiquement de l'utilisation de l'opérateur de dérivation de Galois. Les similitudes formelles entre la théorie des possibilités et l'analyse de concepts formels ont conduit à l'utilisation d'opérateurs possibilistes en ACF, qui ont été ignorés auparavant. Dans cette thèse, une approche basée sur l'utilisation de la composition asymétrique des deux opérateurs possibilistes les plus habituels est proposée.

Elle permet de compléter la base de Duquinne-Guigues, en dérivant les implications d'attributs avec les disjonctions des deux côtés des implications. Outre, L'approche est également généralisée à des contextes incomplets et incertains impliquant des informations positives et négatives. Nous appliquons ces résultats pour compléter une TBox, d'une description logique, avec des GCIs disjonctives.

Mots clés : analyse de concepts formels, opérateurs possibilistes, logiques de description, représentations incomplètes.

Abstract

Formal concept analysis theory (FCA) classically relies on the use of the Galois powerset operator. Formal similarities between possibility theory and formal concept analysis have led to the use of possibilistic operators in FCA, which were ignored before. In this thesis, an approach based on the use of asymmetric composition of the two most usual possibilistic operators is proposed.

It enables us to complement the stem base, by deriving attribute implications with disjunctions on both sides of the implications. Besides, the approach is also generalized to incomplete contexts involving explicit positive and negative information. We outline the potential application of these results to the completion of TBoxes in description logic.

Keywords: formal concepts analysis, possibilistic operators, description logics, incomplete knowledges