



INRIA Grenoble - Rhône-Alpes  
Equipe Exmo  
Innovallée  
655 avenue de l'Europe  
Montbonnot Saint-Martin  
38334 Saint Ismier cedex



Université Pierre Mendès France  
Master IC2A  
151 rue des universités  
38400 Saint Martin d'Hères

Rapport de stage  
M2P Ingénierie de la Communication Personne-Système

# Evolution d'alignements d'ontologies

**Maxime LEFRANÇOIS**

Master Ingénierie de la Cognition, de la Création et des Apprentissages  
M2P Ingénierie de la Communication Personne-Système  
Université Pierre Mendès France

Grenoble, Juillet 2010

Tuteurs de stage :

Jérôme E , Directeur de recherche à l'INRIA.  
Directeur de l'équipe Exmo.

Jérôme D , Maître de Conférence à l'Université Pierre Mendès France.  
Membre permanent de l'équipe Exmo.

# Contents

<b>Introduction</b>	<b>3</b>
<b>1 Des ontologies en réseau</b>	<b>5</b>
1.1 Ontologies en logique de description . . . . .	5
1.1.1 Structures logiques de Tarski . . . . .	5
1.1.2 Logiques de description, et ontologies en logiques de description . . . . .	6
1.2 Alignements entre ontologies . . . . .	9
1.2.1 Définition formelle des alignements . . . . .	9
1.2.2 Utilisation d'une algèbre de relations . . . . .	10
1.3 Réseaux d'ontologies . . . . .	13
1.3.1 Définition, normalisation, union . . . . .	13
1.3.2 Sémantique des réseaux d'ontologies . . . . .	14
<b>2 Evolution des alignements à partir d'un journal de versionnage d'ontologie</b>	<b>17</b>
2.1 Evolution et versionnage d'ontologie . . . . .	17
2.1.1 Evolution manuelle et semi-automatisée d'ontologie . . . . .	17
2.1.2 OWL2 Change Ontology . . . . .	18
2.1.3 PromptDiff . . . . .	18
2.2 Evolution d'alignement d'ontologies à partir d'une table PromptDiff . . . . .	19
2.2.1 Algorithme naïf d'évolution d'alignement d'ontologies . . . . .	19
2.2.2 Inconsistances et construction d'un nouvel algorithme . . . . .	20
2.3 Limites de la méthode, ouvertures . . . . .	21
2.3.1 Inconsistances et consolidation d'un réseau d'ontologies alignées . . . . .	21
2.3.2 Une relation supprimée peut encore être inférée . . . . .	22
2.3.3 Problème des concepts et rôles composites . . . . .	22
2.3.4 Gestion des indices de confiance . . . . .	23
2.3.5 Poursuite de la recherche des correspondances entre les ontologies . . . . .	23
<b>3 Méthodes d'évolution des réseaux d'ontologies basées sur la théorie de la révision</b>	<b>24</b>
3.1 Introduction à la théorie de la révision appliquée aux réseaux d'ontologies . . . . .	24
3.1.1 Opérateurs d'évolution d'une ontologie . . . . .	24
3.1.2 Extension aux réseaux d'ontologies . . . . .	25
3.1.3 Impact d'une évolution locale sur le réseau d'ontologie . . . . .	26
3.2 Etudes préliminaires de satisfaisabilité des postulats . . . . .	27
3.2.1 Solution drastique de redressement des postulats . . . . .	28
3.2.2 Incompatibilité des postulats de rétablissement et de succès pour la contraction . . . . .	28
3.3 Evolution des alignements du réseau pour le redressement des postulats . . . . .	30
3.3.1 Hypothèse supplémentaire . . . . .	30
3.3.2 Choix d'un critère de minimalité . . . . .	30
3.3.3 Reformulation du problème basée sur l'algèbre des relations . . . . .	31
3.3.4 Evaluation d'un algorithme simple : complexité et coût de l'opération . . . . .	31
<b>Conclusion</b>	<b>32</b>
<b>Bibliographie</b>	<b>34</b>

# Remerciements

Ce rapport est le produit de mon stage de Master dans l'équipe EXMO (Echanges de connaissance structurée médiatisés par ordinateur) de l'Inria (Institut National de Recherche en Informatique et en Automatique) Grenoble, Rhône-Alpes.

Je souhaite remercier mes superviseurs Jérôme E et Jérôme D pour leur accueil au sein de l'équipe, leur disponibilité, leur encadrement, et toutes les discussions éclairées. Leur aide m'a été précieuse pendant toute la durée de ce stage.

Toute ma reconnaissance va également aux membres de l'équipe EXMO et de l'équipe WAM que j'ai côtoyé pendant ces 5 mois, leur sympathie a rendu ce stage très agréable : Cassia T D S , François S , Marcelo S , Sinan Y , Yu Y , Melisachew W C , Everardo B P and Manuel A A - .

Enfin, un grand merci à ma famille, et à mes proches pour leur soutien.

# Introduction

## Problématique

Encore aujourd’hui, les documents qui composent le World Wide Web sont majoritairement composés d’informations interprétables uniquement par des personnes. Outre quelques métadonnées décrivant globalement le document, les seules informations interprétables par l’ordinateur concernent le rendu visuel des éléments d’interface. On peut dire que les machines peuvent “lire” et “afficher” le World Wide Web actuel, mais pas le “comprendre”.

A la fin des années 1990, Tim Berners Lee a partagé sa vision d’un Web Sémantique [2, 3]. D’après lui, l’ajout d’informations sémantiques dans les pages web constituerait une première étape vers l’interprétation automatique des contenus des pages web. L’ensemble des connaissances ainsi distribuées et connectées permettrait alors le développement de nouveaux moteurs de recherche et de nouveaux services web plus sophistiqués. En conséquence, la simplicité de navigation et la productivité des utilisateurs serait accrue. Ce développement de la communication personne-système est devenu un sujet de recherche très actifs dans le domaine de l’Intelligence Artificielle. Les ontologies font partie des outils employés pour réaliser ce développement.

### De quoi parle-t-on ?

Notre travail est essentiellement basé sur les ontologies, qui sont des outils de représentation de connaissances utilisés en informatique, pour le Web Sémantique, et les sciences de l’information. Une ontologie :

- décrit le vocabulaire conceptuel d’un domaine ;
- est un modèle formel structuré de données représentant le sens d’un champ d’informations du monde réel ;
- *est une spécification d’une conceptualisation partagée d’un domaine* (formule de Grüber, 1993 [13] souvent reprise dans la littérature).

Une ontologie est constituée d’éléments (i.e., des concepts ; des rôles ; des instances), ainsi que de formules qui expriment des relations entre ces éléments. Elle permet de raisonner dans le domaine concerné.

### Pourquoi aligner les ontologies ?

Quelques initiatives très controversées ont eu l’ambition de développer une ontologie globale – une base de connaissance générale et universelle –, dans le but de permettre à des applications d’intelligence artificielle de raisonner d’une manière similaire à l’être humain. En pratique, différents groupes de travail de contextes variés développent et utilisent des ontologies différentes. Les différents contextes, la pluralité des représentations du monde, les divergences d’opinion, peuvent justifier ce phénomène. Le Web sémantique est par nature une application pour laquelle la distribution des acteurs et des connaissances rendent impossible l’exigence d’une ontologie unique.

Dès lors que plusieurs ontologies co-existent et couvrent partiellement les mêmes domaines, il devient nécessaire de trouver des correspondances entre des éléments de ces ontologies. Trouver des correspondances est une opération nommée alignement d’ontologies [9]. Cette opération consiste à prendre deux ontologies en entrée, et à en sortir un ensemble de correspondances entre des éléments de ces ontologies. Les alignements peuvent alors être utilisés pour importer des données d’une ontologie à une autre, ou pour traduire des requêtes. L’ensemble des ontologies et des alignements qui les interconnectent forme un réseau d’ontologies, que l’on peut également utiliser pour raisonner.

## Pourquoi faire évoluer ces alignements ?

Obtenir un alignement de qualité est une tâche difficile et coûteuse. Par ailleurs, une ontologie est un objet dynamique qui peut être amené à changer en réponse à des besoins. Un changements dans une ontologie peut alors ruiner les efforts investis pour obtenir ces alignements.

Ainsi, la question principale sous-jacent ce travail est : comment peut-on faire évoluer l'alignement entre deux ontologies lorsque l'une d'entre elles évolue ? [8]. Notre étude rend compte des problèmes soulevés par cet objectif et apporte quelques éléments de réponse.

## Solutions envisagées & organisation du rapport

L'objectif de ce travail est de développer de nouveaux résultats à la jonction de domaines de recherche différents :

- les ontologies ;
- les alignements d'ontologies ;
- les algèbres de relations ;
- les réseaux d'ontologies alignées ;
- l'évolution d'ontologies ;
- le versionnage d'ontologies ;
- le changement de croyances.

En conséquence, l'état de l'art dans ces domaines et notre contribution ne feront pas l'objet de chapitres distincts. Au contraire, chaque partie comportera une description de ce qui se fait dans la recherche, ainsi que des propositions spécifiques à notre étude.

La première partie est centrée sur les objets principaux de ce travail. Nous y rappellerons un ensemble de définitions relatives aux ontologies des logiques de description et aux alignements entre ontologies. Nous nous inspirerons ensuite de travaux de la recherche pour proposer :

- 1- une algèbre de relations sur laquelle on basera les alignements ;
- 2- une définition formelle des réseaux d'ontologies avec leur syntaxes et leur sémantique.

Pour faire évoluer les alignements, deux approches sont a priori possibles. Elles feront l'objet des parties 2 et 3 :

**Approche syntaxique – modifications** L'approche syntaxique est basée sur la structure des ontologies. Nous identifierons les modifications possibles dans une ontologie, et on déterminera les conséquences que ces modifications peuvent avoir sur les alignements. Nous pourrons alors considérer les modifications possible de l'alignement en fonction de celles qui peuvent survenir dans une ontologie.

Concrètement, nous développerons un algorithme d'évolution des alignements à partir d'un journal de versionnage d'ontologie obtenue par l'outil PromptDiff [18], développé en 2002. Nous évaluerons ensuite les limites inhérente à cette méthode, celles qui sont liées à l'outils de versionnage choisi, et nous concluerons sur les perspectives de la méthode.

**Approche sémantique – postulats** L'approche sémantique est basée sur la sémantique des ontologies et des alignements. On s'inspirera des travaux qui ont été menés dès le milieu des années 1980 dans le domaine des changements de croyances pour proposer une caractérisation d'opérateurs d'évolution des réseaux d'ontologies. Les postulats de rationalité que doivent satisfaire des opérateurs de révision ou de contraction dans une ontologie de logique de description y sont d'abord rappelés, puis on propose une extension aux réseaux d'ontologies.

On identifiera ensuite l'impact d'une évolution locale sur le réseau d'ontologie. On cherchera alors quelles opérateurs d'évolution peuvent être appliqués au réseau pour palier les éventuels problèmes. Ces opérateurs devront modifier l'alignement de manière minimale vis-à-vis des conséquence que cela devrait avoir sur l'interprétation des ontologies alignées [23].

# Chapter 1

## Des ontologies en réseau

Dans ce premier chapitre nous introduirons les trois outils de représentation de connaissances qui constituent les objets d'étude de notre travail :

- les ontologies ;
- les alignements entre ontologies ;
- les réseaux d'ontologies alignées.

Nous définirons leur syntaxe et leur sémantique, puis nous présenterons une algèbre de relations sur laquelle nous proposerons de baser les alignements.

### 1.1 Ontologies en logique de description

Différentes logiques peuvent être utilisées pour la représentation de connaissances, dont la logique du premier ordre particulièrement bien connue. On s'intéressera à un autre type de logique commune en représentation des connaissances et sur laquelle notre travail repose : les logiques de description. Nous introduirons ensuite les ontologies en logique de description en définissant leur syntaxe et leur sémantique.

Avant cela, il est nécessaire de répondre à la question suivante : *Qu'est-ce qui définit une logique ?*

#### 1.1.1 Structures logiques de Tarski

Les logiques sont basées sur une structure très générale introduite par Jan Lukasiewicz et Alfred Tarski à la fin des années 1920. Une *structure logique* est définie comme une *paire*  $\langle \mathcal{L}, Cn \rangle$ , où  $\mathcal{L}$  représente le langage (i.e., les propositions exprimables dans la logique), et  $Cn$  est un *opérateur de conséquences* défini par un certain nombre de *règles* propres à chaque logique.  $Cn$  associe à chaque ensemble de propositions  $X$  un autre ensemble de propositions  $Cn(X)$ , appelé ensemble des conséquences de  $X$ , ou encore clôture de  $X$ .  $Cn(X)$  est obtenu en appliquant en boucle – éventuellement infinie – les règles de la logique sur les propositions de  $X$ .

Tarski a proposé trois axiomes qui contraignent l'opérateur de conséquences, de sorte qu'il ait un comportement rationnel :

$$\begin{aligned} \forall X \subseteq \mathcal{L}, X \subseteq Cn(X) & \quad (\textit{inclusion}) \\ \forall X \subseteq \mathcal{L}, Cn(Cn(X)) = Cn(X) & \quad (\textit{itération}) \\ \forall X, Y \subseteq \mathcal{L}^2, X \subseteq Y \Rightarrow Cn(X) \subseteq Cn(Y) & \quad (\textit{monotonie}) \end{aligned}$$

De manière équivalente à l'opérateur de conséquence  $Cn$ , On peut définir une *relation d'inférence* entre les ensembles de propositions :

$$X \vdash Y \Leftrightarrow Y \subseteq Cn(X)$$

Ainsi, à partir des *croyances* (i.e., un ensemble de propositions sous-ensemble de  $\mathcal{L}$ ), On peut déduire d'autres propositions par un mécanisme d'inférence. On peut montrer que la relation d'inférence satisfait les propriétés suivantes :

$$\begin{aligned} \forall X, Y \subseteq \mathcal{L}^2, X \subseteq Y \Rightarrow Y \vdash X & \quad (\textit{reflexivité}) \\ \forall X, Y, Z \subseteq \mathcal{L}^3, X \vdash Y \text{ et } X \cup Y \vdash Z \Rightarrow X \vdash Z & \quad (\textit{transitivité}) \\ \forall X, Y, Z \subseteq \mathcal{L}^3, X \vdash Y \Rightarrow X \cup Z \vdash Y & \quad (\textit{affaiblissement}) \end{aligned}$$

### 1.1.2 Logiques de description, et ontologies en logiques de description

La syntaxe d'un langage de représentation des connaissances est caractérisée par un ensemble de *symboles* et par des *règles de construction* qui permettent d'obtenir des formules à partir de ces symboles et moyennant des *constructeurs* spécifiques au langage.

Les *logiques de description* (*Description Logics*, ou *DLs*) forment une famille de langages de représentation de connaissance qui peuvent être utilisés pour représenter la connaissance terminologique d'un domaine d'application d'une façon structurée et formelle. Ces langages ont été introduits dans les années 80 dans le but de rendre la représentation de connaissances plus naturelle qu'en logique du premier ordre. Ils ont néanmoins une sémantique formelle conforme au cadre des logiques de Tarski. Dans ce sens, on peut dire que les DLs ont une *sémantique descriptive formelle*.

Nous verrons dans un premier temps la syntaxe d'une logique de description et d'une ontologie en logique de description, puis nous verrons quelle sémantique on peut leur attribuer, enfin nous décrivons très succinctement les langages de représentation des ontologies.

#### Syntaxe d'une logique de description

Une logiques de description  $\mathcal{LD}$  divise la connaissance en deux sous-ensembles :

- **T-Box – terminology box, les informations terminologiques** : un ensemble de formules relatives aux informations sur les notions basiques ou dérivées et sur la façon dont elles sont liées entre elles. Ces informations sont *génériques* ou *globales*, vraies dans tous les modèles et pour tous les individus.
- **A-Box – assertions box, les informations sur les individus** : un ensemble de formules relatives aux informations *spécifiques* ou *locales*, vraies pour certains individus particuliers.

Une logique de description  $\mathcal{LD}$  possède une *signature*, qui est basée sur les types *concepts*, *rôles* et *individus*. Le langage d'une logique de description autorise également l'utilisation de *constructeurs* pour créer des éléments complexes, ainsi que des *axiomes* pour déclarer des *formules* qui peupleront la T-Box et la A-Box.

**Definition 1** (Concept). *Un concept d'une logique de description  $\mathcal{LD}$  correspond à une classe d'éléments et est interprété comme un ensemble dans un univers donné. Les concepts peuvent être atomiques (correspondant à un symbole de type concept), ou complexes obtenus en utilisant les constructeurs de concepts autorisés par le langage de  $\mathcal{LD}$ .*

**Definition 2** (Rôle). *Un rôle d'une logique de description  $\mathcal{LD}$  correspond à un lien entre deux éléments et est interprété comme une relation binaire sur un univers donné. Les rôles peuvent être atomiques (correspondant à un symbole de type rôle), ou complexes obtenus en utilisant les constructeurs de rôles autorisés par le langage de  $\mathcal{LD}$ .*

**Definition 3** (Individu). *Un individu d'une logique de description  $\mathcal{LD}$  correspond à un élément d'un univers donné.*

**Definition 4** (Signature). *Soit  $\mathcal{LD}$  une logique de description,  $C = \{C_1, C_2, \dots\}$  un ensemble fini de concepts atomiques,  $R = \{R_1, R_2, \dots\}$  un ensemble fini de rôles atomiques et  $\mathcal{A} = \{a_1, a_2, \dots\}$  un ensemble fini d'individus. Pour  $C, R, \mathcal{A}$  disjoints deux à deux,  $\Sigma = \langle C, R, \mathcal{A} \rangle$  est une signature de  $\mathcal{LD}$ .*

**Definition 5** (Formule). *Soit  $\mathcal{LD}$  une logique de description, Une formule de  $\mathcal{LD}$  pour une signature  $\Sigma$  est l'un des axiomes autorisés par le langage de  $\mathcal{LD}$ .*

On peut introduire la notion d'*ontologie d'une logique de description* à partir de ces définitions :

**Definition 6** (Ontologie). *Soit  $\mathcal{LD}$  une logique de description, une ontologie de  $\mathcal{LD}$  est une paire  $\langle \Sigma, F \rangle$  tel que  $\Sigma$  est une signature et  $F = \langle T, A \rangle$  contient des axiomes  $T$  de la T-Box, et des assertions  $A$  de la A-Box.*

**Definition 7** (Élément d'une ontologie). *Un élément d'une ontologie  $o = \langle \Sigma, F \rangle$  de  $\mathcal{LD}$  est:*

- soit une entité atomique de l'ontologie (i.e., un élément de la signature  $\Sigma$ )
- soit une entité complexe construite en utilisant des constructeurs du langage de l'ontologie.

Dans la suite, on utilisera la notation  $Q_{\mathcal{LD}}(o)$  pour désigner l'ensemble des éléments d'une ontologie  $o$  de  $\mathcal{LD}$ .

Le tableau 1.1 donne en exemple les syntaxes de chaque terme atomique, ainsi que différents constructeurs de concept, constructeurs de rôle, axiomes de la T-Box et axiomes de la A-Box. Le nom de la logique minimale qui les autorise est noté dans la colonne de droite. Une logique de description est un langage qui utilise une partie de ces constructeurs. On nomme une logique de description en prenant comme base la logique simple ( $\mathcal{AL}$ ) et en lui accolant le nom des constructeurs supplémentaires que l'on autorise, par exemple  $\mathcal{ALCHI}^{\top, \perp}$ . Des logiques utilisant certaines combinaisons de constructeurs ont été bien étudiées et sont nommées différemment comme  $\mathcal{EL}$ , ou DL-Lite.

Terme atomique	Syntaxe	Sémantique	
individu	$a$	$I(a) \in \Delta$	$(\mathcal{AL})$
concept atomique	$A$	$I(A) \subseteq \Delta$	$(\mathcal{AL})$
rôle atomique	$R$	$I(R) \subseteq \Delta \times \Delta$	$(\mathcal{AL})$
Constructeur de concepts	Syntaxe	Sémantique	nom
concept universel	$\top$	$I(\top) = \Delta$	$(\mathcal{AL})$
concept vide	$\perp$	$I(\perp) = \emptyset$	$(\mathcal{AL})$
conjonction	$C \sqcap D$	$I(C \sqcap D) = I(C) \cap I(D)$	$(\mathcal{AL})$
disjonction	$C \sqcup D$	$I(C \sqcup D) = I(C) \cup I(D)$	$\mathcal{U}$
négation	$\neg C$	$\Delta \setminus I(C)$	$\mathcal{C}$
restriction existentielle	$\exists R. C$	$\{x \mid \exists y, \langle x, y \rangle \in I(R) \wedge y \in I(C)\}$	$(\mathcal{AL})$
restriction de valeur	$\forall R. C$	$\{x \mid \forall y, \langle x, y \rangle \in I(R) \Rightarrow y \in I(C)\}$	$(\mathcal{AL})$
restriction numérique	$\leq nR$	$\{x \mid \text{card}(\{y, \langle x, y \rangle \in I(R)\}) \leq n\}$	$\mathcal{N}$
restriction numérique	$\geq nR$	$\{x \mid \text{card}(\{y, \langle x, y \rangle \in I(R)\}) \geq n\}$	$\mathcal{N}$
nominaux	$\{a_1, \dots, a_n\}$	$\{I(a_1), \dots, I(a_n)\}$	$\mathcal{O}$
Constructeur de rôles	Syntaxe	Sémantique	nom
conjonction de rôles	$R \sqcap S$	$I(R \sqcap S) = I(R) \cap I(S)$	$\cdot \sqcap$
disjonction de rôles	$R \sqcup S$	$I(R \sqcup S) = I(R) \cup I(S)$	$\cdot \sqcup$
complément de rôle	$\neg R$	$(\Delta \times \Delta) \setminus I(R)$	$\cdot \neg$
clôture transitive	$R^+$	<i>clôture transitive de <math>I(R)</math></i>	$\cdot +$
rôle inverse	$R^-$	$\{\langle y, x \mid \langle x, y \rangle \in I(R)\}$	$\cdot -$
composition de rôles	$R \circ S$	$\{\langle x, z \mid \exists y, \langle x, y \rangle \in I(R) \wedge \langle y, z \rangle \in I(S)\}$	$\cdot \circ$
Axiomes de la T-Box	Syntaxe	Contrainte d'interprétation	nom
subsumption	$C \sqsubseteq D$	$I(C) \subseteq I(D)$	$(\mathcal{AL})$
inclusion de rôles	$R \sqsubseteq S$	$I(R) \subseteq I(S)$	$\mathcal{H}$
transitivité de rôles	$\text{Trans}(R)$	$I(R) = I(R^+)$	$\mathcal{S}$
Axiomes de la A-Box	Syntaxe	Contrainte d'interprétation	nom
appartenance à un concept	$C(a)$	$I(a) \in I(C)$	$\mathcal{AL}$
appartenance à un rôle	$R(a_1, a_2)$	$\langle I(a_1), I(a_2) \rangle \in I(R)$	$\mathcal{AL}$
identité	$a_1 = a_2$	$I(a_1) = I(a_2)$	

Table 1.1: Syntaxe et sémantique de différents constructeurs et connecteurs des logiques de description.



## Sémantique des logiques de description

Maintenant que nous avons défini la syntaxe des logiques de description, nous pouvons présenter leur sémantique suivant la *théorie des modèles*.

**Definition 8** (Interprétation). Soient  $\mathcal{LD}$  une logique de description et  $\Sigma = \langle C, \mathcal{R}, \mathcal{A} \rangle$  une signature de  $\mathcal{LD}$ . Une interprétation  $\mathcal{I}$  de  $\Sigma$  est une paire  $\langle \Delta, \mathcal{I} \rangle$ , où le domaine d'interprétation  $\Delta$  est un ensemble non-vide, et  $\mathcal{I}$  est un triplet de fonctions :  $I_C : C \rightarrow 2^\Delta$ ,  $I_{\mathcal{R}} : \mathcal{R} \rightarrow 2^{\Delta \times \Delta}$ , et  $I_{\mathcal{A}} : \mathcal{A} \rightarrow \Delta$ .

**Definition 9** (Relation de satisfaction). Soit  $\mathcal{LD}$  une logique de description. A chaque axiome autorisé par le langage de  $\mathcal{LD}$ , est associé une contrainte d'interprétation. On dit qu'une interprétation  $\mathcal{I}$  satisfait une formule  $f$  si et seulement si la contrainte d'interprétation associée à l'axiome de  $f$  est satisfaite. On note cette relation  $\mathcal{I} \models f$ .

Le tableau 1.1 donne en exemple les relations de satisfaction de chaque terme atomique, constructeur de concept, constructeur de rôle, axiomes de la T-Box et axiomes de la A-Box.

Nous pouvons désormais définir la notion très importante de *modèle* :

**Definition 10** (Modèle). Soient  $\mathcal{LD}$  une logique de description et  $o = \langle \Sigma, F \rangle$  une ontologie de  $\mathcal{LD}$ . Une interprétation de  $\Sigma$  est un modèle de  $o$  si et seulement si elle satisfait l'ensemble des formules de  $F$ . On note cette relation  $\mathcal{I} \models o$ . On note également  $\mathcal{M}(o)$  l'ensemble des modèles de  $o$ .

Les deux notions essentielles de raisonnement logique sont la consistance d'une ontologie et l'opérateur de conséquence sémantique (ou de manière équivalente, une relation d'inférence).

**Definition 11** (Consistance). Soit  $\mathcal{LD}$  une logique de description, une ontologie  $o$  de  $\mathcal{LD}$  est consistante si et seulement si il existe au moins un modèle de  $o$ .

Une autre notion, celle de cohérence, est parfois présente dans la littérature, mais elle est associée à des sens différents. A priori, une ontologie sera dite incohérente si elle ne répond pas à certains critères que l'on peut associer à des règles de bon usage. Ces règles peuvent évidemment varier d'un groupe d'ingénieurs d'ontologie à un autre. Un critère classique consiste à dire qu'une ontologie est incohérente si une de ses classes est nécessairement vide d'individus, sous peine de rendre l'ontologie inconsistante.

A partir de la notion de modèle nous pouvons définir les notions de *conséquences* et de *clôture*, propres aux structures logiques de Tarski, ainsi que des notions relatives au *raisonnement* dans les ontologies.

**Definition 12** (Conséquence sémantique, clôture). Soit  $\mathcal{LD}$  une logique de description,  $o$  une ontologie de  $\mathcal{LD}$  et  $f$  une formule de  $\mathcal{LD}$ . On dit que  $f$  est une conséquence sémantique de  $o$  lorsque tous les modèles de  $o$  satisfont  $f$ . Dans ce cas on note  $o \models f$ . L'ensemble des conséquences sémantiques de l'ontologie  $o$  est la clôture de  $o$  et est noté  $Cn(o)$ .

**Definition 13** (Décidabilité). Soit  $\mathcal{LD}$  une logique de description, on dit que  $\mathcal{LD}$  est décidable si et seulement si pour toute ontologie  $o$  de  $\mathcal{LD}$  et pour toute formule  $f$  de  $\mathcal{LD}$ , il existe un algorithme permettant de déterminer si  $f$  est une conséquence sémantique de  $o$  ou non.

**Definition 14** (Correction et complétude). Soit  $\mathcal{LD}$  une logique de description,  $o$  une ontologie de  $\mathcal{LD}$  et  $f$  une formule de  $\mathcal{LD}$ . Un algorithme  $A$  tel que  $A(o, f) \in \{VRAI, FAUX\}$  est dit :

- correct si et seulement si pour toute ontologie  $o$  et pour toute formule  $f$ ,  $A(o, f) = VRAI \Rightarrow o \models f$  ;
- complet si et seulement si pour toute ontologie  $o$  et pour toute formule  $f$ ,  $A(o, f) = VRAI \Leftarrow o \models f$  ;

## Langages de représentation des ontologies

Les ontologies sont exprimées à l'aide de langages formels tels que RDF, RDF Schema, et OWL 2, pour ne citer que les standards du World Wide Web Consortium (W3C).

Le Web Ontology Language (ou OWL 2) recommandé comme standard par le W3C, est basé sur les fondements théoriques des logiques de description. En ce sens, ce standard propose plusieurs sous-langages, chaque sous-langage correspondant à une logique de description bien connue, avec une certaine expressivité et une certaine complexité. Pour une application particulière, le choix du sous-langage à utiliser est conditionné par le rapport d'importance auquel on s'accorde entre l'expressivité du langage, et la complexité des calculs nécessaires pour raisonner sur l'ontologie. A la limite, le sous-langage qui autorise l'ensemble du langage proposé par OWL 2 a une complexité telle qu'il est indécidable, à l'instar de la logique du premier ordre.

## 1.2 Alignements entre ontologies

Comme nous l'avons vu, différentes ontologies peuvent co-exister et couvrir partiellement les mêmes domaines, ce qui génère la nécessité de trouver des correspondances – des relations sémantiques – entre des éléments de ces ontologies. Un alignement d'ontologies est une description explicite de ces relations sémantiques. Nous suivons ici la définition standard apportée dans [9]. Puis nous nous inspirerons plus particulièrement d'un article publié par Jérôme Euzenat en 2008 [7], dans lequel il est proposé d'utiliser une algèbre de relations pour décrire les relations sémantiques qui peuvent exister entre différents éléments de différentes ontologies.

### 1.2.1 Définition formelle des alignements

**Définition 15** (Relation entre éléments d'ontologies). *Une relation entre éléments d'ontologies est un symbole qui exprime une relation binaire entre deux éléments d'ontologies. L'ensemble des relations d'alignement autorisées dépend du langage d'alignement utilisé. On notera cet ensemble  $\Theta$ .*

A priori, différentes relations peuvent être utilisées : l'équivalence, la subsumption, la disjonction de classes ; des relations temporelles/spatiales ; des relations floues ; etc. Les algorithmes d'alignement utilisent pour la plupart seulement la relation d'équivalence.

**Définition 16** (Correspondance). *Soient  $o$  et  $o'$  deux ontologies de langages de description respectifs  $\mathcal{LD}$  et  $\mathcal{LD}'$ . Soient  $Q_{\mathcal{LD}}(o)$  et  $Q_{\mathcal{LD}'}(o')$  leurs ensembles d'éléments respectifs,  $\Theta$  un ensemble de relations d'alignement, et  $\Xi$  un ensemble fini de valeurs de confiance. Une correspondance est un quadruplet :*

$$\langle e, e', r, n \rangle$$

tel que

- $e \in Q_{\mathcal{LD}}(o)$  et  $e' \in Q_{\mathcal{LD}'}(o')$  sont les deux éléments d'ontologies mis en correspondance ;
- $r \in \Theta$  est la relation qui lie  $e$  à  $e'$  ;
- $n \in \Xi$  est un indice de confiance pour la correspondance.

La correspondance  $\langle e, e', r, n \rangle$  déclare que les éléments d'ontologies  $e$  et  $e'$  sont liés par la relation  $r$  avec un degré de confiance  $n$ . Nous noterons occasionnellement  $e r e'$ .

**Définition 17** (Alignement). *Soient  $o$  et  $o'$  deux ontologies de langages de description respectifs  $\mathcal{LD}$  et  $\mathcal{LD}'$ . Un alignement entre  $o$  et  $o'$  est un ensemble de correspondances entre des paires d'éléments d'ontologies appartenant à  $Q_{\mathcal{LD}}(o)$  et  $Q_{\mathcal{LD}'}(o')$  respectivement.*

L'indice de confiance  $n$  a été introduit pour des raisons pragmatiques. Lors de l'application d'un algorithme d'alignement automatique d'ontologies, cet indice permet de donner une mesure (qualitative) de la confiance que l'on peut avoir dans la correspondance trouvée. Meilicke et al. dans [17] proposent que la fonction qui assigne à chaque correspondance trouvée un indice de confiance  $n$  soit injective. Du coup l'indice de confiance permet de définir une relation d'ordre total sur l'ensemble des correspondances de l'alignement. Après avoir défini la fusion de deux ontologies grâce à l'alignement qui les lie, et après avoir montré que le produit de cette fusion peut être inconsistant, les auteurs utilisent cette relation d'ordre pour rétablir la consistance en supprimant une correspondance. Pour cela, la méthode est de trouver l'ensemble inconsistant minimal de formules (qui contient nécessairement des correspondances de l'alignement) et de supprimer la correspondance de cet ensemble qui présente le plus faible indice de confiance. Une autre utilisation de l'indice de confiance est décrite dans [6] et appliquée dans [16]. Il s'agit lors de l'application d'un algorithme d'alignement d'ontologies, d'utiliser deux seuils de confiance : les correspondances dont l'indice de confiance est au delà du seuil le plus élevé sont considérées comme vraies, celles dont l'indice de confiance est entre les deux seuils sont filtrées en prenant en compte la structure de l'ontologie (la hiérarchie des concepts), enfin les correspondances dont l'indice de confiance est en dessous du seuil bas sont rejetées.

## 1.2.2 Utilisation d'une algèbre de relations

L'utilisation d'une algèbre des relation pour le raisonnement logique a été introduite par Allen en 1983 avec l'algèbre des intervalles, qui permet de faire du raisonnement temporel à propos d'évènements duratifs (des intervalles de temps). Cette algèbre a été reprise et étendue par exemple pour le raisonnement des relations de positionnement spatiale par Egenhofer et Sharma en 1992. Euzenat dans [7] a proposé de fonder les alignements sur l'algèbre des relations. L'auteur y définit le comportement des opérateurs d'union, d'intersection, d'inversion et de composition des correspondances. Les principaux résultats de ce stage sont basés sur cette algèbre de relations, et cette sous-section est donc l'occasion de rappeler cette algèbre et ses opérateurs. Pour plus d'informations, le lecteur peut se référer à [7, 21].

### Introduction aux algèbres des relations

**Definition 18** (Algèbre de Boole). *Soit l'algèbre déterminée sur la classe  $\mathcal{P}(\mathcal{E}) = 2^{\mathcal{E}}$  de l'ensemble des parties d'un ensemble non vide génératif  $\mathcal{E}$ , ou tout du moins sur un sous-ensemble non vide  $\Theta$  de cette classe, fermé par rapport à la disjonction  $\cup$ , à la conjonction  $\cap$ , et la complémentation  $\bar{\cdot}$ , et qui contient l'ensemble vide  $\emptyset$  et  $\mathcal{E}$  lui même. On dit que l'ensemble  $\Theta$  forme une algèbre de Boole et on note cette structure  $\langle \Theta, \cup, \cap, \bar{\cdot}, \emptyset, \mathcal{E} \rangle$ . Toute structure isomorphe à une telle structure est également une algèbre de Boole.*

**Definition 19** (Algèbre de relations). *Une algèbre de relations est une structure  $\langle \Theta, \cup, \cap, \bar{\cdot}, \emptyset, \mathcal{E}, *, ^{-1}, 1' \rangle$  tel que :*

- $\langle \Theta, \cup, \cap, \bar{\cdot}, \emptyset, \mathcal{E} \rangle$  est une algèbre de Boole ;
- $*$  est une loi binaire associative de composition interne à  $\Theta$ , d'élément neutre  $1'$ , et distributive pour  $\cup$  ;
- $^{-1}$  est une opérateur unaire d'inversion interne à  $\Theta$  et involutif, qui est distributif pour  $\cup$ ,  $\cap$ , et  $*$ .

Nous utiliserons simplement l'ensemble des parties d'un ensemble génératif  $\mathcal{E}$  comme ensemble des relations d'alignement autorisées. Soit donc l'algèbre de relations définie par la structure  $\langle 2^{\mathcal{E}}, \cup, \cap, \bar{\cdot}, \emptyset, \mathcal{E}, \cdot, ^{-1}, \{=\} \rangle$  telle que :

- $\mathcal{E}$  est un ensemble de relations entre deux entités conjointement exhaustives et disjointes deux à deux, i.e., quelle que soit la situation, la relation véritable entre deux entités est une et une seule relation de base de l'ensemble  $\mathcal{E}$  ;
- Les relations prises dans l'ensemble  $2^{\mathcal{E}}$  permettent d'exprimer une incertitude, i.e.,  $xRy = \bigvee_{r \in R} x\{r\}y$ . Ainsi, la relation décrite par l'ensemble complet  $\mathcal{E}$  signifie : "on ne sait pas", car cette relation est satisfaite par n'importe quelle paire d'entités ;
- $\cup$  est l'opérateur d'intersection d'ensemble pour les relations, e.g., si on a  $xRy$  ou  $xR'y$ , alors on a  $x(R \cup R')y$ .  $\mathcal{E}$  est un élément absorbant pour  $\cup$  tel que :  $\forall R \in 2^{\mathcal{E}}, R \cup \mathcal{E} = \mathcal{E} \cup R = \mathcal{E}$  ;
- $\cap$  est l'opérateur d'union d'ensemble pour les relations. e.g., si on a  $xRy$  et  $xR'y$ , alors on a  $x(R \cap R')y$ .  $\emptyset$  est un élément absorbant pour  $\cap$  tel que :  $\forall R \in 2^{\mathcal{E}}, R \cap \emptyset = \emptyset \cap R = \emptyset$  ;
- $\bar{\cdot}$  est l'opérateur de complémentation tel que  $\overline{\overline{R}} = R$ ,  $R \cup \overline{R} = \mathcal{E}$ , et  $R \cap \overline{R} = \emptyset$ .  $\overline{R}$  est définie par  $\overline{R} = \{r \in \mathcal{E} | r \notin R\}$ , et les règles de De Morgan s'appliquent  $\overline{R \cup R'} = \overline{R} \cap \overline{R'}$  et  $\overline{R \cap R'} = \overline{R} \cup \overline{R'}$  ;
- $\cdot$  est la loi binaire associative de composition interne, telle que si on a  $xRy$  et  $yR'z$ , alors on a  $x(R \cdot R')z$ . La relation atomique  $\{=\}$  est l'élément neutre pour cette loi. Ainsi on a  $\forall R \in 2^{\mathcal{E}}, R \cdot \{=\} = \{=\} \cdot R = R$  ;
- $^{-1}$  est un opérateur d'inversion tel que  $\forall R \in 2^{\mathcal{E}}, xRy \Leftrightarrow yR^{-1}x$ . On a la propriété  $(R^{-1})^{-1} = R$  ;
- Chacun des opérateurs décrits ci-dessus est appliqué sur les ensembles des relations de base en les distribuant sur chaque élément, e.g.,  $R \cdot R' = \bigcup_{r \in R, r' \in R'} r \cdot r'$ .

**Application aux relations des correspondances**

Pour manier les relations entre des éléments des ontologies, nous utiliserons un ensemble de cinq relations de base conjointement exhaustives et disjointes deux à deux, qui sont : est-équivalent = ; inclut > ; est-inclut < ; chevauche  $\emptyset$  ; est-disjoint  $\perp$ . L'ensemble  $\mathcal{E} = \{=, >, <, \emptyset, \perp\}$  génère un ensemble  $2^{\mathcal{E}}$  de  $2^5 = 32$  relations, parmi lesquelles se trouvent la relation  $\mathcal{E}$  qui signifie "on ne sait pas", et la relation  $\emptyset$  qui exprime l'inconsistance. La figure 1.1 présente le treillis de génération des parties de l'ensemble  $\mathcal{E}$ . Dans ce treillis, l'union de deux relations est la plus basse des relations parentes, et l'intersection de deux relations est la plus haute des relations enfants. Par souci de place, le treillis est représenté avec l'élément bas à gauche, et l'élément haut à droite.

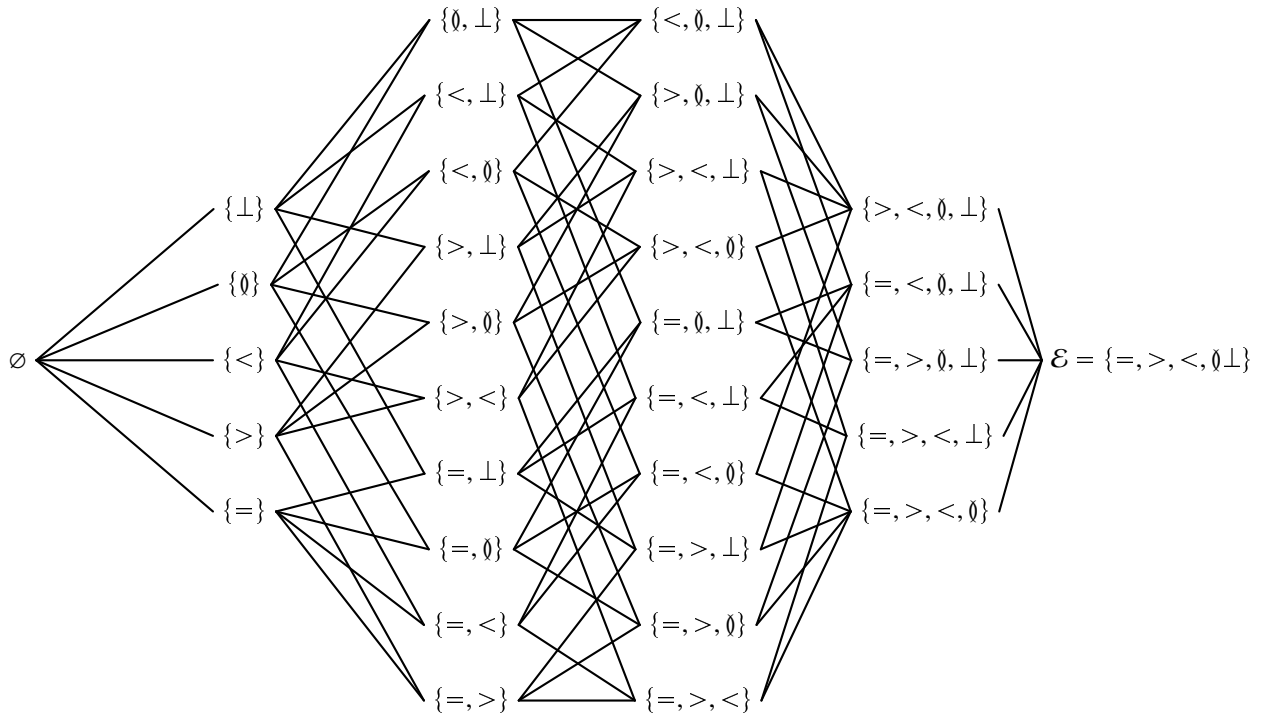


Figure 1.1: Le treillis des 32 relations de l'algèbre de relations.

Chaque relation élémentaire est sa propre inverse, sauf  $<^{-1} = >$  et  $>^{-1} = <$ .

Le tableau 1.2 représente la table de composition de l'opérateur  $\cdot$  pour les concepts atomiques. Comme il est dit plus haut, l'opérateur de composition est appliqué sur les ensembles des relations de base en le distribuant sur chaque élément, e.g.,  $R \cdot R' = \bigcup_{r \in R, r' \in R'} r \cdot r'$ .

	=	>	<	$\emptyset$	$\perp$
=	=	>	<	$\emptyset$	$\perp$
>	>	>	$>< = \emptyset$	$>\emptyset$	$>\emptyset\perp$
<	<	$\mathcal{E}$	<	$<\emptyset\perp$	$\perp$
$\emptyset$	$\emptyset$	$>\emptyset\perp$	$<\emptyset$	$\mathcal{E}$	$>\emptyset\perp$
$\perp$	$\perp$	$\perp$	$<\emptyset\perp$	$<\emptyset\perp$	$\mathcal{E}$

Table 1.2: Table de composition pour l'algèbre de relations.

Dans l'algèbre des intervalles de Allen, si deux intervalles sont dans un état relationnel donné, et si on déplace seulement une borne d'un intervalle, on peut montrer qu'il est parfois impossible de passer directement d'un certain état relationnel à un autre. Pour l'algèbre que l'on utilise, Nous avons représenté cette contrainte par un graphe de proximité des relations élémentaires, c.f., figure 1.2. A l'aide de ce graphe de proximité, nous définirons ce qu'est une relation convexe, ainsi qu'une notion de distance entre deux relations, lorsque l'une est incluse dans l'autre.

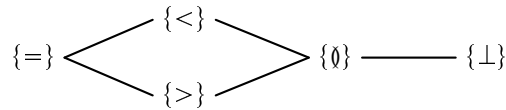


Figure 1.2: Le graphe de proximité des relations élémentaires de l'algèbre de relations.

**Definition 20** (Relation convexe). *Une relation  $R$  de l'algèbre des relations est dite convexe si et seulement si le sous-graphe qui lui est associé dans le graphe de proximité des relations est connexe.*

**Definition 21** (Distance entre deux relations lorsque l'une est incluse dans l'autre). *Soient  $R$  et  $R'$  deux relations telles que  $R \subseteq R'$ , la distance entre  $R$  et  $R'$  est le cardinal de l'ensemble  $\{r \in R' \mid r \notin R\}$ , i.e., le nombre de relations de  $R'$  qui ne sont pas dans  $R$ .*

**Exemple 1** (Relation de conséquence directe dans l'algèbre des relations). *Au niveau des correspondances, de part la structure de l'algèbre de relations, on peut définir une relation de conséquence directe :*

*Une relation  $R'$  est une conséquence directe d'une autre relation  $R$  si et seulement si  $R \subseteq R'$  i.e., si et seulement si chaque relation élémentaire de  $R$  est incluse dans  $R'$ . On note cette relation de conséquence  $R \vdash R'$ .*

### Opérations sur les alignements

Maintenant que nous avons défini la structure de l'algèbre des relations des correspondances, nous pouvons définir formellement les opérations sur les alignements. Admettons que deux ontologies  $o$  et  $o'$  de logiques de description respectives  $\mathcal{LD}$  et  $\mathcal{LD}'$  soient alignées par  $A$ . Un sous-ensemble de  $\mathcal{Q}_{\mathcal{LD}}(o) \times \mathcal{Q}_{\mathcal{LD}'}(o')$  est mis en correspondance. De plus, plusieurs correspondances peuvent exister pour la même paire d'éléments d'ontologies. Une première étape pour simplifier les écritures des opérateurs d'alignements est donc de proposer une normalisation d'un alignement de sorte qu'une et une seule correspondance lie deux éléments d'ontologies. Pour cela nous utilisons un l'alignement "vide de correspondances"  $A^0$  tel que pour chaque paire d'éléments  $(e, e')$  pris dans  $\mathcal{Q}_{\mathcal{LD}}(o) \times \mathcal{Q}_{\mathcal{LD}'}(o')$  on ait une et une seule correspondance associée à la relation "on ne sait pas"  $\mathcal{E}$ . Formellement, on définit  $A^0$  tel que :

$$A^0 = \{\langle e, e', \mathcal{E} \rangle \mid e \in \mathcal{Q}_{\mathcal{LD}}(o), e' \in \mathcal{Q}_{\mathcal{LD}'}(o')\}$$

A partir d'un alignement  $A$ , on définit l'alignement normalisé  $A^*$  par :

$$A^* = \{\langle e, e', \bigcap_{\langle e, e', R \rangle \in A \cup A^0} R \rangle\}$$

Lorsqu'une nouvelle étude (manuelle ou automatisée) apporte des raisons de croire en de nouvelles correspondances entre deux ontologies, on utilise une opération de disjonction entre l'ancien alignement et le nouveau, traduisant une incertitude par rapport aux vrais relations des correspondances. L'opération qui doit être utilisée pour combiner les relations des correspondances est donc l'union :

$$A \vee A' = \{\langle e, e', R \cup R' \rangle \mid \langle e, e', R \rangle \in A^*, \langle e, e', R' \rangle \in A'^*\}$$

Au contraire, lorsqu'une nouvelle étude (manuelle ou automatisée) apporte des restrictions sur les correspondances entre deux ontologies, on utilise une opération de conjonction entre l'ancien alignement et le nouveau, traduisant une restriction des possibilités dans les relations. L'opération qui doit être utilisée pour combiner les relations des correspondances est donc l'intersection :

$$A \wedge A' = \{\langle e, e', R \cap R' \rangle \mid \langle e, e', R \rangle \in A^*, \langle e, e', R' \rangle \in A'^*\}$$

On peut également définir les opérations de complémentation et d'inversion d'un alignement :

$$\begin{aligned} \bar{A} &= \{\langle e, e', \bar{R} \rangle \mid \langle e, e', R \rangle \in A^*\} \\ A^{-1} &= \{\langle e, e', R^{-1} \rangle \mid \langle e, e', R \rangle \in A^*\} \end{aligned}$$

Ainsi que l'opération de composition.

$$A \cdot A' = \{\langle e, e'', R \cdot R' \rangle \mid \langle e, e', R \rangle \in A^*, \langle e', e'', R' \rangle \in A'^*\}$$

## 1.3 Réseaux d'ontologies

Un grand nombre d'ontologies co-existent aujourd'hui sur le Web. De plus, des alignements ont été calculés pour certains couples de ces ontologies. On est donc face à des structures en réseau. De tels réseaux peuvent prendre la forme de réseaux pair-à-pair sémantiques, de réseaux sociaux sémantiques, etc.

Il est ainsi possible d'associer les connaissances décrites par différents groupes de travail, de prendre en compte différents degrés de granularité de connaissances, et également de tirer parti de l'utilisation de différentes logiques de description.

Nous allons définir la notion de réseau d'ontologies, et la sémantique d'un réseau d'ontologies.

### 1.3.1 Définition, normalisation, union

**Definition 22** (Réseau d'ontologies). *Un réseau d'ontologies sur l'ensemble des index  $I$  est une paire  $\langle \Omega, \Lambda \rangle$  constituée d'une famille d'ontologies  $\Omega = (o_i)_{i \in I}$  interconnectées par une famille d'alignements  $\Lambda$ . on note  $\Lambda(o_i, o_j)$  l'ensemble des alignements entre  $o_i$  et  $o_j$ .*

Pour des raisons de désambiguïsation, nous préférons la désignation réseau d'ontologies introduite par Zimmermann [22] à système distribué originellement proposée par [12].

Par simplicité, nous manipulerons des réseaux d'ontologies que l'on dira normalisés, i.e., qui contiennent un et un seul alignement normalisé  $A_{ij}^*$  par paire d'ontologies  $o_i$  et  $o_j$ , vide de conséquences pour  $i \geq j$  :

**Definition 23** (Réseau d'ontologies normalisé). *Soit  $S = \langle \Omega, \Lambda \rangle$  un réseau d'ontologies sur l'ensemble des index  $I$ . le réseau d'ontologies normalisé  $S^* = \langle \Omega, \Lambda^* \rangle$  est tel que  $\Lambda^* = \{A_{ij}^*, (i, j) \in I^2\}$  avec :*

$$\begin{aligned} \forall i \geq j, A_{ij}^* &= A_{ij}^0 \\ \forall i < j, A_{ij}^* &= \{A_{ij}^0 \wedge \bigwedge_{A \in \Lambda(o_j, o_i)} A^{-1} \wedge \bigwedge_{A \in \Lambda(o_i, o_j)} A^*\} \end{aligned}$$

Dans la suite de ce rapport, nous ne considérerons que les réseaux d'ontologies normalisés, et nous utiliserons l'expression réseau d'ontologies pour faire référence à un réseau d'ontologies normalisé.

**Definition 24** (Union de deux réseaux d'ontologies). *On définit l'union de deux réseaux d'ontologies sur l'ensemble des index  $I$  par :*

$$\langle (o_i)_{i \in I}, (A_{ij})_{(i,j) \in I^2} \rangle \cup \langle (o'_i)_{i \in I}, (A'_{ij})_{(i,j) \in I^2} \rangle = \langle (o_i \cup o'_i)_{i \in I}, (A_{ij} \cup A'_{ij})_{(i,j) \in I^2} \rangle^*$$

*Cet opérateur est une loi de composition interne sur l'ensemble des réseaux d'ontologies indexées par  $I$ .*

**Definition 25** (Intersection de deux réseaux d'ontologies). *On définit l'intersection de deux réseaux normalisés d'ontologies sur l'ensemble des index  $I$  par :*

$$\langle (o_i)_{i \in I}, (A_{ij})_{(i,j) \in I^2} \rangle^* \cap \langle (o'_i)_{i \in I}, (A'_{ij})_{(i,j) \in I^2} \rangle^* = \langle (o_i \cap o'_i)_{i \in I}, (A_{ij} \cap A'_{ij})_{(i,j) \in I^2} \rangle^*$$

*Cet opérateur est une loi de composition interne sur l'ensemble des réseaux normalisés d'ontologies indexées par  $I$ .*

**Definition 26** (Inclusion d'un réseau d'ontologies dans un autre). *On définit l'inclusion d'un réseau d'ontologies sur l'ensemble des index  $I$  dans un autre réseau d'ontologies sur l'ensemble des index  $I$  par :*

$$\langle (o_i)_{i \in I}, (A_{ij})_{(i,j) \in I^2} \rangle \subseteq \langle (o'_i)_{i \in I}, (A'_{ij})_{(i,j) \in I^2} \rangle \Leftrightarrow \langle (o_i)_{i \in I}, (A_{ij})_{(i,j) \in I^2} \rangle \cup \langle (o'_i)_{i \in I}, (A'_{ij})_{(i,j) \in I^2} \rangle = \langle (o'_i)_{i \in I}, (A'_{ij})_{(i,j) \in I^2} \rangle^*$$

### 1.3.2 Sémantique des réseaux d'ontologies

En partant du constat que les domaines d'interprétation des ontologies sont a priori incompatible, Zimmermann et Euzenat [23] ont étudié les implications les avantages et les problèmes de trois sémantiques des systèmes distribués :

- La sémantique distribuée simple : considère l'ensemble du système distribué comme une base de connaissances cohérente, avec un seul domaine d'interprétation  $\Delta$ . Cette hypothèse permet de fusionner des ontologies de manière cohérente grâce aux correspondances. Néanmoins, cette hypothèse rend cette sémantique faible vis-à-vis des inconsistances, qu'elles soient locales ou globales. Cela n'est pas souhaitable dans un environnement hautement hétérogène comme le web sémantique.
- La sémantique distribuée contextuelle : est inspirée par des travaux sur la contextualisation de la logique du premier ordre (D-FOL) [11], des logiques de description (DDL) [4] et des ontologies (C-OWL) [5], le domaine d'interprétation  $\Delta_o$  de chaque ontologie  $o$  peut être différent, donc les domaines sont liés à l'aide de relations de domaine  $r_{o_1, o_2}$  qui relie les éléments de  $\Delta_{o_1}$  aux éléments de  $\Delta_{o_2}$ .
- La sémantique distribuée intégrée : est une version étendue de la sémantique distribuée simple. Chaque représentation locale des connaissances est interprétée dans un domaine local, mais la connaissance globale est interprétée dans un domaine d'interprétation universel  $\mathcal{U}$  "virtuel", qui peut être défini comme l'union de tous les domaines d'interprétation, et d'un ensemble de fonctions d'égalisation  $\gamma_o : \Delta_o \rightarrow \mathcal{U}$  pour lier chaque élément de chaque ontologie à un élément du domaine d'interprétation global.

C'est cette troisième sémantique que nous nous proposons de reprendre ici, elle nécessite de définir ce qu'est une fonction d'égalisation :

**Definition 27** (Fonction d'égalisation). *Soit une famille d'interprétations  $\mathcal{I}_{o \in \Omega} = \langle \Delta_o, \mathcal{I}_o \rangle_{o \in \Omega}$  d'un ensemble d'ontologies  $\Omega$ , une fonction d'égalisation pour  $\mathcal{I}_{o \in \Omega}$  est une famille de fonctions  $\gamma = (\gamma_o : \Delta_o \rightarrow \mathcal{U})_{o \in \Omega}$  d'un domaine d'interprétation vers le domaine global d'interprétation  $\mathcal{U}$ . L'ensemble des fonctions d'égalisation est noté  $\Gamma$ .*

Nous avons déjà défini formellement la sémantique des ontologies. Nous allons maintenant définir celle des alignements, puis celle des réseaux d'ontologies pour le modèle de sémantique distribuée intégrée.

## Sémantique des alignements

**Definition 28** (Interprétation des relations d'alignement). *Soit  $R$  une relation d'alignement et  $\mathcal{U}$  un domaine global d'interprétation, une fonction d'interprétation de  $R$  est une relation binaire sur  $\mathcal{U}$ , i.e.,  $R^{\mathcal{U}} \subseteq \mathcal{U} \times \mathcal{U}$ .*

La définition de la satisfaction d'une correspondance dépend de la fonction d'égalisation  $\gamma$  et de l'interprétation des relations. On cherche la condition pour que les correspondances soient satisfaites par les projections des modèles dans le domaine d'interprétation universel  $\mathcal{U}$ .

**Definition 29** (Satisfaction d'une correspondance). *Une correspondance  $c = \langle e, e', R \rangle$  est satisfaite pour une fonction d'égalisation par deux modèles  $(m, m') \in \mathcal{M}(o) \times \mathcal{M}(o')$  si et seulement si  $\gamma_o(m) \in \mathcal{M}(o)$ ,  $\gamma_{o'}(m') \in \mathcal{M}(o')$ , et*

$$(\gamma_o(m(e)), \gamma_{o'}(m'(e'))) \in R^{\mathcal{U}}$$

On note cette relation  $m, m' \models_{\gamma} c$ .

**Exemple 2** (Satisfaction des correspondances pour l'algèbre des relations).

$$\forall (m, m') \in \mathcal{M}(o) \times \mathcal{M}(o'), m, m' \models_{\gamma} \langle e, e', = \rangle \Leftrightarrow (\gamma_o(m), \gamma_{o'}(m')) \in \mathcal{M}(o) \times \mathcal{M}(o') \text{ et } \gamma_o(m(e)) = \gamma_{o'}(m'(e'))$$

$$\forall (m, m') \in \mathcal{M}(o) \times \mathcal{M}(o'), m, m' \models_{\gamma} \langle e, e', > \rangle \Leftrightarrow (\gamma_o(m), \gamma_{o'}(m')) \in \mathcal{M}(o) \times \mathcal{M}(o') \text{ et } \gamma_o(m(e)) \supset \gamma_{o'}(m'(e'))$$

$$\forall (m, m') \in \mathcal{M}(o) \times \mathcal{M}(o'), m, m' \models_{\gamma} \langle e, e', < \rangle \Leftrightarrow (\gamma_o(m), \gamma_{o'}(m')) \in \mathcal{M}(o) \times \mathcal{M}(o') \text{ et } \gamma_o(m(e)) \subset \gamma_{o'}(m'(e'))$$

$$\forall (m, m') \in \mathcal{M}(o) \times \mathcal{M}(o'), m, m' \models_{\gamma} \langle e, e', \emptyset \rangle \Leftrightarrow (\gamma_o(m), \gamma_{o'}(m')) \in \mathcal{M}(o) \times \mathcal{M}(o') \text{ et } \gamma_o(m(e)) \emptyset \gamma_{o'}(m'(e'))$$

$$\forall (m, m') \in \mathcal{M}(o) \times \mathcal{M}(o'), m, m' \models_{\gamma} \langle e, e', \perp \rangle \Leftrightarrow (\gamma_o(m), \gamma_{o'}(m')) \in \mathcal{M}(o) \times \mathcal{M}(o') \text{ et } \gamma_o(m(e)) \cap \gamma_{o'}(m'(e')) = \emptyset$$

**Definition 30** (Satisfaction d'un alignement). *Soit  $A$  un alignement entre deux ontologies  $o$  et  $o'$ ,  $m$  et  $m'$  deux modèles de  $o$  et  $o'$  pour leurs domaines d'interprétation respectifs, et  $\gamma$  une fonction d'égalisation. On dit que l'alignement  $A$  est satisfait par le triplet  $\langle m, m', \gamma \rangle$  si et seulement si ce triplet satisfait toutes les correspondances de  $A$ . On note cette relation  $m, m' \models_{\gamma} A$ .*

**Definition 31** (Alignement valide). *Un alignement  $A$  entre deux ontologies  $o$  et  $o'$  est dit valide si et seulement si*

$$\forall (m, m', \gamma) \in \mathcal{M}(o) \times \mathcal{M}(o') \times \Gamma, (\gamma(m), \gamma(m')) \in \mathcal{M}(o) \times \mathcal{M}(o') \Rightarrow m, m' \models_{\gamma} A$$

Les alignements valides n'apportent pas d'information additionnelles aux ontologies alignées. En effet, les alignements valides sont des conséquences directes des deux ontologies.

**Exemple 3** (Alignement valide pour l'algèbre des relations). *L'alignement vide  $A^0 = \{\langle e, e', \mathcal{E} \rangle \mid e \in \mathcal{Q}_{\mathcal{LD}}(o), e' \in \mathcal{Q}_{\mathcal{LD}}(o')\}$  entre deux ontologies  $o$  et  $o'$  est valide.*

**Definition 32** (Alignement satisfaisable). *Un alignement  $A$  entre deux ontologies  $o$  et  $o'$  est satisfaisable si et seulement si*

$$\exists (m, m', \gamma) \in \mathcal{M}(o) \times \mathcal{M}(o') \times \Gamma \mid m, m' \models_{\gamma} A$$

i.e., si il existe des modèles des ontologies qui peuvent être combinés de telle sorte que l'alignement ait un sens.

**Exemple 4** (Non-satisfaisabilité pour l'algèbre des relations). *On ne peut pas donner directement de critère pour la satisfaisabilité d'un alignement  $A$  basé sur l'algèbre des relations, mais on peut néanmoins affirmer que si un alignement normalisé a une correspondance vide (i.e.,  $\exists \langle e, e', R \rangle \in A^* \mid R = \emptyset$ ), alors il est insatisfaisable.*

**Definition 33** (Modèle d'ontologies alignées). *Soient deux ontologies  $o$  et  $o'$ , et un alignement  $A$  entre ces ontologies, un modèle de ces ontologies alignées est un triplet  $\langle m, m', \gamma \rangle \in \mathcal{M}(o) \times \mathcal{M}(o') \times \Gamma \mid m, m' \models_{\gamma} A$*

Ainsi, l'alignement agit comme un filtre sur les modèles des ontologies, en sélectionnant ceux qui lui sont compatibles. On procède ainsi à un transfert d'information d'une ontologie à une autre puisqu'une réduction de l'ensemble des modèles implique une augmentation des conséquences dans chacune des ontologies alignées. La simplicité de la sémantique distribuée intégrée nous permet d'étendre très simplement ces définitions aux réseaux d'ontologies.



### Sémantique des réseaux d'ontologies et raisonnement

**Definition 34** (Modèles d'un réseau d'ontologies). Soit  $\langle \Omega, \Lambda \rangle$  un réseau d'ontologies sur l'ensemble des index  $I$ , avec  $\Omega = (o_i)_{i \in I}$  une famille d'ontologies, et  $\Lambda = (A_{ij}^*)_{(i,j) \in I^2}$  une famille d'alignements normalisés. Un modèle pour  $\langle \Omega, \Lambda \rangle$  est un ensemble constitué d'une famille de modèles et d'une fonction d'égalisation :

$$\langle (m_i)_{i \in I}, \gamma \rangle \in \prod_{i \in I} \mathcal{M}(o_i) \times \Gamma \mid \forall (i, j) \in I^2, m_i, m_j \models_{\gamma} A_{ij}$$

L'ensemble des modèles du réseau d'ontologies sera noté  $\mathcal{M}(\langle \Omega, \Lambda \rangle)$ .

**Remarque.** Pour l'ensemble des réseaux d'ontologies normalisés que l'on étudie, les alignements  $A_{ij}$  tels que  $i \geq j$  sont valides car vides de correspondances, et sont donc toujours satisfaits quelqusoit la fonction d'égalisation  $\gamma$ .

De la même manière que pour deux ontologies alignées, l'ensemble des alignements  $\Lambda$  agit comme un filtre sur les modèles des ontologies, ce qui implique de nouvelles conséquences pour le réseau. Ces conséquences peuvent être de deux sortes :

- des correspondances dans les alignements, qui ont déjà été décrites dans la littérature et qui sont appelées  $\alpha$ -conséquences du réseau d'ontologies. Il s'agit des correspondances satisfaites par tous les modèles du réseau d'ontologies ;
- de nouvelles formules dans les ontologies. Bien que ces conséquences n'aient pas encore fait l'objet de publication, Jérôme Euzenat les a mises en évidence<sup>1</sup> et les a nommées  $\omega$ -conséquences du réseau d'ontologies. Il s'agit des formules associées à une ontologie qui sont satisfaites par chaque modèle de cette ontologie impliqué dans un modèle du réseau d'ontologies.

**Definition 35** ( $\alpha$ -conséquences d'un réseau d'ontologies). Soit  $\langle \Omega, \Lambda \rangle$  un réseau d'ontologies, une correspondance  $\mu$  entre deux ontologies  $o_i$  et  $o_j$  de  $\Omega$  est une  $\alpha$ -conséquence de  $\langle \Omega, \Lambda \rangle$  pour l'alignement  $A_{ij}$  (on note  $\langle \Omega, \Lambda \rangle \models \mu$ ) si et seulement si :

$$\forall \langle (m_i)_{i \in I}, \gamma \rangle \in \mathcal{M}(\langle \Omega, \Lambda \rangle), m_i, m_j \models_{\gamma} \mu$$

L'ensemble des  $\alpha$ -conséquences de  $\langle \Omega, \Lambda \rangle$  entre  $o_i$  et  $o_j$  est la clôture  $\alpha$  de  $\langle \Omega, \Lambda \rangle$  entre  $o_i$  et  $o_j$  et est noté  $Cn_{\Omega, \Lambda}^{\alpha}(o_i, o_j)$ .

**Definition 36** ( $\omega$ -conséquences d'un réseau d'ontologies). Soit  $\langle \Omega, \Lambda \rangle$  un réseau d'ontologies et  $\mathcal{LD}_i$  la logique de description d'une ontologie  $o_i$  de  $\Omega$ . Une formule  $f$  de  $\mathcal{LD}_i$  est une  $\omega$ -conséquence de  $\langle \Omega, \Lambda \rangle$  pour  $o_i$  (on note  $\langle \Omega, \Lambda \rangle \models f$ ) si et seulement si :

$$\forall \langle (m_i)_{i \in I}, \gamma \rangle \in \mathcal{M}(\langle \Omega, \Lambda \rangle), m_i \models f \text{ et } m_i \models_{\gamma} f$$

L'ensemble des  $\omega$ -conséquences de  $\langle \Omega, \Lambda \rangle$  pour  $o_i$  est la clôture  $\omega$  de  $\langle \Omega, \Lambda \rangle$  pour  $o_i$  et est noté  $Cn_{\Omega, \Lambda}^{\omega}(o_i)$ .

Ces définitions sont sémantiques et en pratique une définition syntaxique doit être donnée suivant les logiques de description des ontologies et l'algèbre de relations sur laquelle les alignements sont basés.

Nous pouvons finalement définir la clôture d'un réseau d'ontologies :

**Definition 37** (Clôture d'un réseau d'ontologies). Soit  $\langle \Omega, \Lambda \rangle$  un réseau d'ontologies sur l'ensemble des index  $I$ , la clôture de  $\langle \Omega, \Lambda \rangle$  est un réseau d'ontologies sur l'ensemble des index  $I$  noté  $Cn(\langle \Omega, \Lambda \rangle)$  tel que :

$$Cn(\langle \Omega, \Lambda \rangle) = \langle (Cn_{\Omega, \Lambda}^{\omega}(o_i))_{i \in I}, (Cn_{\Omega, \Lambda}^{\alpha}(o_i, o_j))_{(i,j) \in I^2} \rangle$$

i.e., Il s'agit d'un réseau d'ontologies formé de l'ensemble des clôtures  $\omega$  des ontologies, interconnectées par l'ensemble des clôtures  $\alpha$  des alignements.

<sup>1</sup>Jérôme Euzenat, Revision in networks of ontologies, communication personnelle

## Chapter 2

# Evolution des alignements à partir d'un journal de versionnage d'ontologie

Une ontologie peut être amenée à changer en réponse à des besoins. Les efforts fournis pour découvrir les alignements entre ontologies doivent être préservés autant que possible lorsqu'une des ontologies est amenée à évoluer. Dans ce chapitre nous proposerons une réflexion autour d'un outil de versionnage d'ontologie. Plus précisément, nous chercherons à utiliser un tel outil pour adapter au mieux un alignement à une évolution dans une ontologie. Nous présenterons dans un premier temps l'évolution et le versionnage d'ontologies. Nous proposerons ensuite une méthode pour adapter les alignements. Enfin, nous identifierons des limites d'une telle méthode.

## 2.1 Evolution et versionnage d'ontologie

Dans la littérature, une ontologie est souvent définie suivant cette formule de Gruber : *une ontologie est une spécification d'une conceptualisation partagée d'un domaine*. [13]. Une ontologie peut donc être amenée à changer en réponse à des besoins que l'on peut diviser en trois catégories :

- un changement dans le domaine : une ontologie modélise souvent un environnement dynamique.
- un changement dans la conceptualisation : permet de prendre en compte de nouvelles informations, un changement de point de vue, un nouvel usage, etc.
- un changement dans la spécification : lorsqu'il s'agit de traduire avec le minimum de pertes une ontologie d'un langage de représentation des connaissances à un autre, ou bien de traduire sa signature.

L'*évolution d'ontologie* est le domaine de l'ingénierie des ontologies qui englobe tout le processus d'adaptation d'une ontologie en réponse aux deux premiers de ces besoins. Le troisième besoin est l'objet d'un domaine voisin : la *traduction d'ontologies*. Nous donnerons un aperçu des méthodes utilisées usuellement pour faire évoluer les ontologies dans la première section de ce chapitre.

Le domaine du *versionnage d'ontologie* consiste à garder les traces des versions successives d'une ontologie. Cela permet par exemple d'assurer la validité, l'interopérabilité et la gestion de toutes les versions d'une ontologie, y compris la version actuelle. On présentera en particulier deux médiums de représentation des évolutions entre des versions successives d'une ontologie, qui permettent de faire du versionnage.

### 2.1.1 Evolution manuelle et semi-automatisée d'ontologie

Pour disséquer le problème de l'évolution d'une ontologie, les auteurs dans [20] ont défini un cycle en six phases qui se répète :

1. capture des changements : le besoin de changements est identifié et formalisé ;
2. évaluation de la représentation des changements : les changements à effectuer dans l'ontologie sont identifiés ;
3. évaluation de la sémantique des changements : les conséquences des changements sont identifiés, et les éventuels problèmes sont résolus ;
4. implémentation des changements : les changements sont concrétisés ;
5. propagation des changements : les systèmes dépendants de cette ontologie s'adaptent à la nouvelle évolution ;
6. validation des changements : permet aux ingénieurs d'évaluer les changements, et éventuellement de les annuler.

Durant la phase d'évaluation de la représentation des changements, on doit décider quels sont les changements, élémentaires ou composites, que l'on doit faire sur l'ontologie. Les auteurs de [20] ont identifié deux classes de changements :

- les changements élémentaires (ou atomiques), sont simples, de bas niveau, et très locaux ;
- les changements composites (ou complexes), sont de plus haut niveau et impliquent une plus grande partie de l'ontologie. Ils sont décomposables en changements élémentaires, mais remplacer un changement composite par une série de changements élémentaires risque d'engendrer des effets de bord.

La définition du répertoire de ces changements accessible dans un logiciel de gestion d'ontologie est un problème majeur de l'évolution d'ontologies. En effet le nombre de changements composites peut tendre à l'infini si on ne se limite pas à un certain niveau de granularité des changements [15].

Il semble que les méthodes proposées actuellement par les logiciels de gestion d'ontologies émulent les comportements des humains, en se basant sur des heuristiques qui reposent fortement sur l'expertise de leurs développeurs et des ingénieurs d'ontologies [10]. Elles n'ont pas de base théorique et requièrent des interventions d'ingénieurs pour fonctionner. Cela représente une limite importante, en particulier pour la gestion de grosses ontologies ou la gestion collaborative d'ontologies.

### 2.1.2 OWL2 Change Ontology

Une ontologie des changements dans les ontologies décrites en OWL2 a été proposée par Palma et al. dans [19]. Une évolution d'une ontologie serait une instance de cette ontologie des changements, qui devrait être produite par le logiciel d'édition d'ontologies utilisé pour modifier manuellement l'ontologie.

Une des critiques de l'on peut faire à cette méthode de représentation des changements est qu'elle propose de définir une classe par changement composite, et qu'on a vu qu'il peut exister une infinité de tels changements. De plus, à ma connaissance, aucun éditeur d'ontologies n'implémente de génération d'instances de l'ontologie des changements. Enfin, cette ontologie des changements ne prend pas en compte la possibilité que les changements soient menés automatiquement dans l'ontologie, comme nous le verrons dans le chapitre suivant.

### 2.1.3 PromptDiff

PromptDiff est un outil présenté par Noy et Musen dans [18] qui permet de générer un journal de versions entre deux ontologies. Le journal produit est similaire à un journal de versionnage de code de logiciels comme CVS. Cependant le problème est plus subtil pour les ontologies : par exemple, un changement de positions de deux déclarations de classes dans un fichier texte ne modifie pas la structure de l'ontologies. PromptDiff analyse donc les différences structurelle qui existent entre deux versions d'ontologies.

Pour cela, différents algorithmes de recherche heuristique de correspondances entre éléments d'ontologies sont appliqués en boucle tant qu'aucune nouvelle correspondance entre les deux versions d'ontologies n'est identifiée.

Noy et Musen ont fait l'hypothèse que le modèle de connaissances est compatible avec le protocole OKBC (Open Knowledge Base Connectivity), qui est basé sur la notion de système de représentation en frames (FRS – Frame Representation System) introduit par Minsky dans les années 1970. Une ontologie est donc basiquement constituée de *concepts atomiques* hiérarchisées, d'*individus* peuplant ces classes, de rôles atomiques associés à des classes en *des emplacements* (*slots*), et associés à des *facettes* (*facets*) qui restreignent la valeur des emplacements. Les éléments d'ontologie (i.e., concepts atomiques, individus, rôles atomiques) sont nommés frames (en français : cadres).

Entre deux versions  $o_1$  et  $o_2$  d'une même ontologie, les changements sont représentés dans une *table PromptDiff* [18] par un ensemble de quintuplets  $\langle F_1, F_2, valeur\_renommé, valeur\_opération, niveau\_correspondance \rangle$  où :

- $F_1$  et  $F_2$  sont des frames appartenant respectivement à  $o_1$  et  $o_2$ . Dans le cas où une frame est ajoutée (*resp.* supprimée) entre les deux versions,  $F_1$  (*resp.*  $F_2$ ) peut être *null* ;
- *valeur\_renommé* est *vrai* si les noms des frames sont identiques entre les versions, *faux* sinon ;
- *valeur\_opération* est une valeur prise dans l'ensemble  $\{ajout, suppression, division, fusion, correspondance\}$  ;
- *niveau\_correspondance* est une valeur prise dans l'ensemble  $\{inchangé, isomorphique, changé\}$ .

Un avantage certain de PromptDiff est qu'il est possible de calculer un journal de version quelque soit la méthode d'évolution d'ontologie utilisée : tout simplement à partir des deux ontologies avant et après l'évolution. Dans la suite de ce chapitre nous étudierons dans quelle mesure il est possible d'exploiter une *table PromptDiff* pour faire évoluer un alignement.

## 2.2 Evolution d'alignement d'ontologies à partir d'une table PromptDiff

Nous construirons dans cette section une méthode (un algorithme) d'évolution des alignements à partir d'une table PromptDiff de versionnage d'ontologie.

### 2.2.1 Algorithme naïf d'évolution d'alignement d'ontologies

On se place dans le cas d'un réseau d'ontologies  $S = \langle \Omega, \Lambda \rangle$  très simple, où  $\Omega = \{o_1, o_2\}$  sont deux ontologies des langages de description respectifs  $\mathcal{LD}_1$  et  $\mathcal{LD}_2$ , et  $\Lambda = \{A\}$  est un alignement entre  $o_1$  et  $o_2$ . On considère une nouvelle version  $o'_1$  de  $o_1$  et on se base sur une *table PromptDiff* qui décrit les différences entre  $o_1$  et  $o'_1$ .

Soit  $A = \{\langle e_1, e_2, R(e_1, e_2) \rangle \mid (e_1, e_2) \in \mathcal{Q}_{\mathcal{LD}_1}(o_1) \times \mathcal{Q}_{\mathcal{LD}_2}(o_2)\}$  l'alignement entre  $o_1$  et  $o_2$ , où :  
 $R : \mathcal{Q}_{\mathcal{LD}_1}(o_1) \times \mathcal{Q}_{\mathcal{LD}_2}(o_2) \rightarrow 2^E$  est la fonction qui associe à chaque paire d'éléments d'ontologie la relation qui les lie. On peut créer de manière très simple un nouvel alignement  $A'$  entre  $o'_1$  et  $o_2$  à partir de l'alignement vide de correspondances entre  $o'_1$  et  $o_2$  à l'aide de l'algorithme 1.

---

**Algorithm 1** Evolution naïve d'alignement d'ontologies à partir d'une table PromptDiff

---

**Require:**  $A = \{\langle e_1, e_2, R(e_1, e_2) \rangle \mid (e_1, e_2) \in \mathcal{Q}_{\mathcal{LD}_1}(o_1) \times \mathcal{Q}_{\mathcal{LD}_2}(o_2)\}$  un alignement normalisé

**Require:**  $R : \mathcal{Q}_{\mathcal{LD}_1}(o_1) \times \mathcal{Q}_{\mathcal{LD}_2}(o_2) \rightarrow 2^E$

1:  $A' \leftarrow A^0$

2: **for all**  $\langle e_1, e'_1, \text{valeur\_renommé}, \text{valeur\_opération}, \text{niveau\_correspondance} \rangle$  tel que  $e_1 \neq \text{null}$  et  $e_2 \neq \text{null}$  **do**

3:   **for all**  $e_2 \in \mathcal{Q}_{\mathcal{LD}_2}(o_2)$  **do**

4:      $A' \leftarrow A' \wedge \{\langle e'_1, e_2, R(e_1, e_2) \rangle\}$

5:   **end for**

6: **end for**

7: **return**  $A'$

---

Pour le moment on ignore complètement les informations complémentaires *valeur\_renommé*, *valeur\_opération*, et *niveau\_correspondance*. Sont-elles nécessaires ? A priori, la réponse est négative pour *valeur\_renommé*, et on continuera à ignorer cette information. L'information *niveau\_correspondance* n'est renseignée que lorsque *valeur\_opération* = *map* et pourra être utilisé pour manier les indices de confiance (c.f., section 2.3.4). Enfin, l'information *valeur\_opération* peut être déduite du nombre de lignes dans la table pour une même frame (c.f., tableau 2.1).

<i>valeur_opération</i>	$o_1$	$o'_1$
<i>ajout</i>	0	1
<i>suppression</i>	1	0
<i>division</i>	1	2..*
<i>fusion</i>	2..*	1
<i>correspondance</i>	1	1

Table 2.1: Nombre de frames dans  $o_1$  et  $o'_1$  associés à chaque opération.

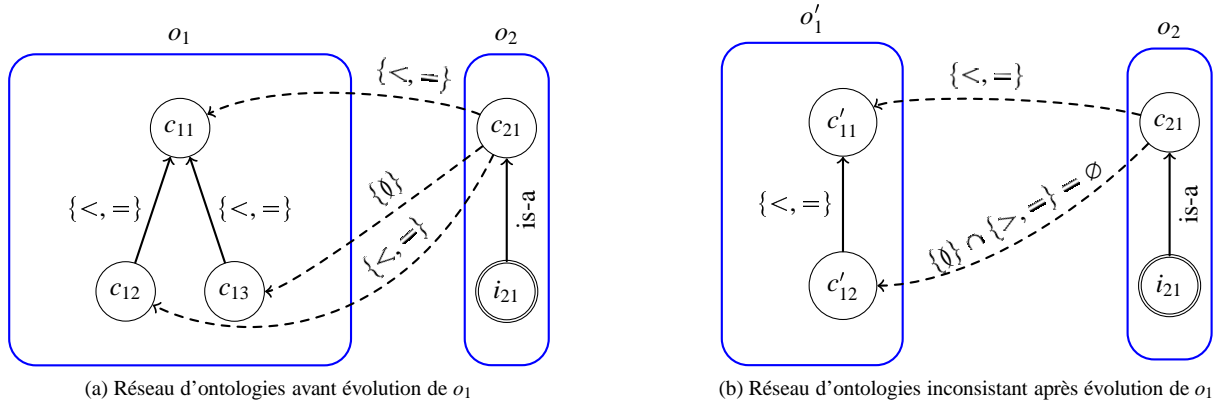
Pour le moment, le comportement de cet algorithme en fonction de la valeur de l'information *valeur\_opération* est le suivant :

- *ajout*, on n'ajoute pas de correspondance pour les nouveaux éléments d'ontologies (c'est l'objectif d'un algorithme de poursuite d'alignement, comme nous le verrons dans la section 2.3.5) ;
- *suppression*, on supprime une correspondance lorsque l'élément qu'elle liait est supprimé ;
- *division*, on duplique une correspondance lorsque le concept qu'elle lie est divisé ;
- *fusion*, on fusionne les correspondances à l'aide de l'opérateur d'intersection lorsque les éléments qu'elles lient sont fusionnés ;
- *correspondance*, on recopie simplement une correspondance lorsque l'élément qu'elle lie est dans une relation 1 pour 1.

Les comportements pour l'ajout, la suppression, la division, la correspondance, sont conformes à notre intuition. On verra qu'il est néanmoins possible qu'une inconsistance soit générée.

En revanche, le choix de l'opérateur d'intersection des relations lors d'une fusion de concepts semble être plus problématique. Il est évident que procéder à l'intersection de relations incompatibles crée une relation vide, donc une

incohérence. Si une instance peuple une des classes nécessairement vides, on se retrouve alors avec un réseau inconsistant. La figure 2.1 illustre très simplement un tel cas : on fusionne ici deux éléments d'une ontologie, tous deux en correspondance avec le même élément d'une autre ontologie, mais avec des relations incompatibles : l'une est une équivalence, et l'autre est une disjonction. Si on procède à la conjonction des deux relations des correspondances, alors on obtient un réseau inconsistant.



$F_1$	$F_2$	valeur_renommé	valeur_opération	niveau_correspondance
$c_{11}$	$c'_{11}$	faux	correspondance	inchangé
$c_{12}$	$c'_{12}$	vrai	fusion	
$c_{13}$	$c'_{12}$	vrai	fusion	

(c) Table PromptDiff entre les versions  $o_1$  et  $o'_1$

Figure 2.1: Inconsistance après conjonction des relations des correspondances pour la fusion de  $c_{12}$  et  $c_{13}$ .

## 2.2.2 Inconsistances et construction d'un nouvel algorithme

Il serait possible de rétablir la consistance en supprimant simplement la correspondance (c'est-à-dire en remplaçant la relation par  $\mathcal{E}$ ), mais on peut éviter ce phénomène en proposant d'autres comportements pour la fusion : on peut choisir seulement une des correspondances, ou simplement les supprimer. Cependant ces comportements ne semblent pas minimaux car ils ne prennent pas en compte l'ensemble des correspondances que l'on fusionne. On propose donc d'utiliser l'opérateur d'union des relations des correspondances (c.f., algorithme 2).

---

### Algorithm 2 Evolution d'alignement d'ontologies à partir d'une table PromptDiff

---

**Require:**  $A = \{\langle e_1, e_2, R(e_1, e_2) \rangle \mid (e_1, e_2) \in \mathcal{Q}_{\mathcal{LD}_1}(o_1) \times \mathcal{Q}_{\mathcal{LD}_2}(o_2)\}$  un alignement normalisé

**Require:**  $R : \mathcal{Q}_{\mathcal{LD}_1}(o_1) \times \mathcal{Q}_{\mathcal{LD}_2}(o_2) \rightarrow 2^{\mathcal{E}}$

**Require:**  $R' : \mathcal{Q}_{\mathcal{LD}_1}(o'_1) \times \mathcal{Q}_{\mathcal{LD}_2}(o_2) \rightarrow 2^{\mathcal{E}}$

```

1:  $A' \leftarrow \{\langle e_1, e_2, \emptyset \rangle \mid (e_1, e_2) \in \mathcal{Q}_{\mathcal{LD}_1}(o_1) \times \mathcal{Q}_{\mathcal{LD}_2}(o_2)\}$ 
2: for all  $\langle e_1, e'_1, \text{valeur\_renommé}, \text{valeur\_opération}, \text{niveau\_correspondance} \rangle$  tel que  $e_1 \neq \text{null}$  et  $e_2 \neq \text{null}$  do
3:   for all  $e_2 \in \mathcal{Q}_{\mathcal{LD}_2}(o_2)$  do
4:      $A' \leftarrow A' \vee \{\langle e'_1, e_2, R(e_1, e_2) \rangle\}$ 
5:   end for
6: end for
7: {A} ce point, un certain nombre de relations peuvent être vides
8: for all  $\langle e'_1, e_2, R'(e'_1, e_2) \rangle \in A'$  do
9:   if  $R'(e'_1, e_2) = \emptyset$  then
10:     $R'(e'_1, e_2) \leftarrow \mathcal{E}$ 
11:   end if
12: end for
13: return  $A'^*$ 

```

---

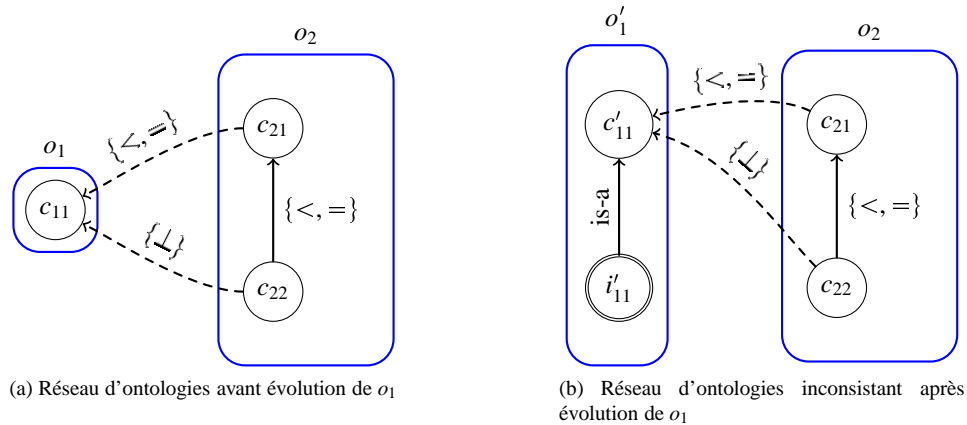
Cet algorithme présente également un effet de bord. En effet, si une relation vide  $\emptyset$  est associée à une correspondance de l'alignement de départ, alors cette relation sera transformée en  $\mathcal{E}$  dans l'alignement de sortie.

## 2.3 Limites de la méthode, ouvertures

Bien que l'application d'un tel algorithme d'évolution des alignements permette de faire évoluer la plupart des correspondances sans problème, il existe néanmoins des cas problématiques. Dans cette section nous proposons une mise en évidence de deux problèmes inhérents à la méthode, un problème plus particulièrement lié à PromptDiff. Nous proposons enfin deux ouvertures de la méthode.

### 2.3.1 Inconsistances et consolidation d'un réseau d'ontologies alignées

Malgré toutes les précautions que l'on peut prendre lors de l'évolution des alignements, il peut arriver qu'une inconsistance soit générée dans le réseau lorsqu'une ontologie évolue. La figure 2.2 illustre un exemple très simple dans lequel le simple ajout d'une instance dans une ontologie  $o_1$  rend inconsistant un réseau qui était déjà incohérent.



$F_1$	$F_2$	<i>valeur_renommé</i>	<i>valeur_opération</i>	<i>niveau_correspondance</i>
$c_{11}$	$c'_{11}$	<i>faux</i>	<i>correspondance</i>	<i>changé</i>
<i>null</i>	$i'_{11}$	<i>faux</i>	<i>ajout</i>	

(c) Table PromptDiff entre les versions  $o_1$  et  $o'_1$

Figure 2.2: Réseau d'ontologies inconsistant après ajout d'une instance dans  $o_1$ .

L'objectif de cette partie est de propager l'alignement à une nouvelle version d'ontologie, et il n'est pas question a priori de consolider le réseau (i.e., de rétablir sa consistance) lorsqu'une inconsistance est générée. Cela devrait faire l'objet d'un travail sur la consolidation d'un réseau d'ontologies inconsistant. Cependant, l'évolution d'alignements d'ontologies recouvre en partie la consolidation d'un réseau d'ontologies localement consistant.

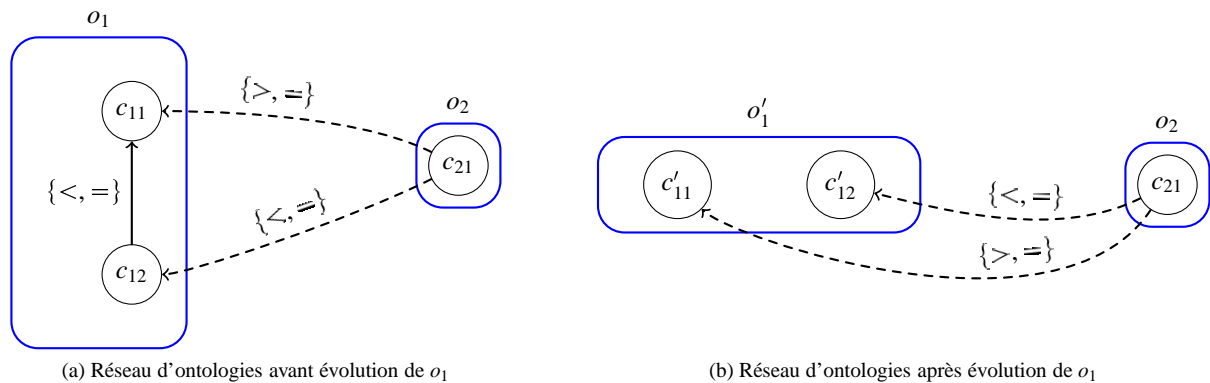
En effet, puisqu'on fait l'hypothèse que chaque ontologie est consistante localement, il est toujours possible de consolider le réseau : une possibilité drastique est de supprimer les alignements, c'est-à-dire de remplacer chaque alignement  $A$  par un alignement vide de correspondances  $A^0$ .

D'un autre côté, la consolidation du réseau ne s'applique pas au seul contexte où on a une évolution dans une ontologie du réseau. En effet, la consolidation du réseau telle qu'on l'étudie dans cette partie est un processus qui prend en entrée un réseau localement consistant mais globalement inconsistant, et qui a en sortie un réseau localement et globalement consistant.

On peut voir la consolidation du réseau comme une évolution particulière du réseau : la contraction du concept vide dans le réseau (i.e., on ne veut plus pouvoir inférer le concept vide). On étudiera cela en tant que cas particulier de la contraction d'un ensemble de propositions dans le réseau, au chapitre suivant.

### 2.3.2 Une relation supprimée peut encore être inférée

Un autre exemple de limite inhérente à la méthode peut être illustré sur la figure 2.3 suivante : une relation de subsumption a été supprimée entre deux concepts d'une ontologie, et cette relation était et est encore également une conséquence du réseau d'ontologies.



$F_1$	$F_2$	<i>valeur_renommé</i>	<i>valeur_opération</i>	<i>niveau_correspondance</i>
$c_{11}$	$c'_{11}$	<i>faux</i>	<i>correspondance</i>	<i>changé</i>
$c_{12}$	$c'_{12}$	<i>faux</i>	<i>correspondance</i>	<i>changé</i>

(c) Table PromptDiff entre les versions  $o_1$  et  $o'_1$

Figure 2.3: Une subsumption supprimée est encore une conséquence du réseau.

Dans le cas où des formules ont été supprimées de l'ontologie qui a évolué, la méthode proposée dans ce chapitre ne permet pas de vérifier que ces formules ne sont plus des conséquences du réseau. En effet, on a un accès explicite aux évolutions des éléments de l'ontologie, et un accès seulement implicite aux évolutions des formules qui lient ces éléments d'ontologie.

Le chapitre suivant sera l'occasion de proposer une méthode qui permette de palier ce problème. On considèrera qu'on a accès aux informations de l'évolution à un niveau plus sémantique : quelles formules ont été contractées ? (i.e., retirées de l'ontologie).

### 2.3.3 Problème des concepts et rôles composites

Étant basé sur le protocole OKBC, PromptDiff n'est pas en mesure de prendre en compte des classes ou des rôles composites. Ainsi, le processus de construction du nouvel alignement décrit ci-dessus est incomplet, puisqu'il existe des éléments de  $\mathcal{Q}_{\mathcal{L}\mathcal{D}_1}(o'_1)$  qui ne sont pas des frames. En théorie, dès que le langage de description d'une ontologie possède des constructeurs de concepts ou de rôles, le nombre d'éléments composites de  $\mathcal{Q}_{\mathcal{L}\mathcal{D}_1}(o_1)$  est beaucoup plus grand que le nombre d'éléments atomiques. En pratique toutefois, le nombre de concepts et de rôles composites est limité. Le taux de correspondances propagées à la nouvelle version de l'ontologie par la méthode que l'on propose dépend donc directement du taux d'éléments composites de l'ontologie.

Pour propager les correspondances qui lient des éléments composites, on peut utiliser le fait qu'ils sont construits à partir de frames. Une extension de ce travail pourrait donc consister en envisager une solution pour reconstruire les lignes de la table PromptDiff qui concernent les éléments composites des ontologies. Dans les graphes des deux versions d'ontologies, cette solution pourrait prendre la forme d'une recherche de graphes homomorphiques ayant leurs concepts et rôles atomique appareillés dans la table PromptDiff. Cela pourra être un point de recherche pour un travail ultérieur sur l'évolution d'alignement à partir d'un journal de PromptDiff.

### 2.3.4 Gestion des indices de confiance

On peut considérer qu'on peut perdre un peu confiance en certaines correspondances si elles sont propagées de manière automatique à une nouvelle version d'ontologie. Dans ce travail on a mis les indices de confiance de côté, mais on va néanmoins proposer quelques idées qualitatives concernant leur gestion dans l'algorithme, et leur utilisation pour la consolidation du réseau.

Lorsqu'un élément de l'ancienne version est dans une relation 1 pour 1 avec un élément de la nouvelle version d'ontologie, on peut imaginer réduire l'indice de confiance en multipliant l'ancien indice de confiance par :

- 1 lorsque *niveau\_correspondance* = *inchangé* ;
- 0,9 lorsque *niveau\_correspondance* = *isomorphic* ;
- 0,8 lorsque *niveau\_correspondance* = *changé*.

De la même manière, on peut imaginer calculer un nouvel indice de confiance pour la correspondance fusionnée à partir d'une mesure de la valeur centrale des indices de confiance des correspondances initiales (e.g., moyenne arithmétique, moyenne géométrique, etc.), avec éventuellement une réduction en multipliant cette moyenne par une valeur inférieure à 1 (e.g., 0,9).

Ainsi, lorsqu'on détecte une inconsistance dans le réseau, que l'on extrait l'ensemble inconsistant minimal de formules, et que des correspondances sont impliquées dans cet ensemble de formules, on pourrait supprimer la correspondance au plus faible indice de confiance, à la manière de Meilicke et al. dans [17]. Ce point pourra également faire l'objet d'un travail ultérieur, plus précisément sur la consolidation d'un réseau inconsistant d'ontologies.

### 2.3.5 Poursuite de la recherche des correspondances entre les ontologies

Une fois notre méthode de propagation des alignements appliquée, de nouveaux éléments peuvent nécessiter une recherche plus précise de correspondances. En particulier pour les éléments qui ont été ajoutés ou ceux qui ont été divisés.

L'intérêt de la méthode d'évolution des alignements qu'on a proposé est double : d'une part elle prémâche le travail d'un algorithme de recherche incrémentale d'alignements en lui fournissant un alignement de départ, et d'autre part elle devrait permettre à cet algorithme de cibler les éléments qui sont susceptibles d'être impliqués dans de nouvelles correspondances.



## Chapter 3

# Méthodes d'évolution des réseaux d'ontologies basées sur la théorie de la révision

La théorie de la révision dans le domaine du changement de croyances a été introduite par Alchourròn, Gärdenfors, et Makinson en 1985 [1]. Cette théorie a pour objectif de modifier automatiquement une logique lorsque de nouvelles informations sont prises en compte. Alchourròn, Gärdenfors, et Makinson ont défini différentes opérations binaires pour prendre en compte une proposition  $\phi$  dans une théorie  $K$  d'une logique de Tarski. Les auteurs ont proposé plusieurs postulats – connus sous le nom de postulats AGM – pour chaque opérateurs de sorte qu'ils se comportent de façon rationnelle.

Puisque nous ne considérons que des ontologies de logiques de description, dans notre étude l'évolution d'ontologies est un cas particulier du changement de croyances. Nous nous proposons donc d'étudier comment il est possible d'adapter cette théorie pour faire évoluer les alignements des réseaux d'ontologies.

Dans la première section nous présenterons la théorie de la révision appliquée aux ontologies, puis nous proposerons une extension de cette théorie aux réseaux d'ontologies. Nous verrons alors dans quelle mesure une évolution locale dans une ontologie peut être problématique au niveau du réseau tout entier.

L'étude de la satisfaisabilité des postulats de rationalité pour un opérateur d'évolution globale fera l'objet de la seconde section de ce chapitre.

En dernier lieu nous proposerons une étude plus approfondie en nous basant sur l'algèbre de relations présentée au premier chapitre de ce rapport.

### 3.1 Introduction à la théorie de la révision appliquée aux réseaux d'ontologies

Les postulats AGM ont été généralisés de bien des façons, nous présentons ci-dessous une généralisation qui permet de prendre en compte la révision (resp. contraction) de plusieurs formules en paquet. Nous nous inspirons de Hansson [14] et de Euzenat<sup>1</sup>.

#### 3.1.1 Opérateurs d'évolution d'une ontologie

Soit  $K = Cn(K)$  une théorie d'une logique de Tarski  $\mathcal{L}$  (un ensemble de croyances clôt par inférence), et  $X$  un ensemble de formules. Un opérateur d'évolution d'un ensemble de croyances est une fonction  $\mathcal{P}(\mathcal{L}) \times \mathcal{P}(\mathcal{L}) \rightarrow \mathcal{P}(\mathcal{L})$ , où  $\mathcal{P}(\mathcal{L})$  est l'ensemble des parties de l'ensemble  $\mathcal{L}$ .

L'expansion  $K + X = Cn(K \cup X)$  est la plus simple des opérations. Il s'agit d'ajouter l'ensemble des formules logiques  $X$  sans vérifier la consistance du nouvel ensemble de croyances. D'autres opérateurs sont définis lorsqu'on a une nouvelle croyance qu'un ensemble de formules  $X$  est vrai (révision  $K \oplus X$ ), n'est pas vrai (contraction  $K \ominus X$ ), est désormais vrai (mise-à-jour), est désormais plus vrai (effacement). On se concentrera sur les opérateurs de révision, et de contraction.

---

<sup>1</sup>Jérôme Euzenat, Revision in networks of ontologies, communication personnelle

## Révision

$K \oplus X$  est l'ensemble des croyances résultant de la révision de  $K$  par l'ensemble de formules  $X$ .

Pour se comporter de manière rationnelle, l'opérateur  $\oplus$  doit satisfaire les postulats suivants :

- ⊕1/clôture  $K \oplus X = Cn(K \oplus X)$
- ⊕2/succès=R1  $X \subseteq K \oplus X$
- ⊕3/confinement  $K \oplus X \subseteq Cn(K \cup X)$
- ⊕4/conservation  $\perp \notin Cn(K \cup X) \Rightarrow Cn(K \cup X) \subseteq K \oplus X$
- ⊕5/consistance  $\perp \subseteq Cn(K \oplus X) \Rightarrow \perp \subseteq K$  ou  $\perp \subseteq Cn(X)$
- ⊕6/équivalence=R4  $Cn(X) = Cn(Y) \Rightarrow K \oplus X = K \oplus Y$

Les six premiers postulats sont nommés “postulats AGM de base”. Le nom après le slash correspond à une traduction littérale de la classification de Sven Hansson [14]. Enfin, la classification par des “R” correspond à la numérotation donnée dans l'article d'Alchourrón, Gärdenfors, et Makinson [1]. Deux autres postulats (intersection et conjonction) ne sont souvent pas pris en compte et ne sont pas reportés ici. En effet, ils utilisent la conjonction d'assertions, qui n'est pas forcément définie dans la logique utilisée. Le sens intuitif de ces postulats est le suivant :

- ⊕1/clôture le résultat est déductivement clôt ;
- ⊕2/succès=R1  $X$  doit être dans la théorie révisée ;
- ⊕3/confinement l'opération ne doit pas ajouter plus de croyance que la simple addition de  $X$  ;
- ⊕4/conservation si  $X$  est compatible avec les croyances actuelles, alors l'ajouter simplement ;
- ⊕5/consistance la révision ne doit pas générer d'inconsistance ;
- ⊕6/équivalence=R4 la révision doit être neutre syntaxiquement.

## Contraction

$K \ominus X$  est l'ensemble des croyances résultant de la contraction de  $K$  par l'ensemble de formules  $X$ .

Pour se comporter de manière rationnelle, l'opérateur  $\ominus$  doit satisfaire les postulats suivants :

- ⊖1/clôture  $K \ominus X = Cn(K \ominus X)$
- ⊖2/succès=C3  $\perp \notin Cn(X) \Rightarrow X \cap K \ominus X = \emptyset$
- ⊖3/confinement=C1  $K \ominus X \subseteq K$
- ⊖4/conservation=C2  $X \cap K = \emptyset \Rightarrow K \ominus X = K$
- ⊖5/consistance  $\perp \subseteq Cn(K \ominus X) \Rightarrow \perp \subseteq K$  ou  $\perp \subseteq Cn(X)$
- ⊖6/équivalence=C4  $Cn(X) = Cn(Y) \Rightarrow K \ominus X = K \ominus Y$
- ⊖7/rétablissement=C5  $K \subseteq (K \ominus X) \cup X$

Soit en toute lettre:

- ⊖1/clôture le résultat est déductivement clôt ;
- ⊖2/succès=C1 aucun élément de  $X$  ne doit être dans la théorie contractée ;
- ⊖3/confinement=C3 la contraction ne doit pas ajouter de croyance ;
- ⊖4/conservation=C2 si aucun élément de  $X$  n'est dans les croyances actuelles, alors on ne fait rien ;
- ⊖5/consistance la contraction ne doit pas générer d'inconsistance ;
- ⊖6/équivalence=C4 la contraction doit être neutre syntaxiquement ;
- ⊖7/rétablissement=C5 si on ajoute de nouveau  $X$ , on retrouve au moins la théorie de départ.

Si la négation est définie dans le langage utilisé, et si la négation d'une base de connaissances a un sens, alors les opérateurs de révision et de contraction sont inter-définis par l'identité de Lévi :

$$K \oplus X = Cn((K \ominus \neg X) \cup X)$$

et son contraire, l'identité de Harper :

$$K \ominus X = Cn((K \cap (K \oplus \neg X)))$$

### 3.1.2 Extension aux réseaux d'ontologies

Soit le réseau d'ontologies  $\langle \Omega, \Lambda \rangle$  formé d'un ensemble d'ontologies  $\Omega = (o_i)_{i \in I}$  et d'un ensemble d'alignements  $\Lambda = (A_{ij})_{(i,j) \in I^2, i \neq j}$ .

Pour représenter l'ensemble des formules par lesquelles on révisé (resp. contracte) le réseau, on propose d'utiliser la même structure que celle du réseau d'ontologies :  $\langle X, Y \rangle$ .  $X = (X_i)_{i \in I}$  où  $X_i$  est un ensemble de formules associées à l'ontologie  $o_i$ .  $Y = (Y_{ij})_{(i,j) \in I^2, i \neq j}$ , où  $Y_{ij}$  est un ensemble de correspondances associées à l'alignement  $A_{ij}$ .

Une fois qu'on a défini les opérateurs  $\cup$ ,  $Cn$ , la relation  $\subseteq$  et la notion d'inconsistance globale pour les réseaux d'ontologies, l'écriture des postulats de rationalité des opérateurs globaux découlent directement des postulats de rationalité des opérateurs locaux.

### Révision

$\langle \Omega, \Lambda \rangle \boxplus \langle X, Y \rangle$  est le réseau d'ontologies résultant de la révision de  $\langle \Omega, \Lambda \rangle$  par l'ensemble de formules  $\langle X, Y \rangle$ .

Pour se comporter de manière rationnelle, l'opérateur  $\boxplus$  doit satisfaire les postulats suivants :

- $\boxplus 1$ /clôture  $\langle \Omega, \Lambda \rangle \boxplus \langle X, Y \rangle = Cn(\langle \Omega, \Lambda \rangle \boxplus \langle X, Y \rangle)$
- $\boxplus 2$ /succès  $\langle X, Y \rangle \subseteq \langle \Omega, \Lambda \rangle \boxplus \langle X, Y \rangle$
- $\boxplus 3$ /confinement  $\langle \Omega, \Lambda \rangle \boxplus \langle X, Y \rangle \subseteq Cn(\langle \Omega, \Lambda \rangle \cup \langle X, Y \rangle)$
- $\boxplus 4$ /conservation  $\perp \notin Cn(\langle \Omega, \Lambda \rangle \cup \langle X, Y \rangle) \Rightarrow Cn(\langle \Omega, \Lambda \rangle \cup \langle X, Y \rangle) \subseteq \langle \Omega, \Lambda \rangle \boxplus \langle X, Y \rangle$
- $\boxplus 5$ /consistance  $\perp \subseteq Cn(\langle \Omega, \Lambda \rangle \boxplus \langle X, Y \rangle) \Rightarrow \perp \subseteq \langle \Omega, \Lambda \rangle$  ou  $\perp \subseteq Cn(\langle X, Y \rangle)$
- $\boxplus 6$ /équivalence  $Cn(\langle X, Y \rangle) = Cn(\langle X', Y' \rangle) \Rightarrow \langle \Omega, \Lambda \rangle \boxplus \langle X, Y \rangle = \langle \Omega, \Lambda \rangle \boxplus \langle X', Y' \rangle$

### Contraction

$\langle \Omega, \Lambda \rangle \boxminus \langle X, Y \rangle$  est le réseau d'ontologies résultant de la contraction de  $\langle \Omega, \Lambda \rangle$  par l'ensemble de formules  $\langle X, Y \rangle$ .

Pour se comporter de manière rationnelle, l'opérateur  $\boxminus$  doit satisfaire les postulats suivants :

- $\boxminus 1$ /clôture  $\langle \Omega, \Lambda \rangle \boxminus \langle X, Y \rangle = Cn(\langle \Omega, \Lambda \rangle \boxminus \langle X, Y \rangle)$
- $\boxminus 2$ /succès  $\perp \notin Cn(\langle X, Y \rangle) \Rightarrow \langle X, Y \rangle \cap \langle \Omega, \Lambda \rangle \boxminus \langle X, Y \rangle = \emptyset$
- $\boxminus 3$ /confinement  $\langle \Omega, \Lambda \rangle \boxminus \langle X, Y \rangle \subseteq \langle \Omega, \Lambda \rangle$
- $\boxminus 4$ /conservation  $\langle X, Y \rangle \cap \langle \Omega, \Lambda \rangle = \emptyset \Rightarrow \langle \Omega, \Lambda \rangle \boxminus \langle X, Y \rangle = \langle \Omega, \Lambda \rangle$
- $\boxminus 5$ /consistance  $\perp \subseteq Cn(\langle \Omega, \Lambda \rangle \boxminus \langle X, Y \rangle) \Rightarrow \perp \subseteq \langle \Omega, \Lambda \rangle$  ou  $\perp \subseteq Cn(\langle X, Y \rangle)$
- $\boxminus 6$ /équivalence  $Cn(\langle X, Y \rangle) = Cn(\langle X', Y' \rangle) \Rightarrow \langle \Omega, \Lambda \rangle \boxminus \langle X, Y \rangle = \langle \Omega, \Lambda \rangle \boxminus \langle X', Y' \rangle$
- $\boxminus 7$ /rétablissement  $\langle \Omega, \Lambda \rangle \subseteq Cn(\langle \Omega, \Lambda \rangle \boxminus \langle X, Y \rangle \cup \langle X, Y \rangle)$

## 3.1.3 Impact d'une évolution locale sur le réseau d'ontologie

### Révision

On considère dans un réseau d'ontologies  $\langle \Omega, \Lambda \rangle$  sur l'ensemble des index  $I$ , que l'ontologie  $o_i$  indiquée  $i$  subit une opération locale de révision par  $X_i$ . Soit  $o'_i = o_i \oplus X_i$  l'ontologie résultante de cette révision,  $\oplus$  étant un opérateur qui satisfait les postulats  $\oplus 1 - \oplus 6$ . Soit  $\langle X, Y \rangle$  tel que  $X = (X_k)_{k \in I}$  où  $X_k = \emptyset \forall k \neq i$ ,  $i$  est l'index de l'ontologie révisée, et  $Y = (Y_{kl})_{(k,l) \in I^2, k \neq l}$  où  $Y_{kl} = \emptyset \forall (k, l) \in I^2, k \neq l$ .

Soit  $\boxplus$  un opérateur de révision global défini par  $\langle \Omega, \Lambda \rangle \boxplus \langle X, Y \rangle = Cn(\langle \Omega', \Lambda \rangle)$ , où  $\Omega' = (o'_k)_{k \in I}$ ,  $o'_k = o_k \forall k \neq i$ , et  $o'_i = o_i \oplus X_i$ . La question que l'on se pose est : parmi les postulats  $\boxplus 1 - \boxplus 6$ , lesquels sont satisfaits par l'opérateur  $\boxplus$  ?

$\boxplus 1$  est satisfait de par la définition de l'opérateur. Les postulats  $\boxplus 2 - \boxplus 4$  et  $\boxplus 6$  découlent directement des postulats  $\oplus 2 - \oplus 4$  et  $\oplus 6$ . Reste  $\boxplus 5$  (i.e., consistance) qui n'est pas satisfait a priori (c.f., figure 3.1)

$$o'_1 = o_1 \oplus \{c_{11} \perp c_{12}\}$$

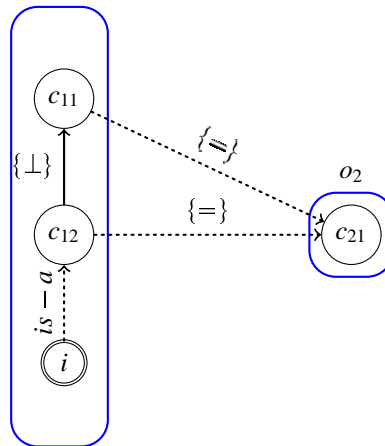


Figure 3.1: Une incohérence dans le réseau après révision locale par  $c_{11} \perp c_{12}$

Ainsi, lorsqu'une opération de révision a lieu dans une (ou plusieurs) ontologies du réseau, alors il suffit de rétablir la consistance dans le réseau si nécessaire, c'est-à-dire appliquer une opération de consolidation à l'ensemble du réseau.

### Contraction

On considère dans un réseau d'ontologies  $\langle \Omega, \Lambda \rangle$  sur l'ensemble des index  $I$ , que l'ontologie  $o_i$  indiquée  $i$  subit une opération locale de contraction par  $X_i$ . Soit  $o'_i = o_i \ominus X_i$  l'ontologie résultante de cette révision,  $\ominus$  étant un opérateur qui satisfait les postulats  $\ominus 1 - \ominus 7$ . Soit  $\langle X, Y \rangle$  tel que  $X = (X_k)_{k \in I}$  où  $X_k = \emptyset \forall k \neq i$ ,  $i$  est l'index de l'ontologie contractée, et  $Y = (Y_{kl})_{(k,l) \in I^2, k \neq l}$  où  $Y_{kl} = \emptyset \forall (k,l) \in I^2, k \neq l$ .

Soit  $\square$  un opérateur global de contraction défini par  $\langle \Omega, \Lambda \rangle \square \langle X, Y \rangle = Cn(\langle \Omega', \Lambda \rangle)$ , où  $\Omega' = (o'_k)_{k \in I}$ ,  $o'_k = o_k \forall k \neq i$ , et  $o'_i = o_i \ominus X_i$ . La question qui se pose est : parmi les postulats  $\square 1 - \square 7$ , lesquels sont satisfaits par l'opérateur  $\square$  ?

$\square 1$  est satisfait de par la définition de l'opérateur. Les postulats  $\square 3 - \square 7$  découlent directement des postulats  $\ominus 3 - \ominus 7$ . Reste  $\square 2$  (i.e., succès) qui n'est pas satisfait a priori (c.f., figure 3.2)

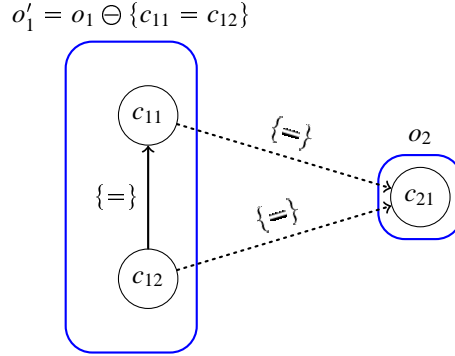


Figure 3.2: Ce qui a été contracté dans  $o_1$  peut encore être inféré dans le réseau.

Ainsi, lorsqu'une opération de contraction a lieu dans une (ou plusieurs) ontologies du réseau, alors il est juste nécessaire de s'assurer que ce qui a été contracté ne peut plus être inféré dans le réseau, tout en conservant la satisfaction des autres postulats.

## 3.2 Etudes préliminaires de satisfaisabilité des postulats

Soit  $S$  un réseau d'ontologies qui subit une évolution locale de l'une de ses ontologies. On cherche une opération à faire subir au réseau pour redresser les postulats. i.e., après cette opération, on souhaite que les postulats de l'opération globale d'évolution correspondant à l'opération locale qui a eu lieu soient satisfaits.

Puisque le sujet de ce travail est l'évolution des alignements, on fait les hypothèses suivantes :

- le réseau d'ontologies est consistant avant l'évolution locale de l'une des ontologies ;
- chacune des ontologies du réseau est consistante après l'évolution locale de l'une des ontologies ;
- on ne peut modifier que les alignements du réseau.

Pour factoriser les notations, on note  $\circledast$  un opérateur d'évolution local (indifféremment  $\oplus$  ou  $\ominus$ ), et  $\square$  un opérateur d'évolution global (indifféremment  $\boxplus$  ou  $\boxminus$ ). Soit donc  $S = \langle \Omega, \Lambda \rangle$  un réseau d'ontologies normalisé formé d'une famille d'ontologies  $\Omega = (o_i)_{i \in I}$ , et d'une famille d'alignements normalisés  $\Lambda = (A_{ij}^*)_{(i,j) \in I^2}$ .

Une évolution  $\circledast$  est appliquée sur l'ontologie  $o_i$  par l'ensemble de formules  $X_i$  :  $o'_i = o_i \circledast X_i$ . Soit  $\langle X, Y \rangle$  tel que  $X = (X_k)_{k \in I}$  où  $X_k = \emptyset \forall k \neq i$ ,  $i$  est l'index de l'ontologie contractée, et  $Y = (Y_{kl})_{(k,l) \in I^2, k \neq l}$  où  $Y_{kl} = \emptyset \forall (k,l) \in I^2, k \neq l$ .

Soit  $\Omega' = \{(o_j)_{j \in I, j < i}, o'_i, (o_j)_{j \in I, j > i}\}$ . On cherche une opération de redressement des postulats qui transforme  $\Lambda$  en  $\Lambda'$  de façon minimale, et de sorte que  $S' = \langle \Omega, \Lambda \rangle \square \langle X, Y \rangle = Cn(\langle \Omega', \Lambda' \rangle)$  soit un produit valide de l'évolution globale du réseau par  $\langle X, Y \rangle$ .

### 3.2.1 Solution drastique de redressement des postulats

Etudions l'effet de la solution drastique qui consiste à remplacer chaque alignement  $A_{ij}$  du réseau par l'alignement vide de correspondance  $A_{ij}^0$  correspondant.

On a fait l'hypothèse que chacune des ontologies du réseau est consistante après la révision locale, on a donc au moins une famille de modèles pour les ontologies. Soit  $(m_i)_{i \in I}$  une telle famille :  $(m_i)_{i \in I} \in \times_{o \in \Omega} \mathcal{M}(o)$ . Soit également  $\Gamma$  l'ensemble des fonctions d'égalisation pour le réseau d'ontologies, et  $\gamma \in \Gamma$  une fonction d'égalisation particulière qui conserve les modèles, c'est-à-dire que  $(\gamma(m_i))_{i \in I} \in \times_{o \in \Omega} \mathcal{M}(o)$ .

#### Révision

Le résultat pour la révision est immédiat. Il suffit de rappeler que  $A_{ij}^0$  est un alignement valide quels que soient les index  $i$  et  $j$ , et que comme  $\gamma$  conserve les modèles,  $\forall (i, j) \in I^2, m_i, m_j \models_{\gamma} A_{ij}^0$ .

Ainsi, il est direct que  $\langle (m_i)_{i \in I}, \gamma \rangle \in \times_{i \in I} \mathcal{M}(o_i) \times \Gamma \mid \forall (i, j) \in I^2, m_i, m_j \models_{\gamma} A_{ij}^0$ .

$\langle (m_i)_{i \in I}, \gamma \rangle \in \mathcal{M}(\langle \Omega, \Lambda \rangle)$  est donc un modèle du réseau d'ontologies, qui est donc consistant.

Ainsi, en choisissant des alignements vides de correspondances, chaque famille de modèles des ontologies associée à une fonction d'égalisation qui conserve les modèles est un modèle pour le réseau d'ontologies. Un corollaire intéressant pour la suite est que chaque modèle d'une ontologie est impliqué dans au moins un modèle du réseau, i.e.,

$$\forall i \in I, \forall m_i \in \mathcal{M}(o_i), \exists ((m_j)_{j \in I, j \neq i}, \gamma) \in \times_{j \in I, j \neq i} \mathcal{M}(o_j) \times \Gamma \mid \langle (m_j)_{j \in I}, \gamma \rangle \in \mathcal{M}(\langle \Omega, \Lambda \rangle) \quad (3.2.1)$$

Le postulat  $\boxplus 5$  de consistance est donc satisfaisable.

#### Contraction

Pour la contraction, l'ontologie  $o_i$  indexée  $i$  de la logique de description  $\mathcal{LD}_i$  a été contractée par l'ensemble des formules  $X_i$ . Le postulat  $\ominus 2$  étant satisfait, on peut dire que  $\forall f \in X_i, \exists m_i \in \mathcal{M}(o_i) \mid m_i \not\models f$ .

Pour mener la démonstration simplement, on restreint le choix de la fonction d'égalisation  $\gamma$  pour qu'elle soit telle que  $\forall (f, m_i) \in X_i \times \mathcal{M}(o_i), m_i \not\models f \Rightarrow m_i \not\models_{\gamma} f$ . Ainsi, pour toute formule de l'ensemble des formules contractées, il existe un modèle de  $o$  tel que cette formule n'est pas une conséquence sémantique de ce modèle dans le domaine d'interprétation universel. i.e.,

$$\forall f \in X_i, \exists m_i \in \mathcal{M}(o) \mid m_i \not\models_{\gamma} f \quad (3.2.2)$$

L'association des formules 3.2.2 et 3.2.1 nous permet de montrer que pour toute formule  $f$  de l'ensemble des formules contractées  $X_i$ , il existe un modèle du réseau tel que  $f$  n'est pas une conséquence sémantique du réseau. i.e.,

$$\forall f \in X_i, \exists \langle (m_j)_{j \in I}, \gamma \rangle \in \mathcal{M}(\langle \Omega, \Lambda \rangle) \mid m_i \not\models f$$

Par contraposition : chaque formule  $f$  de l'ensemble des formules contractées  $X_i$  ne fait pas partie de la clôture du réseau pour l'ontologie  $o_i$ .

Le postulat  $\boxminus 2$  de succès est donc satisfaisable. Cependant cette méthode nous fait perdre la satisfaction du postulat  $\boxminus 7$  de rétablissement.

### 3.2.2 Incompatibilité des postulats de rétablissement et de succès pour la contraction

Ainsi dans le cas d'une contraction, plus on appauvrit les relations des alignements en ajoutant des relations élémentaires, plus on a de chance de rétablir la satisfaction du postulat de succès. en contrepartie, on augmente les chances de perdre la satisfaction du postulat de rétablissement. Nous allons montrer qu'il existe un réseau et un ensemble de formules pour lesquels aucun opérateur global de contraction ne peut satisfaire à la fois les postulats de succès  $\boxminus 2$  et de rétablissement  $\boxminus 7$ .

Prenons le réseau clôt très simple de la figure 3.3 ci-dessous. L'ontologie  $o_1$  subit une contraction par la formule  $c_1 \{>, \perp\} c_2$ . Le réseau est choisi de telle sorte qu'il est possible de raisonner avec l'algèbre des relations, ce qui nous simplifie la tâche puisqu'on a défini dans la section 1.2.2 un ensemble d'opérations sur ces relations, ainsi que la notion de conséquence directe entre deux relations.

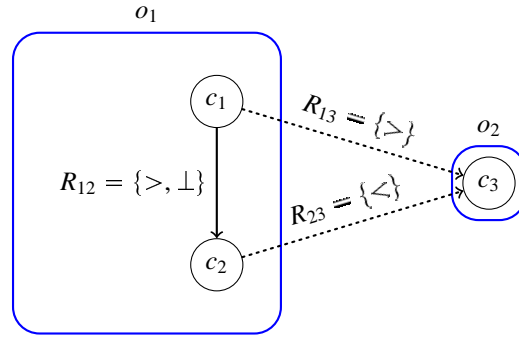


Figure 3.3: Réseau support de démonstration.

Soient les relations suivantes :  $R_{12} = \{>, \perp\}$  ;  $R_{21} = R_{12}^{-1}$  ;  $R_{13} = \{>\}$  ;  $R_{31} = R_{13}^{-1}$  ;  $R_{32} = \{>\}$  ;  $R_{23} = R_{32}^{-1}$ .

La formule  $c_1 R_{12} c_2$  est contractée dans l'ontologie  $o_1$ . Or, cette formule est encore une conséquence du réseau puisque au niveau des relations,  $R_{13} \cdot R_{32} = \{>\} \subseteq R_{12} = \{>, \perp\}$ . Donc  $R_{13} \cdot R_{32} \vdash R_{12}$ .

Pour respecter le postulat  $\square 2$ , i.e., pour que cette formule ne soit plus une conséquence du réseau, on doit ajouter des relations élémentaires dans les correspondances de l'alignement. On commence donc par ajouter une relation élémentaire à  $R_{13}$  (cas 1) ou à  $R_{32}$  (cas 2).

**Cas 1** On pose  $R'_{13} = \{>, r\}$ ,  $r \in \mathcal{E} \setminus \{>\}$ .

**Condition de satisfaction de  $\square 2$**  :  $\mathcal{E} \cap R'_{13} \cdot R_{32} \not\subseteq \{>, \perp\}$ .

$$\begin{aligned} \mathcal{E} \cap R'_{13} \cdot R_{32} &= \{>, r\} \cdot \{>\} \\ &= \{>\} \cup (\{r\} \cdot \{>\}) \end{aligned}$$

Finalement,  $\{>\} \cup (\{r\} \cdot \{>\}) \not\subseteq \{>, \perp\} \Rightarrow r \in \{<, \emptyset\}$ .

**Condition de satisfaction de  $\square 7$**  :  $R'_{13} \cap R_{12} \cdot R_{23} \subseteq \{>\}$ .

$$\begin{aligned} R'_{13} \cap R_{12} \cdot R_{23} &= \{>, r\} \cap (\{>, \perp\} \cdot \{<\}) \\ &= \{<, r\} \cap (\{>, <, =, \emptyset\} \cup \{<, \emptyset, \perp\}) \\ &= \{>, r\} \end{aligned}$$

Or  $r \in \mathcal{E} \setminus \{>\}$ , donc la condition de satisfaction de  $\square 7$  :  $R'_{13} \cap R_{12} \cdot R_{23} \subseteq \{>\}$  est impossible à réaliser.

**Cas 2** On pose  $R'_{32} = \{>, r\}$ ,  $r \in \mathcal{E} \setminus \{>\}$ .

**Condition de satisfaction de  $\square 2$**  :  $\mathcal{E} \cap R_{13} \cdot R'_{32} \not\subseteq \{>, \perp\}$ .

$$\begin{aligned} \mathcal{E} \cap R_{13} \cdot R'_{32} &= \{>\} \cdot \{>, r\} \\ &= \{>\} \cup (\{>\} \cdot \{r\}) \end{aligned}$$

Finalement,  $\{>\} \cup (\{>\} \cdot \{r\}) \not\subseteq \{>, \perp\} \Rightarrow r \in \{<, \emptyset, \perp\}$ .

**Condition de satisfaction de  $\square 7$**  :  $R'_{32} \cap R_{31} \cdot R_{12} \subseteq \{>\}$ .

$$\begin{aligned} R'_{32} \cap R_{31} \cdot R_{12} &= \{>, r\} \cap \{<\} \cdot \{>, \perp\} \\ &= \{>, r\} \end{aligned}$$

Or  $r \in \mathcal{E} \setminus \{>\}$ , donc la condition de satisfaction de  $\square 7$  :  $R'_{32} \cap R_{31} \cdot R_{12} \subseteq \{>\}$  est impossible à réaliser.

## Conclusions

En conclusion, si on veut que la formule  $c_{11} = c_{12}$  ne soit plus une conséquence du réseau lorsqu'on la contracte de l'ontologie  $o_1$ , alors on doit ajouter au moins une relation élémentaire à l'une des correspondances du réseau. On a vu que si on en ajoute une seule, alors on ne peut pas satisfaire à la fois le postulat de succès et le postulat de rétablissement. Il est trivial que si on ajoute une seconde relation élémentaire, alors le postulat de rétablissement sera encore plus dur à satisfaire.

Ainsi, on a montré que pour l'algèbre de relations qu'on utilise, il existe au moins un cas pour lequel on devra choisir entre la satisfaction du postulat de succès, ou la satisfaction du postulat de rétablissement, on choisit donc de mettre de côté le postulat de rétablissement pour la suite de notre travail. On ne proposera pas de version affaiblie de ce postulat, mais intuitivement on peut dire que satisfaire ce nouveau postulat serait équivalent à minimiser l'impact de la contraction sur le réseau.

Une ouverture possible consisterait à répondre à la question : existe-t-il une algèbre des relations qui permette de trouver un opérateur de contraction qui satisfasse à la fois le postulat de succès et le postulat de rétablissement ? On pense en particulier à une restriction de cet algèbre de relations pour laquelle seules les relations convexes seraient autorisées. Une telle recherche est très théorique et dépasse le cadre de notre travail.

## 3.3 Evolution des alignements du réseau pour le redressement des postulats

### 3.3.1 Hypothèse supplémentaire

Pour mener notre étude on a fait une hypothèse supplémentaire qui permet de toujours raisonner à partir de l'algèbre des relations. Comme on l'a dit dans la section 3.2.2, cela nous simplifie la tâche puisqu'on a défini dans la section 1.2.2 un ensemble d'opérations sur ces relations, ainsi que la notion de conséquence directe entre deux relations.

- Dans le cas de la révision, on fait l'hypothèse que les ensembles minimaux de formules inconsistants peuvent être exprimés de manière équivalente par une relation composite égale à la relation vide  $\emptyset$ .
- Dans le cas de la contraction, on fait l'hypothèse que la contraction de formules est équivalente à un appauvrissement d'une certaine relation composite.

L'expression relation composite fait référence à une relation construite à partir de plusieurs relations en utilisant différents opérateurs de l'algèbre de relations. Ces deux hypothèses sont intimement liées. En effet, consolider une relation composite correspond à y contracter la relation vide  $\emptyset$ .

Par exemple, si on considère l'algèbre de relations dans OWL 2 EL, les axiomes de classe autorisés sont *SubClassOf*( $A, B$ ), *EquivalentClasses*( $A, B$ ), et *DisjointClasses*( $A, B$ ). Dans l'algèbre des relations, ces relations correspondent à  $A \{<, =\} B$ ,  $A \{=\} B$ , et  $A \{\perp\} B$  respectivement. Ces relations peuvent être utilisées seules, ou conjointement à l'aide de l'opérateur de conjonction des classes, traduit dans l'algèbre des relations par l'opération d'intersection des relations. La disjonction des classes est autorisée à partir de OWL 2 RL.

### 3.3.2 Choix d'un critère de minimalité

Informellement, on souhaite qu'après évolution de l'une des ontologies, l'alignement évolue de manière minimale. Puisqu'on ne peut qu'ajouter ou retirer des relations élémentaires dans des relations des alignements, alors on va proposer une métrique relative au nombre de relations élémentaires ajoutées.

On a vu qu'il est toujours possible de consolider un réseau d'ontologies localement consistantes en faisant évoluer les alignements. Une solution grossière consiste à supprimer tous les alignements. Plus précisément cette solution consiste pour chaque correspondance de l'alignement normalisé, à ajouter autant de relation élémentaires que nécessaire jusqu'à ce que la relation de la correspondance soit la relation générique  $\mathcal{E}$  : "je ne sais pas". Suivant la définition 21 de la distance entre deux relations lorsque l'une est incluse dans l'autre, on comprend pourquoi on peut dire que cette solution est la plus grossière des solutions de consolidation.

On devra donc dans tous les cas ajouter un certain nombre de relations élémentaires dans les alignements, tout en minimisant une notion de distance entre les alignements. Connaissant la distance entre deux relations, on peut proposer plusieurs mesures de distance entre les alignements, comme par exemple la somme des distances entre les relations des correspondances, ou la somme des carrés des distances entre les relations des correspondances. C'est la première que l'on choisira ici.

**Definition 38** (Distance entre des alignements). *La distance entre des alignements est égale à la somme des distances entre les relations des correspondances des alignements*

Pour que cette définition soit valide, l'ensemble des correspondances doit être fini. En théorie et suivant les logiques cette condition peut ne pas être satisfaite, mais pour les cas réels, on aura toujours un ensemble fini de correspondances.

### 3.3.3 Reformulation du problème basée sur l'algèbre des relations

Forts des hypothèses supplémentaires et de la définition d'une distance entre des alignements, on se replace dans le cadre de l'étude préliminaire décrit au début de la section 3.2.

Un réseau normalisé et consistant d'ontologies  $S = \langle \Omega, \Lambda \rangle$ , formé d'une famille d'ontologies  $\Omega = (o_i)_{i \in I}$  et d'une famille d'alignements normalisés  $\Lambda = (A_{ij}^*)_{(i,j) \in I^2}$ , subit une évolution locale  $\otimes$  sur l'ontologie  $o_i$  par l'ensemble de formules  $X_i$ . Soit  $\langle X, Y \rangle$  tel que  $X = (X_k)_{k \in I}$  où  $X_k = \emptyset \forall k \neq i$ ,  $i$  est l'index de l'ontologie contractée, et  $Y = (Y_{kl})_{(k,l) \in I^2, k \neq l}$  où  $Y_{kl} = \emptyset \forall (k,l) \in I^2, k \neq l$ . Soit également  $\Omega' = \{(o_j)_{j \in I, j < i}, o_i \otimes X_i, (o_j)_{j \in I, j > i}\}$ . On cherche une opération de redressement des postulats qui transforme  $\Lambda$  en  $\Lambda'$  de façon minimale, et de sorte que  $S' = \langle \Omega, \Lambda \rangle \boxtimes \langle X, Y \rangle = Cn(\langle \Omega', \Lambda' \rangle)$  soit un produit valide de l'évolution globale du réseau par  $\langle X, Y \rangle$ .

L'hypothèse supplémentaire décrite ci-dessus nous permet de reformuler le problème par un ensemble minimal de  $n$  conditions sur des relations entre éléments :  $\forall i \in \llbracket 1, n \rrbracket, S' \not\models e_i R_i e'_i$ . Dans le cas de la consolidation, chaque relation  $R_i$  est l'ensemble vide  $\emptyset$ . Reformulées dans l'algèbre des relations, chacune de ces conditions se note :  $R'_i \not\models R_i$ , où  $R'_i$  est la relation qui lie les éléments  $e_i$  et  $e'_i$  après modification des alignements. L'ensemble de ces conditions s'écrit donc de manière équivalente :  $\forall i \in \llbracket 1, n \rrbracket, R_i \not\models R'_i$ . Pour résoudre le problème de façon minimale suivant notre métrique, on doit ajouter le minimum de relations élémentaires dans les correspondances des alignements de sorte que ces conditions soient satisfaites.

### 3.3.4 Evaluation d'un algorithme simple : complexité et coût de l'opération

Soit  $N = \text{card}(I)$  le nombre d'ontologies dans le réseau ;  $n_k$  le nombre d'éléments dans l'ontologie  $k$ , et  $n = \max_k(n_k)$  le maximum du nombre d'éléments dans les ontologies.

Pour chaque condition  $S' \not\models e_i R_i e'_i$ , les éléments  $e$  et  $e'$  appartiennent à l'ontologie qui a évolué. La relation  $R'_i$  de la condition  $i$  est égale à l'intersection pour tous les chemins qui lient les éléments  $e_i$  et  $e'_i$ , des compositions des relations de ces chemins.

$$R'_i = \bigcap_{k \in I_{\text{chemins}}} \bullet_{l \in I_{\text{chemin } k}} R'_{kl}$$

On identifie deux catégories de chemins entre  $e_i$  et  $e'_i$  :

- Les chemins *intra*, qui ne sortent pas de l'ontologie. La relation résultante de ces chemins est notée  $R'_{\text{intra}}$  ;
- Les chemins *inter*, qui sortent de l'ontologie – et qui doivent y rentrer à nouveau –. La relation résultante de ces chemins est notée  $R'_{\text{inter}}$ .

La relation  $R'$  résultante est l'intersection de ces deux relations  $R' = R'_{\text{intra}} \cap R'_{\text{inter}}$ .

Etant donné que les postulats de l'opération locale d'évolution sont satisfaits, la première catégorie ne pose pas de problème. Il existe donc au moins une relation  $r$  qui est dans  $R'_{\text{intra}}$  et qui n'est pas dans  $R$ . On doit retrouver au moins une de ces relations élémentaires dans la relation  $R'_{\text{inter}}$ .

Faisons l'hypothèse démesurée que le graphe de chaque ontologie est complet, i.e., chaque couple d'éléments de l'ontologie est lié par une relation dans l'ontologie. Un chemin peut partir de l'ontologie  $i$  par n'importe quelle correspondance, et rentrer par n'importe quelle correspondance. Sous cette hypothèse, chaque correspondance de chaque alignement qui lie l'ontologie  $i$  à une autre ontologie du réseau est donc potentiellement impliquée dans  $R'_{\text{inter}}$ . Or, il y a au maximum  $n_i \times \sum_{j \neq i} n_j < (N - 1) \times n^2$  de ces relations.

D'autre part, on remarque dans le tableau de composition des relations élémentaires 1.2 que les relations élémentaires  $\emptyset$  et  $\perp$  se conservent par composition des relations à droite comme à gauche. De plus,  $\emptyset \cdot \emptyset = \mathcal{E}$  et  $\perp \cdot \perp = \mathcal{E}$ .

Ainsi, si on ajoute une de ces relation élémentaire – toujours la même, par exemple  $\emptyset$  – à chaque relation de correspondance de chaque alignement qui lie l'ontologie qui vient d'évoluer à une autre ontologie du réseau, alors chaque relation des chemins *inter* vaudra  $\mathcal{E}$ . Par cette méthode, on satisfait l'ensemble des conditions d'un seul coup. Le coût total d'une telle opération est majoré par  $N \times n^2$ . Ce coût est également la complexité de l'algorithme simple de résolution de ce problème, puisqu'il se contenterait d'ajouter  $\emptyset$  à chaque relation de correspondance de chaque alignement qui lie l'ontologie qui vient d'évoluer à une autre ontologie du réseau.



# Conclusion

**Cadre de l'étude** Les ontologies sont des outils communément développés et utilisés à des fins de représentation des connaissances par des groupes de recherche et des entreprises. L'alignement d'ontologies est une pratique qui devient nécessaire dès que différentes ontologies co-existent et couvrent partiellement les mêmes domaines. L'ensemble des ontologies et des alignements qui les interconnectent forme aujourd'hui un vaste réseau de connaissances distribuées.

Après avoir présenté succinctement le cadre des structures logiques de Tarski, nous avons rappelé les syntaxe et sémantique des ontologies en logique de description, ainsi que celles des alignements d'ontologies. Nous avons également présenté une algèbre des relations entre ensembles sur laquelle nous avons proposé de baser les relations des correspondances des alignements entre ontologies. L'utilisation de cette algèbre des relations nous a permis de définir des opérations utiles sur les alignements (normalisation, conjonction, disjonction, composition).

Nous avons décrit une sémantique propre à un réseau d'ontologies sous le modèle de la sémantique distribuée intégrée. Les alignements entre ontologies agissent comme un filtre sur les  $n$ -uplets de modèles des ontologies pour conserver ceux qui leur sont compatibles. Il est ainsi possible de déduire de nouvelles conséquences :

- dans les alignements – clôture alpha du réseau ;
- dans les ontologies – clôture oméga du réseau .

Nous avons également défini les opérations de normalisation, d'union et d'intersection pour les réseaux d'ontologies, ainsi que la relation d'inclusion d'un réseau d'ontologie dans un autre. Cela nous a servi pour la suite de notre travail.

**Motivations** L'alignement est une tâche coûteuse et les ontologies sont des objets dynamiques en constante évolution. Le premier objectif de ce travail était donc d'étudier dans quelle mesure il est envisageable de réutiliser un alignement rendu a priori obsolète à cause d'une évolution dans l'une des ontologies qu'il lie. Nous avons étendu cet objectif aux réseaux d'ontologies, en considérant qu'une évolution dans une ontologie devait se propager à l'ensemble des alignements du réseau.

Nous avons développé nos investigations sur deux fronts :

- **Une approche syntaxique** – Pour laquelle on utilise les informations disponibles à propos des modifications qu'a subit une ontologie alignée ;
- **Une approche sémantique** – Pour laquelle on considère que l'évolution dans l'ontologie a été menée par le biais d'opérateurs qui satisfont certains postulats de rationalité.

**Résultats pour l'approche syntaxique** L'approche syntaxique est basée sur la notion de modifications des ontologies. Nous avons introduit les domaines concernés : l'évolution et le versionnage d'ontologies. Nous considérons pour cette approche que les ontologies sont modifiées manuellement ou semi-automatiquement à l'aide d'éditeurs d'ontologies.

Nous avons donc choisi d'utiliser un outils de versionnage d'ontologies nommé  $P \rightarrow D$ , qui décrit explicitement les différences structurelles entre deux versions d'ontologies. Pour chaque élément d'ontologie, cinq modifications élémentaires peuvent être décrites dans une table  $P \rightarrow D$  : *ajout* ; *suppression* ; *fusion* ; *division* ; *correspondance*.

A partir d'une table  $P \rightarrow D$ , nous avons développé un premier algorithme très simple et intuitif de propagation des alignements d'une version à une autre. Nous avons montré que la fusion d'éléments pouvait poser problème, car la conjonction des relations des correspondances qu'on utilise est génératrice d'incohérence dans les alignements. Nous avons donc proposé une nouvelle version de l'algorithme de propagation des alignements qui palie ce problème particulier.

Nous avons mis en évidence deux limites inhérentes à la méthode :

- il n'est pas question a priori que le réseau d'ontologies soit consistant absolument après l'évolution ;
- une formule supprimée dans une ontologie risque d'être encore inférable dans le réseau.

Une autre limite vient de l'utilisation de l'outil P D qui ne prend pas en compte les éléments composites des ontologies.

**Résultats pour l'approche sémantique** L'approche sémantique est basée sur le domaine du changement de croyances, initié par Alchourrón, Gärdenfors, et Makinson en 1985. Nous nous sommes inspirés de travaux récents de la recherche et avons repris et adapté les postulats de rationalité que doivent satisfaire des opérateurs de révision (on croit désormais qu'une formule est vraie) et de contraction (on croit désormais qu'une formule n'est plus vraie). Notre adaptation permet d'ajouter ou de retirer plusieurs formules d'un coup, et nous avons adapté ces postulats aux réseaux d'ontologies.

Nous avons évalué l'impact d'une évolution locale sur le réseau : Si on considère qu'une opération d'évolution du réseau ne modifie qu'une ontologie de façon locale et rationnelle, nous avons montré que certains postulats de rationalité de l'opérateur global correspondant peuvent ne pas être satisfaits. En particulier :

- lors d'une révision locale par un ensemble de formules, la consistance du réseau entier peut être compromise ;
- lors d'une contraction locale par un ensemble de formules, les formules risquent d'être encore des conséquences du réseau tout entier.

On reconnaît là les deux problèmes identifiés comme inhérents à la méthode développée dans l'approche syntaxique.

La suite de nos travaux avait donc pour objectif de proposer une évolution des alignements de sorte de redresser ces postulats de rationalité. Nous avons d'abord montré qu'il est toujours possible de redresser ces postulats en appliquant la solution drastique qui consiste à vider de leurs correspondances chacun des alignements du réseau d'ontologies. On peut dire que par cette solution on maximise le nombre de modèles du réseau d'ontologies (en contrepartie on minimise le nombre de conséquences inférables).

Cependant, même si dans le cas de la contraction on essaie de limiter l'augmentation de l'ensemble des modèles nécessaire afin de ne plus pouvoir inférer les formules contractées localement, on a montré qu'on risque d'insatisfaire un autre postulat dit de rétablissement : "si on ajoute à nouveau ce qu'on a contracté, on doit retrouver au moins les croyances de départ". Nous avons justement identifié un exemple de réseau et de contraction locale pour lesquels il est impossible de satisfaire ces deux postulats à la fois, pour l'algèbre des relations qu'on utilise. Nous avons donc relaxé ce deuxième postulat dans la fin de notre travail au profit du postulat de succès.

Finalement, nous avons formalisé une unification des problèmes de redressement des postulats de rationalité dans le cas de la révision et de la contraction, en utilisant l'algèbre des relations. Nous avons défini une fonction coût à minimiser lors du processus, qui est une mesure de la distance entre des alignements du réseau avant et après évolution. Nous avons enfin proposé un algorithme simple de redressement des postulats et avons cherché un majorant du coût de l'opération.

**Bilan** A travers l'ensemble de ces travaux, nous avons donc effectivement rendu compte des problèmes qui peuvent survenir lorsqu'une ontologie vient à évoluer dans un réseau d'ontologies. Suivant le type d'évolution que cette ontologie subit, nous avons apporté des éléments de réponse pour savoir comment adapter les alignements à cette évolution.

Nous avons montré enfin que l'approche syntaxique présente au moins deux problèmes qui ne se posent pas dans l'approche sémantique car ces problèmes réfèrent à des postulats que tout opérateur d'évolution valide du réseau doit satisfaire. L'approche sémantique n'est cependant pas encore pour aujourd'hui puisque les logiciels de gestion d'ontologies ne proposent pas encore d'opérateurs d'évolution automatique des ontologies.

# Bibliography

- [1] C.E. Alchourrón, P. Gärdenfors, and D. Makinson. On the logic of theory change: Partial meet contraction and revision functions. *Journal of symbolic logic*, pages 510–530, 1985.
- [2] T. Berners-Lee. Semantic web road map, 1998. <http://www.w3.org/DesignIssues/Semantic.html>.
- [3] T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, May 2001.
- [4] A. Borgida and L. Serafini. Distributed description logics: Assimilating information from peer sources. *Journal on Data Semantics*, pages 153–184, 2003.
- [5] P. Bouquet, F. Giunchiglia, F. Harmelen, L. Serafini, and H. Stuckenschmidt. C-owl: Contextualizing ontologies. *The SemanticWeb-ISWC 2003*, pages 164–179, 2003.
- [6] B. Chen, H. Tan, and P. Lambrix. Structure-based filtering for ontology alignment. *Enabling Technologies, IEEE International Workshops on*, 0:364–369, 2006.
- [7] J. Euzenat. Algebras of ontology alignment relations. pages 387–402. 2008.
- [8] J. Euzenat, A. Mocan, and F. Scharffe. Ontology Alignments—An Ontology Management Perspective. *Martin Hepp, Pieter De Leenheer, Aldo de Moor et York Sure, éditeurs: Ontology Management*, 7:177–206, 2008.
- [9] J. Euzenat and P. Shvaiko. *Ontology matching*. Springer-Verlag, Heidelberg (DE), 2007.
- [10] G. Flouris. *On Belief Change and Ontology Evolution*. PhD thesis, University of Crete, Department of Computer Science, 2006.
- [11] C. Ghidini and L. Serafini. Distributed first order logics. *Frontiers of Combining Systems 2, Studies in Logic and Computation*, 1998.
- [12] C. Ghidini and L. Serafini. Distributed first order logics. *Frontiers of Combining Systems*, 2:121–139, 2000.
- [13] T.R. Gruber et al. A translation approach to portable ontology specifications. *Knowledge acquisition*, 5:199–199, 1993.
- [14] S.O. Hansson. A test battery for rational database updating. *Artificial intelligence*, 82(1-2):341–352, 1996.
- [15] M. Klein and N. Noy. A component-based framework for ontology evolution. In *Workshop on Ontologies and Distributed Systems at IJCAI*, 2003.
- [16] P. Lambrix, H. Tan, and Q. Liu. SAMBO and SAMBOdtf results for the ontology alignment evaluation initiative 2008. In *The 7th International Semantic Web Conference*, page 190, 2008.
- [17] C. Meilicke and H. Stuckenschmidt. An efficient method for computing alignment diagnoses. *Web Reasoning and Rule Systems*, pages 182–196, 2009.
- [18] N. Noy and M. Musen. PROMPTDIFF: A Fixed-Point Algorithm for Comparing Ontology Versions. In *in Eighteenth National Conference on Artificial Intelligence (AAAI-2002)*, pages 744–750, 2002.
- [19] R. Palma, P. Haase, Ó. Corcho, and A. Gómez-Pérez. Change representation for owl 2 ontologies. In Rinke Hoekstra and Peter F. Patel-Schneider, editors, *OWLED*, volume 529 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2008.
- [20] L. Stojanovic, A. Maedche, B. Motik, and N. Stojanovic. User-driven ontology evolution management. *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web*, pages 133–140, 2002.
- [21] A. Tarski. On the calculus of relations. *Journal of Symbolic Logic*, 6(3):73–89, 1941.

- 
- [22] A. Zimmermann. *Sémantique des réseaux de connaissances : gestion de l'hétérogénéité fondée sur le principe de médiation*. Thèse d'informatique, Université Joseph Fourier, Grenoble (France), November 2008.
- [23] A. Zimmermann and J. Euzenat. Three semantics for distributed systems and their relations with alignment composition. *Lecture Notes in Computer Science*, 4273:16, 2006.