



HAL
open science

Inconsistency and Uncertainty Handling in Lightweight Description Logics

Zied Bouraoui

► **To cite this version:**

Zied Bouraoui. Inconsistency and Uncertainty Handling in Lightweight Description Logics. Computer Science [cs]. Université d'Artois, 2015. English. NNT : . tel-02876684

HAL Id: tel-02876684

<https://hal.science/tel-02876684>

Submitted on 21 Jun 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Inconsistency and uncertainty handling in lightweight description logics

THÈSE

présentée et soutenue publiquement le 5 juin 2015

en vue de l'obtention du

Doctorat de l'Université d'Artois
(Spécialité Informatique)

par :

Zied Bouraoui

devant le jury composé de

Anthony HUNTER	Professor - University College London, United Kingdom	(Rapporteur)
Henri PRADE	Directeur de recherche CNRS - IRIT Toulouse	(Rapporteur)
Salem BENFERHAT	Professeur des universités - Université d'Artois	(Directeur de Thèse)
Sylvain LAGRUE	Maître de conférences (HDR) - Université d'Artois	(Examineur)
Marie-Laure MUGNIER	Professeur des universités - Université de Montpellier II	(Examinatrice)
Odile PAPINI	Professeur des universités - Université d'Aix Marseille	(Examinatrice)
Ramón PINO PÉREZ	Professor - University of the Andes, Venezuela	(Examineur)
Karim TABIA	Maître de conférences - Université d'Artois	(Examineur)

CENTRE DE RECHERCHE EN INFORMATIQUE DE LENS – CNRS UMR-8188

UNIVERSITÉ D'ARTOIS, RUE JEAN SOUVRAZ, S.P. 18 F-62307, LENS CEDEX FRANCE

SECRÉTARIAT : TÉL.: +33 (0)3 21 79 17 23 – FAX : +33 (0)3 21 79 17 70

<http://www.cril.univ-artois.fr>

ABSTRACT

Abstract

This thesis investigates the dynamics of beliefs and uncertainty management in *DL-Lite*, one of the most important lightweight description logics. The first part of the thesis concerns the problem of handling uncertainty in *DL-Lite*. First, we propose an extension of the main fragments of *DL-Lite* to deal with the uncertainty associated with axioms using a possibility theory framework without additional extra computational costs. We then study the revision of possibilistic *DL-Lite* bases when a new piece of information is available. Lastly, we propose a min-based assertional merging operator when assertions of ABox are provided by several sources of information having different levels of priority. The second part of the thesis concerns the problem of inconsistency handling in flat and prioritized *DL-Lite* knowledge bases. We first propose how to reason from a flat *DL-Lite* knowledge base, with a multiple ABox, which can be either issued from multiple information sources or resulted from revising *DL-Lite* knowledge bases. This is done by introducing the notions of modifiers and inference strategies. The combination of modifiers plus inference strategies can be mapped out in order to provide a principled and exhaustive list of techniques for inconsistency management. We then give an approach based on selecting multiple repairs using a cardinality-based criterion, and we identified suitable strategies for handling inconsistency in the prioritized case. Lastly, we perform a comparative analysis, followed by experimental studies, of the proposed inconsistency handling techniques. A tool for representing and reasoning in possibilistic *DL-Lite* framework is implemented.

Résumé

Cette thèse étudie la dynamique des croyances et la gestion de l'incertitude dans *DL-Lite*, une des plus importantes familles des logiques de description légères. La première partie de la thèse porte sur la gestion de l'incertitude dans *DL-Lite*. En premier lieu, nous avons proposé une extension des principaux fragments de *DL-Lite* pour faire face à l'incertitude associée aux axiomes en utilisant le cadre de la théorie des possibilités. Cette extension est réalisée sans engendrer des coûts calculatoires supplémentaires. Nous avons étudié ensuite la révision des bases *DL-Lite* possibilistes en présence d'une nouvelle information. Enfin, nous avons proposé un opérateur de fusion lorsque les assertions de ABox sont fournies par plusieurs sources d'information ayant différents niveaux de priorité. La deuxième partie de la thèse traite le problème de la gestion d'incohérence dans les bases de connaissances *DL-Lite*. Nous avons étudié, tout d'abord, comment raisonner à partir d'une base *DL-Lite* standard avec des ABox multiples en introduisant les notions de modificateurs et de stratégies d'inférence. La combinaison des modificateurs et de stratégies d'inférence fournit une liste exhaustive des principales techniques de gestion de l'incohérence. Nous avons proposé ensuite une approche, basée sur un critère de cardinalité, de sélection des réparations, et nous avons identifié les stratégies appropriées pour la gestion de l'incohérence pour les bases *DL-Lite* stratifiées. Enfin, nous avons effectué une analyse comparative, suivie par des études expérimentales, des différentes techniques de gestion d'incohérence proposées. Finalement, un outil de représentation et de raisonnement à partir des bases *DL-Lite* possibiliste est réalisé.

CONTENTS

Introduction	1
Context and motivations	1
Contributions	3
Organization of the thesis	5
I Preliminaries	7
1 Knowledge representation and ontologies	9
1.1 Introduction	9
1.2 Ontology languages	10
1.3 Logic-based languages	13
1.3.1 Propositional logic	13
1.3.2 First order logic	15
1.3.3 Description logics	16
1.4 The <i>DL-Lite</i> family	22
1.4.1 The <i>DL-Lite</i> family and <i>OWL2-QL</i>	22
1.4.2 The extended <i>DL-Lite</i> family	29
1.5 Conclusion	31
2 Belief change and uncertainty management	33
2.1 Introduction	33
2.2 Uncertainty management	34
2.2.1 Probability theory	36
2.2.2 Possibility theory	37
2.2.3 Uncertainty management in description logics	45
2.3 Belief change	47
2.3.1 Belief revision	47
2.3.2 Belief merging	50
2.3.3 Inconsistency handling	55
2.4 Conclusion	56
II On the possibilistic extension of <i>DL-Lite</i>	57
3 Min-based possibilistic <i>DL-Lite</i>	59
3.1 Introduction	59
3.2 Possibility distribution over <i>DL-Lite</i> interpretations	60
3.2.1 Possibility distribution	60

3.2.2	Possibility and necessity measures	60
3.3	Possibilistic $DL-Lite_{core}$	62
3.3.1	Syntax of $\pi-DL-Lite_{core}$	62
3.3.2	From a $\pi-DL-Lite_{core}$ knowledge base to a $\pi-DL-Lite_{core}$ possibility distribution	63
3.3.3	Logical properties of $\pi-DL-Lite_{core}$	64
3.4	Possibilistic negated closure in $\pi-DL-Lite_{core}$	65
3.4.1	Rules used to obtain $\pi-negated$ closure	65
3.4.2	Properties of $\pi-negated$ closure	67
3.5	Checking inconsistency degrees	68
3.5.1	Additional properties of $\pi-neg(\mathcal{T})$	68
3.5.2	Computing inconsistency degrees in $\pi-DL-Lite_{core}$	69
3.5.3	An algorithm for computing inconsistency degrees	73
3.6	Possibilistic $DL-Lite_F$ and possibilistic $DL-Lite_R$	74
3.6.1	$\pi-DL-Lite_F$ negated closure	75
3.6.2	$\pi-DL-Lite_R$ negated closure	76
3.7	Basic inferences in $\pi-DL-Lite_{core}$	77
3.8	Query answering in possibilistic DL-Lite	81
3.9	Discussions and related works	85
3.10	Conclusion	86
4	Min-based conditioning and merging approach of $DL-Lite$ knowledge bases	87
4.1	Introduction	87
4.2	Min-based merging of $\pi-DL-Lite$ knowledge bases	88
4.2.1	Merging of $\pi-DL-Lite$ possibility distributions	88
4.2.2	Syntactical merging of $\pi-DL-Lite$ knowledge bases	90
4.3	Min-based assertional merging approach for $\pi-DL-Lite$ knowledge bases	91
4.3.1	Syntactic merging of $\pi-DL-Lite$ assertional bases	91
4.3.2	Semantic counterpart	95
4.3.3	Logical properties	97
4.4	Conditioning of possibilistic $DL-Lite$ knowledge bases: Preliminary results	98
4.4.1	Conditioning of $DL-Lite$ possibility distributions	99
4.4.2	Syntactic revision	102
4.5	Conclusion	108
III	Inconsistency handling in flat $DL-Lite$ knowledge bases	109
5	Non-merge inconsistency management roadmap in flat $DL-Lite$ knowledge bases	111
5.1	Introduction	111
5.2	Reasoning from MBox knowledge bases	112
5.2.1	MBox: Multiple ABox	112
5.2.2	Elementary modifiers on MBox	112
5.2.3	Composite modifiers on MBox	115
5.3	Inference-based strategies from MBox	117

5.3.1	Universal inference	117
5.3.2	Existential inference	118
5.3.3	Safe inference	119
5.3.4	Other inferences	119
5.3.5	Comparing inference-based strategies from a fixed MBox	120
5.4	Handling inconsistency=Composite modifiers+inference strategies	122
5.5	Comparative studies	124
5.5.1	Productivity	124
5.5.2	Complexity analysis	126
5.6	Related works and discussions	130
5.6.1	Related works	130
5.6.2	Majority-based inference within <i>DL-Lite</i> framework	131
5.6.3	Which inference relations are appropriate to deal with inconsistent <i>DL-Lite</i> knowledge bases?	131
5.7	Conclusion	132
IV	On the revision of prioritized <i>DL-Lite</i> knowledge bases	133
6	Prioritized Assertional-Based Removed Sets Revision of <i>DL-Lite</i> knowledge Bases	135
6.1	Introduction	135
6.2	Assertional-based revision of <i>DL-Lite</i> knowledge bases	136
6.2.1	Prioritized <i>DL-Lite</i> assertional base	137
6.2.2	Prioritized Removed Sets Revision of <i>DL-Lite</i> knowledge bases	137
6.2.3	Examples of selection functions	141
6.2.4	Multiple revision	142
6.3	Logical properties	143
6.3.1	Hansson's postulates reformulated	144
6.3.2	Prioritized Removed Sets Revision: logical properties	144
6.4	Computing the revision operation outcome	145
6.4.1	Computing the conflicts	146
6.4.2	Computing the PRSR outcome	147
6.5	Discussions and related works	150
6.6	Conclusion	151
7	How to select one preferred assertional-based repair from inconsistent and prioritized <i>DL-Lite</i> knowledge bases?	153
7.1	Introduction	153
7.2	Existing assertional-based preferred repairs	154
7.2.1	Preferred inclusion-based repair	155
7.2.2	Lexicographic preferred-based repair	156
7.2.3	Possibilistic-based repair	157
7.2.4	Linear-based repair	157
7.3	Sensitivity to the prioritized closure	158

7.4	New strategies for selecting one preferred repair	159
7.4.1	Non-defeated repair	159
7.4.2	Adding the deduction closure	161
7.4.3	Combining linear entailment and non-defeated entailment: Adding consistency .	161
7.4.4	Introducing cardinality in non-defeated inference	162
7.4.5	Adding priorities to non-defeated inference	163
7.5	Comparative analysis	164
7.6	Experimental evaluation	165
7.7	Conclusion	166
Conclusion		167
Appendix		173
A Appendix of Chapter 3		173
B Appendix of Chapter 5		177
References		195

GENERAL INTRODUCTION

Context and motivations

In the last years, there is a growing use of ontologies in many application areas. Description Logics (DLs for short), mostly based on first order logic, are recognized as powerful formal frameworks for representing and reasoning on ontologies. A DL knowledge base is built upon two distinct components: A terminological base (called *TBox*), representing generic structural knowledge about an application domain, and an assertional base (called *ABox*), containing the assertional facts (*i.e.* individuals or constants) that instantiate the generic knowledge. In the context of Semantic Web, DLs provide the logical basis of the Web Ontology Language (OWL), standardized by the W3C.

In many applications, ontologies are generally very large and some reasoning tasks can be prohibitive. There exist several description languages where the majority of them are intractable (*e.g.* *SHOIQ*, *SROIQ*), in the sense that they do not guarantee a polynomial complexity when reasoning. To this end, several lightweight DLs (*e.g.* \mathcal{EL} [Baader *et al.*, 2005a], *DL-Lite* [Calvanese *et al.*, 2005]), mainly motivated by applications (like the ones involving large ontological knowledge), have been proposed. In particular, these logics allow a flexible representation of knowledge with a tractable computational complexity of the reasoning process.

DL-Lite [Calvanese *et al.*, 2005] is a family of tractable DLs specifically dedicated to applications that use large volumes of data where query answering is the most important reasoning task. The consistency checking problem and all standard reasoning tasks are polynomial with respect to the size of the assertional base [Calvanese *et al.*, 2007a]. In these logics, an important reasoning task is the one of answering complex queries (especially conjunctive queries) where reasoning complexity is in LogSpace for data complexity (namely, the size of the data) [Artale *et al.*, 2009]. This fact makes *DL-Lite* especially well-suited for the context of Ontology-Based Data Access (OBDA), which studies how to query a set of data sources using an unified generic (ontological) view. In such settings, the terminological base acts as a schema used to reformulate the queries in order to offer a better access to the set of data stored in several assertional bases [Poggi *et al.*, 2008]. A crucially important problem that arises in the OBDA setting is how to handle efficiently the multiple data sources.

In this context, assertions are often provided by several and potentially conflicting sources having different reliability levels. Moreover, a given source may provide different sets of uncertain assertions with different confidence levels. Gathering such sets of assertions gives a prioritized or a stratified assertional base. This stratification generally results from two situations as pointed out in several research papers (*e.g.* [Baral *et al.*, 1992; Benferhat *et al.*, 1995; Benferhat *et al.*, 1998b]).

- The first one is when each source provides its set of data without any priority between them, but there exists a *total pre-ordering* between different sources reflecting their *reliability*.
- The other one is when the sources are considered as equally reliable (*i.e.* having the same reliability level), but there exists a preference ranking between the set of provided data according to their *level of certainty*.

The standard *DL-Lite* framework does not offer means of taking advantage of priority or uncertainty in the knowledge. In [Dubois and Prade, 1991a], it is argued that handling priority/uncertainty is in a complete agreement with possibility theory [Dubois and Prade, 1988b]. This latter offers a very natural framework to deal with ordinal, qualitative uncertainty, preferences and priorities. This framework

is particularly appropriate when the uncertainty/priority scale only reflects a priority relation between different pieces of information. Recently, several works have been proposed to deal with probabilistic and non-probabilistic uncertainty [Dubois *et al.*, 2006; Lukasiewicz *et al.*, 2012a] on the one hand and to deal with fuzzy information [Bobillo and Straccia, 2012; Lukasiewicz and Straccia, 2009] on the other hand. A particular attention was given to fuzzy extensions of DLs (e.g. [Bobillo *et al.*, 2012; Bobillo *et al.*, 2013; Pan *et al.*, 2007; Straccia, 2006b; Straccia, 2013]). Besides, some works are devoted to possibilistic extensions of description logics (e.g. [Dubois *et al.*, 2006; Hollunder, 1995; Qi *et al.*, 2011]) which are basically based on standard reasoning services. This thesis fills this gap and proposes an extension of *DL-Lite* within a possibility theory setting.

In *DL-Lite* and OBDA settings, inconsistency and contradictions are always defined with respect to some assertions that contradict the terminology. Indeed, a *DL-Lite* terminology may be incoherent but never inconsistent. Faced to inconsistency, there are two main attitudes:

- i) The first one consists in *merging* (e.g. [Kotis *et al.*, 2006; Moguillansky and Falappa, 2007]) the knowledge base using some aggregation strategies. Knowledge bases merging or belief merging (e.g. [Bloch *et al.*, 2001; Konieczny and Pino Pérez, 2002]), is a problem largely studied within the propositional logic setting. It focuses on aggregating pieces of information issued from distinct, and may be conflicting or inconsistent, sources in order to obtain a unified point of view by taking advantage of pieces of information provided by each source. Generally in OBDA setting, applying merging techniques on data sources lead to removing some assertions that contradict the terminological base (which may be seen as an integrity constraint) in order to restore consistency. This approach is too cautious since it comes down to throw out an important part of the data which becomes no longer useful when reasoning. In ontology area, there are few works which studied the application of merging techniques proposed within propositional logics to merge DL knowledge bases. The existing works mainly reduce the merging problem to an inconsistency handling one.
- ii) The second attitude consists in accepting and leaving inconsistency while coping with it when performing inference using different inconsistency-tolerant inference strategies. Handling inconsistency is also a problem largely studied within the propositional logic setting [Bertossi *et al.*, 2005b]. Several approaches were proposed to deal with inconsistency in propositional logic knowledge bases. These approaches focus either on restoring consistency (e.g. [Benferhat *et al.*, 1997a]), using paraconsistent logics (e.g. [Hunter, 1998]), analyzing and measuring the inconsistency (e.g. [Hunter and Konieczny, 2005]), employing argumentation framework (e.g. [Besnard and Hunter, 2008]). In the same spirit, several works (e.g. [Qi and Hunter, 2007; Corcho *et al.*, 2009; Ma and Hitzler, 2010; Lukasiewicz *et al.*, 2013]) were proposed to handle inconsistency or incoherency in ontologies (Ontology debugging or repairing). Regarding *DL-Lite*, in the context of OBDA, existing works (e.g. [Lembo *et al.*, 2010; Bienvenu, 2012; Bienvenu and Rosati, 2013]), basically inspired by the approaches proposed in the database area, tried to deal with inconsistency in *DL-Lite* by proposing and adapting several inconsistency-tolerant inference methods. All the proposed approaches are based on the notion of repair which is closely related to the notion of database repair defined in order to answer queries raised to inconsistent databases. A repair of a database contradicting a set of integrity constraints is a database obtained by applying a minimal set of changes in order to restore consistency. This notion was extended to the *DL-Lite* setting [Lembo *et al.*, 2010] by defining assertional-based repair for *DL-Lite* knowledge bases which is simply a maximal assertional subbase consistent with the *TBox*. Clearly, these works are closely related to works on restoring consistency, proposed for handling inconsistency in propositional logic knowledge bases.

In this thesis, we study merging of different data sources linked to the same ontological view, seen as integrity constraints. Besides, we more investigate inconsistency handling for *DL-Lite* in case where the

knowledge base is flat or prioritized.

Another problem addressed in this thesis is the one of ontology dynamics. In fact, description logics have been proposed to represent the static knowledge of a domain of interest. However, knowledge may be non static and may evolve and change from one situation to another in order to take into account and integrate the changes that occur over time. One of the fundamental issues in Web applications is the dynamics of the knowledge base (e.g. [Qi *et al.*, 2006c; Wang *et al.*, 2010; Kharlamov *et al.*, 2013]) which is a problem closely related to the belief revision one (e.g. [Alchourrón *et al.*, 1985; Katsuno and Mendelzon, 1991; Hansson, 1998]). Several approaches were proposed by adapting revision operators proposed within propositional logic setting to description logics ones. There are two main approaches for revision: Model-based approaches (e.g. [Qi and Du, 2009; Wang *et al.*, 2010]) or formula-based approaches (e.g. [Halaschek-wiener *et al.*, 2006; Ribeiro and Wassermann, 2007]). As pointed out in [Calvanese *et al.*, 2010], model-based approaches of revision are in general not appropriate for *DL-Lite* in the sense that the result of the revision is not expressible in the initial *DL-Lite* language (before the revision process). In this thesis, we follow a formula-based approach for revising *DL-Lite* knowledge bases when the ABox is prioritized.

This thesis investigates the dynamics of knowledge and beliefs and uncertainty management in *DL-Lite*, one of the most important lightweight DLs. It contains three main contributions: i) an extension of *DL-Lite* to possibility theory, ii) a proposition for a roadmap for handling inconsistency in flat *DL-Lite* knowledge bases, and iii) an analysis of revision and inconsistency handling in prioritized *DL-Lite* knowledge bases.

Contributions

The first contribution of the thesis concerns the problem of handling uncertainty in *DL-Lite*. In fact, in many contexts, the available information and knowledge may be uncertain or prioritized requiring a framework to manage uncertainty and priorities.

Part I

- **Possibilistic *DL-Lite*:** We investigate an extension of the main fragments of *DL-Lite* to deal with uncertainty associated with objects, concepts or relations using a possibility theory framework. It is particularly useful for handling inconsistent knowledge. We first provide foundations of possibilistic *DL-Lite*, denoted by π -*DL-Lite*, by extending the *DL-Lite_{core}* logic, the core fragment of all *DL-Lite* logics, within a possibility theory setting. We present syntax and semantics of π -*DL-Lite_{core}*, study the reasoning tasks and show how to compute the inconsistency degree of a π -*DL-Lite_{core}* knowledge base. We then extend our possibilistic approach to *DL-Lite_F* and *DL-Lite_R*, two important fragments of *DL-Lite* family. Finally, we address the problem of query answering over a π -*DL-Lite* knowledge bases. An important result is that the extension of the expressive power of *DL-Lite* is done without additional extra computational costs. A tool for representing and reasoning in π -*DL-Lite* is implemented.
- **Conditioning and merging of possibilistic *DL-Lite* knowledge bases:** We first focus on the use of a minimum-based operator, well known as idempotent conjunctive operator, to combine π -*DL-Lite* possibility distributions and show that this semantic fusion has a natural syntactic counterpart when dealing with π -*DL-Lite* knowledge bases. The min-based fusion operator is recommended when distinct sources that provide information are dependent. We then define a merging operator for π -*DL-Lite* knowledge bases based on conflict resolution. We investigate a *min*-based assertional merging operator. We study in particular the situation where the sources share the generic

knowledge. We present a syntactic method based on conflict resolution which has a meaningful semantic counterpart when merging possibility distributions. We finally provide an analysis in the light of a new set of postulates dedicated to uncertain *DL-Lite* bases merging.

We also give first results on the revision of possibilistic *DL-Lite* knowledge bases when a new input piece of information, possibly conflicting or uncertain, becomes available. We first study revision at the semantic level consisting in directly conditioning possibility distributions. In particular, we show that such conditioning provides in some situations some counterintuitive results compared with the ones of conditioning directly the knowledge base syntactically. We then study revision at the syntactic level of possibilistic *DL-Lite* knowledge bases.

Part II

- **Inconsistency handling in flat *DL-Lite* knowledge bases:** We propose how to reason from a *DL-Lite* knowledge base with a multiple ABox, called an MBox *DL-Lite* knowledge base. An MBox is a multi-set of ABox's which can be either issued from multiple information sources or resulted from revising inconsistent *DL-Lite* knowledge bases. We provide different inference strategies for query answering from MBox *DL-Lite* knowledge bases. We then discuss three main elementary changes or modifiers that can be operated on an MBox : i) Expansion modifiers, ii) Splitting modifiers and iii) Selection-based modifiers. The second part of the chapter uses the concept of MBox to provide a roadmap for handling inconsistent standard *DL-Lite* knowledge bases. We view the problem of repairing an inconsistent *DL-Lite* knowledge base as composed of a composite or a complex modifier on a given MBox followed by an inference-based strategy. Lastly, we show that there are exactly eight major composite modifiers that can be applied on an inconsistent standard *DL-Lite* knowledge base and identify those that produce a single consistent and preferred repair.

Part III

The question addressed in this part is how to revise *DL-Lite* knowledge bases with a new piece of information, in a prioritized setting. Moreover, as highlighted in fusion problems, there is a need to deal with inconsistent prioritized information. It appears that only few works addressed this problem in the *DL-Lite* setting. So, the third part of the thesis concerns the evolution and inconsistency handling of *DL-Lite* knowledge bases when the assertions in the ABox's are prioritized.

- **Assertional-based revision:** We investigate "Prioritized Removed Sets Revision" (PRSR) for revising stratified *DL-Lite* knowledge bases when a new sure piece of information, called the input, is added. The strategy of revision is based on inconsistency minimization and consists in determining the smallest subsets of assertions (prioritized removed sets) that should be dropped from the current stratified knowledge base in order to restore consistency and accept the input. We consider different forms of input: A membership assertion, a positive or a negative inclusion axiom. In some situations, the revision process leads to several possible revised knowledge bases where defining a selection function is required to keep the results within *DL-Lite* fragment. Lastly, we show how to use the notion of hitting set in order to compute the PRSR outcome.
- **Selecting one preferred repair from prioritized *DL-Lite* knowledge bases:** We first review the existing approaches for selecting preferred assertional-based repairs. Then, we focus on suitable strategies for handling inconsistency in *DL-Lite*. We propose, in particular, new approaches based on the selection of one assertional-based repair. These approaches have as a starting point the non-defeated assertional-based repair followed by additional ingredients like the linear-based, cardinality-based, deductive closure, etc. Lastly, we provide a comparative analysis followed by experimental studies of the different studied approaches.

Organization of the thesis

This thesis is organized as follows. We first give in Chapter 1, a refresher on description logics, with a focus on *DL-Lite*. We then give in Chapter 2, an overview about possibility theory, belief change problem in the context of propositional logic and description logics. Chapter 3 presents the extension of *DL-Lite* within a possibility theory setting. Chapter 4 first investigates merging of possibilistic *DL-Lite* and proposes a method based on conflict resolution to aggregate several sets of data linked to the same terminological base. It also gives preliminary results on conditioning in possibilistic *DL-Lite* framework. Chapter 5 provides a non-merging roadmap for inconsistency handling in flat *DL-Lite* knowledge bases. Chapter 6 proposes a lexicographic-based approach for revising stratified *DL-Lite* knowledge bases when a new sure piece of information becomes available. Chapter 7 studies inconsistency handling in *DL-Lite* knowledge bases where the assertional base is prioritized. Finally, the thesis contains a conclusion and some future works. We also provide in the appendix additional material for Chapter 5 and a description of our tool developed for representing and reasoning in possibilistic *DL-Lite* framework.

Publications achieved in this thesis

International journal papers

1. Salem Benferhat and Zied Bouraoui. Min-based possibilistic *DL-Lite*. In the Journal of Logic and Computation (JLC 2015).
2. Salem Benferhat, Zied Bouraoui, Odile Papini, and Eric Würbel. Prioritized Assertional-Based Removed Sets Revision of *DL-Lite* Belief Bases, In the Annals of Mathematics and Artificial Intelligence (Accepted with minor revision).

International conference papers

1. Salem Benferhat, Zied Bouraoui, Karim Tabia. How to select one preferred assertional-based repair from inconsistent and prioritized *DL-Lite* knowledge bases?. In the International Joint Conference on Artificial Intelligence (IJCAI 2015).
2. Salem Benferhat, Zied Bouraoui, Odile Papini, and Eric Würbel. A prioritized assertional-based revision for *DL-Lite* knowledge bases. In the European Conference on Logics in Artificial Intelligence (JELIA 2014).
3. Salem Benferhat, Zied Bouraoui, Sylvain Lagrue, and Julien Rossit. Min-based assertional merging approach for prioritized *DL-Lite* knowledge bases. In the International Conference on Scalable Uncertainty Management (SUM 2014).
4. Salem Benferhat, Zied Bouraoui, Odile Papini, and Eric Würbel. Assertional-based prioritized removed sets revision of *DL-Lite* knowledge bases. In the European Conference on Artificial Intelligence (ECAI 2014).
5. Salem Benferhat, Zied Bouraoui, and Karim Tabia. On the revision of prioritized *DL-Lite* knowledge bases. In the International Conference on Scalable Uncertainty Management (SUM 2014).
6. Salem Benferhat, Zied Bouraoui, Odile Papini and Eric Würbel. Assertional-based removed set revision of *DL-Lite* belief bases. In the International Symposium on Artificial Intelligence and Mathematics (ISAIM 2014).

7. Salem Benferhat and Zied Bouraoui. Possibilistic *DL-Lite*. In International Conference on Scalable Uncertainty Management (SUM 2013).
8. Salem Benferhat, Zied Bouraoui and Zied Loukil. Min-based fusion of possibilistic *DL-Lite* knowledge bases. In the IEEE/WIC/ACM International Conference on Web Intelligence (WI 2013).

PART I

Preliminaries

KNOWLEDGE REPRESENTATION AND ONTOLOGIES

1.1 Introduction

The efficiency of information and knowledge handling is one of the most crucial challenges in many applications such as medicine, biology, economics, etc. This is due to the fact that the volume of knowledge continuously increases while the structure of this latter becomes more and more complex. In fact, knowledge may be provided by multiple, heterogeneous and often conflicting sources of information. A real need to compactly represent and structure this information is required. Information should be faithfully handled, while avoiding confusions, incoherencies, contradictions or ambiguities between elements representing the domain of interest.

During the two last decades, lines of research from both the database and the artificial intelligence communities have focused on complex knowledge representation formalisms. A particular attention was given to the use of ontologies. An ontology provides an explicit and semantically rich framework for representing knowledge [Mika *et al.*, 2004; Mika and Akkermans, 2004]. Ontologies play a crucial role in sharing resources [Torniai *et al.*, 2008] and reasoning about the modeled domain with the ability of checking contradictions.

There exist various languages supporting ontologies such as fragments of first order logic, conceptual graphs (*e.g.* [Chein and Mugnier, 2009; Chein and Mugnier, 2014]), UML class diagrams, description logics [Baader *et al.*, 2010], etc. In this thesis, we are interested in the use of description logics, a family of logic-based languages of ontologies, mainly based on first order logics, which allows an efficient encoding and reasoning about the knowledge of a particular domain. Description logics have regained an important place in various domain areas such as the ontology-based data access (*e.g.* [Pinto *et al.*, 2012; Artale *et al.*, 2013]), information and data integration (*e.g.* [Meyer *et al.*, 2005]) and the Semantic Web (*e.g.* [Baader *et al.*, 2005b]) where they provide the foundations of the Web Ontology Language (OWL).

Nowadays, there exist several description logic languages that serve ontologies. As all logical formalisms, each description language is characterized by its expressive power and its reasoning complexity (the complexity of algorithms used for inference). There is a tradeoff between expressiveness and complexity of reasoning. Namely, more the language is expressive, more the computational complexity of reasoning is high [Brachman and Levesque, 1985].

The compromise between expressivity and complexity of reasoning is one of the main concerns in description logics area. Besides, most of the well-known classical description logics are intractable, in the sense that they do not guarantee polynomial complexity when reasoning. In general, these description logics are not designed to face recent applications where new challenges have raised. One can cite for instance, applications that involve large generic knowledge or huge volume of data where reasoning algorithms should scale up.

In recent years, several lines of research led to the introduction of lightweight description logics. These logics offer a nice compromise between the expressiveness and the tractability of the basic reasoning tasks. In this thesis, we focus on prominent members of the *DL-Lite* family that underly the *OWL2-QL*

profile, especially dedicated for applications that use large amounts of data.

The rest of this chapter is organized as follows: Section 1.2 introduces the notion of ontology and recalls the main languages that support ontologies. Section 1.3 overviews main concepts of description logics. Section 1.4 presents the *DL-Lite* family. Section 1.5 concludes this chapter.

1.2 Ontology languages

The term ontology has its origin in philosophy and refers to the study of existence and being. In knowledge representation and reasoning, a branch of artificial intelligence, the term ontology refers to a representation framework that explicitly describes a formal conceptualization of a domain of interest [Hitzler *et al.*, 2009]. An ontology specifies elements of a particular domain and describes relations and constraints holding over them. This latter is given by two distinct levels:

An intensional level: It describes a set of elements and specifies how to structure them using a set of rules called axioms.

Extensional level: It represents basic objects of the different elements given in the intensional level.

An ontology is supported by a language used to structure a domain of interest. In general, a language used to express an intensional level is usually built upon the following elements:

- **Concept:** Also called class, entity type or frame, it is used to denote a collection of objects (*e.g.* the concept "Mother" denotes the set of mothers of a particular domain).
- **Relationship:** Also called association, relationship, role or object property, it is used to express an association among concepts (*e.g.*, "hasChild" is defined on "Mother" and "Person").
- **Property:** Also called attribute, feature, slot, data property, it is used to qualify an element of the ontology (*i.e.* a concept or a relationship). A property can be either atomic (*e.g.* integer, real, string, etc) or structured (*e.g.* set, list, etc).
- **Axiom:** Also called assertion, it is a logical formula used to express constraints or rules that must be satisfied by the elements specified at the extensional level (*e.g.* Subsumption axiom: "Male" is a "Person", disjointness axiom: "Female" is not a "Male", etc).

A language used to encode an extensional level usually includes:

- **Instances:** An instance represents an individual or an object that belongs to a concept (*e.g.* Paul is an instance of Person).
- **Facts:** A fact represents a relationship holding between instances (*e.g.* HasChild(Marie,Paul)).

Note that in the rest of the thesis, we do not make difference between facts and instances. In the following, we present a classification of main ontology languages.

- **Graph-based languages:** Family of languages based on graphs to represent ontologies such: UML class diagrams, semantic networks [Sowa, 1987], conceptual graphs [Chein and Mugnier, 2009], etc.

- Frame-based languages : Family of languages based on the frame approach [Gruber, 1995]. The most known languages based on frames are OKBC¹ (Open Knowledge Base Connectivity), KM² (Knowledge Machine).
- Logic-based languages: It is a family of languages based on logics to represent ontologies such as first order logic (e.g. KIF³, CycL⁴), Description Logics [Baader *et al.*, 2010], Existential Rule (e.g. [Calì *et al.*, 2012; Mugnier and Thomazo, 2014]), F-logic [Kifer and Lausen, 1989], etc.

In this thesis, we are interested in logic-based languages, and in particular, the use of description logics as ontology languages. To obtain a semantically rich representation of a domain of interest, one can formalize the intensional level and the extensional level as a theory and then use this latter to perform reasoning tasks.

Semantic Web has been conceived as an extension of the World Wide Web that allows computers to intelligently search, combine, and process Web contents based on the meaning that this content has to humans [Hitzler *et al.*, 2009; Shadbolt *et al.*, 2006]. As for today, the most prominent standard technologies, recommended by the W3C⁵, for Semantic Web are based on ontologies. Description logics provide the foundations of the Web Ontology Language, one of the most important markup ontology languages recommended for the Semantic Web. In the following, we recall the widely used markup ontology languages:

- Resource Description Framework (RDF)⁶: It is a standard used for data interchange on the Web. RDF has features that assist data merging even if the underlying schemas differ. Moreover, RDF specifically supports the evolution of schemas over time without requiring the modification of the data which are based on it.
- RDF Schema (RDFS)⁷: It is a semantic extension of RDF that provides a data-modelling vocabulary for RDF data.
- Ontology Web Language (OWL)⁸: It is an ontology language for the Semantic Web based on description logics.

According to [W3C, 2008], a concrete syntax is needed in order to store *OWL2* ontologies and to exchange them among tools and applications. The primary exchange syntax for *OWL2* is the *RDF/XML* language, which is compatible with the *XML* serializations of *RDF* documents, and it is the syntax that must be supported by all *OWL2* tools. There are also other concrete syntaxes that may also be used, for instance the Manchester syntax (largely used in ontology editing tools) and the functional syntax (used to specify the structure of the ontology language). Finally, two alternative semantics for *OWL* are proposed: the direct semantics (*OWL2-DL*) which is based on a description logic called *SROIQ* and the RDF-based semantics (*OWL2-Full*). To develop applications that manipulate ontologies, the most commonly used tools are:

- *Protégé-OWL API*⁹: The Protege-OWL API is an open-source Java library for the OWL and RDF(S). The API provides classes and methods to load and save OWL files, to query and ma-

¹<http://www.ai.sri.com/~okbc/>

²<http://www.cs.utexas.edu/users/mfkb/RKF/km.html>

³<http://www.ksl.stanford.edu/knowledge-sharing/kif/>

⁴<http://www.cyc.com/documentation/syntax-cycl>

⁵<http://www.w3.org>

⁶<http://www.w3.org/RDF/>

⁷<http://www.w3.org/TR/rdf-schema/>

⁸<http://www.w3.org/standards/techs/owl>

⁹http://protegewiki.stanford.edu/wiki/ProtegeOWL_API_Programmers_Guide

nipulate OWL data models, and to perform reasoning based on description logic algorithms. Furthermore, the API is optimized for the implementation of graphical user interfaces. Protégé-OWL API is built on top of OWL API.

- *OWL API* ¹⁰: It is a Java API and reference implementation for creating, manipulating and serializing OWL ontologies. The OWL API includes the following components: An API for OWL2 and an efficient in-memory reference implementation and parsers and writers for OWL in several formats, *e.g.* RDF/XML, OWL/XML parser, functional syntax of OWL, etc. Finally, it provides a reasoner interface that is supported by many description logics reasoners.
- *Jena API* ¹¹: Jena is an open source framework for Java. It provides an API to extract data from and write to RDF graphs. The graphs are represented as an abstract "model". A model can be sourced with data from files, databases, URLs or a combination of these.

In recent years, there is a large use of ontologies in various application areas where new challenges emerged. These challenges mainly consist in equipping ontologies with new reasoning capabilities (*e.g.* evolution, merging, inconsistency handling, etc) or additional expressivity (*e.g.* uncertainty management) in order to face new requirements. In the following, we present some current lines of research on ontology formalisms:

1. **Ontology matching** (*e.g.* [Euzenat and Shvaiko, 2013; Shvaiko and Euzenat, 2013]): Given two heterogeneous ontologies, matching consists in producing an unified ontology associated with mappings that explicit the different correspondences between the vocabularies used in the input ontologies.
2. **Ontology translation** (*e.g.* [Dou *et al.*, 2005; Gruber, 1993; Dou *et al.*, 2011]): Consists in equivalently translating an ontology, (*i.e.* axioms and/or vocabulary) expressed using a language \mathcal{L}_1 , into an ontology using another representation language \mathcal{L}_2 .
3. **Ontology integration** (*e.g.* [Meyer *et al.*, 2005; Hou *et al.*, 2005]): Given a set of ontologies that represent knowledge about a similar domain, integration consists in combining these ontologies in order to obtain more knowledge by unifying the domain.
4. **Ontology modularity** (*e.g.* [Grau *et al.*, 2008; Grau *et al.*, 2009]): It consists in extracting the smallest independent subsets of an ontology, called modules, with the aim to reusing them later in other applications.
5. **Ontology evolution** (*e.g.* [Noy *et al.*, 2006; Plessers *et al.*, 2007]): It consists in modifying an ontology according to a set of change operations that may concern the knowledge about the domain or the structure of the ontology.
6. **Ontology merging** (*e.g.* [Moguillansky and Falappa, 2007; Kotis *et al.*, 2006; Noy and Musen, 2000]): Given two ontologies that represent knowledge on the same domain, merging consists in producing a single ontology that represents a global point of view.
7. **Ontology debugging** (*e.g.* [Kalyanpur *et al.*, 2005; Corcho *et al.*, 2009]): It includes i) ontology diagnosis which consists in restoring the coherency of the intensional level of an ontology (*e.g.* [Peñaloza and Sertkaya, 2010; Ludwig and Peñaloza, 2014]) and ii) ontology repairing which consists in restoring consistency of the ontology (*e.g.* [Lembo *et al.*, 2010; Bienvenu, 2012]).

¹⁰<http://owlapi.sourceforge.net>

¹¹<https://jena.apache.org>

8. Ontology uncertainty management ¹² (e.g. [Lukasiewicz and Straccia, 2008; Lukasiewicz *et al.*, 2012a; Bobillo *et al.*, 2013]): This consists in dealing with imprecision/uncertainty that can be attached to the elements of the vocabulary (e.g. concepts reflecting imprecise terms) or axioms (e.g. uncertainty attached to an axiom).
9. Ontology peer-to-peer systems (e.g. [Adjiman *et al.*, 2006]): It consists in offering access to the information maintained by a set of peers linked to each other.
10. Ontology-based Data Access (e.g. [Pinto *et al.*, 2012; Artale *et al.*, 2013]): This consists in unifying access to independent sets of data sources using an unified ontological view to which the data sources are linked.

This thesis focuses on some ontology challenges given above, namely ontology evolution, merging, debugging (especially inconsistency handling within Ontology-Based Data Access setting) and uncertainty management in ontologies. Note that these tasks are closely related to existing works done on belief revision, belief merging, uncertainty and inconsistency handling within the propositional logic setting. Chapter 2 gives an overview of these topics.

As said before, we use description logics as ontology language, with a focus on *DL-Lite*, a family of lightweight description logics especially tailored for Ontology-Based Data Access. Before introducing description logics and *DL-Lite*, we first recall basic concepts of propositional logic and first order logic, needed to follow this thesis, and we fix some notations.

1.3 Logic-based languages

The aim of logic in artificial intelligence is to develop languages to formally represent knowledge of a domain and make them available for reasoning [Huth and Ryan, 2004]. A formal language, denoted by \mathcal{L} , is equipped with a syntax, allowing a logical expression of formulas attached to a formal semantics telling the right meaning of these formulas. In general, the semantics of a language specifies how one can reason on the knowledge encoded syntactically through formulas.

1.3.1 Propositional logic

Propositional logic is one of the simplest languages for knowledge representation and reasoning. It is used in many applications to express statements to which one assigns a truth value (*i.e.* true or false) according to the possible world. This section recalls the syntax and the semantics of propositional logic. For more details, see for example [Garriga, 2013].

Syntax. The propositional logic vocabulary \mathcal{V} is given in terms of propositional variables, called also propositions or atoms and denoted by tiny letters (a, b, \dots). A propositional variable is a boolean variable that one can assign either true or false as truth value. The language \mathcal{L} of propositional logic is built over a set of propositional variables, boolean constants: True (\top) and False (\perp), a set of logical connectors composed of: negation (\neg), conjunction (\wedge), disjunction (\vee), implication (\rightarrow) and equivalence (\leftrightarrow).

Definition 1.1. Given a propositional language \mathcal{L} , the elements of \mathcal{L} are called propositional formulas (or well-formed formulas) and expressed in the following way:

- \perp and \top are formulas,

¹²<http://www.w3.org/2005/Incubator/urw3/XGR-urw3-20080331/>

- if $p \in \mathcal{V}$, then p is a formula.
- if ϕ is a formula, then $\neg\phi$ is a formula (Negation).
- if ϕ_1 and ϕ_2 are formulas, then $(\phi_1 \wedge \phi_2)$ is a formula. (Conjunction)
- if ϕ_1 and ϕ_2 are formulas, then $(\phi_1 \vee \phi_2)$ is a formula. (Disjunction)
- if ϕ_1 and ϕ_2 are formulas, then $(\phi_1 \rightarrow \phi_2)$ is a formula. (Implication)
- if ϕ_1 and ϕ_2 are formulas, then $(\phi_1 \leftrightarrow \phi_2)$ is a formula (Equivalence)

Propositional formulas are built using formulas given above. A literal l is either a propositional variable, called a positive literal, or its negation, called a negative literal. A clause is a finite disjunction of literals (in particular the constant \top , when the set of literals is empty) and a term is a finite conjunction of literals (in particular the constant \perp , when the set of literals is empty). A propositional formula ϕ is said to be in a Conjunctive Normal Form (CNF) if it is formed by a conjunction of clauses. A propositional formula ϕ is said to be in a Disjunctive Normal Form (DNF) if it is constituted by a disjunction of terms.

Semantics. The semantics of propositional logic is given in terms of interpretations.

Definition 1.2. An interpretation, denoted by I , is a mapping that assigns to each propositional variable p of a formula a truth value, true or false, denoted by p^I . Given an interpretation I , the propositional formulas are interpreted as follows:

- $\top^I = true$ and $\perp^I = false$.
- $(\neg\phi)^I = true$ if $(\phi)^I = false$, and $(\neg\phi)^I = false$ otherwise.
- $(\phi_1 \wedge \phi_2)^I = true$ if $(\phi_1)^I = true$ and $(\phi_2)^I = true$ and $(\phi_1 \wedge \phi_2)^I = false$ otherwise.
- $(\phi_1 \vee \phi_2)^I = true$ if $(\phi_1)^I = true$ or $(\phi_2)^I = true$ and $(\phi_1 \vee \phi_2)^I = false$ otherwise.
- $(\phi_1 \rightarrow \phi_2)^I = true$ if $(\phi_1)^I = false$ or $(\phi_2)^I = true$, and $(\phi_1 \rightarrow \phi_2)^I = false$ otherwise.

Let I be an interpretation (an instantiation of all propositional variables) and ϕ be a propositional formula. We say that I is a model of ϕ or I satisfies ϕ , denoted by $I \models \phi$ if and only if $(\phi)^I = true$, otherwise, we say that I falsifies ϕ or I is a counter-model of ϕ and it is denoted by $I \not\models \phi$. ϕ is said to be valid (*i.e.* a tautology) if it does not admit any counter-model. Otherwise, it is said to be invalid. A contradiction is a formula that does not admit any model.

Up to now, we presented the knowledge representation aspect in propositional logic. We now introduce the reasoning aspect which consists in deriving implicit knowledge from the ones explicitly represented. The principle of logical deduction represents the central element of reasoning in all logics. Logical deduction in propositional logic can be defined as follows:

Definition 1.3 (Logical deduction). Given two formulas ϕ_1 and ϕ_2 . We say that ϕ_2 is entailed by ϕ_1 , denoted $\phi_1 \models \phi_2$, if for every interpretation I that is model of ϕ_1 , I is also a model of ϕ_2 .

The truth table is a sure way to check the validity of a logical deduction (all reasoning tasks in general). However, it is not practically possible, since one should, in the worst case, enumerate 2^n interpretations to find a model, where n is the number of propositional variables present in the considered formulas. To this end, logical deduction can be done syntactically using the well-known refutation theorem, which states that ϕ_2 is a logical consequence of ϕ_1 if and only if $\phi_1 \wedge \neg\phi_2$ is unsatisfiable. Finally, note that the complexity of satisfiability problem of a set of propositional formulas is NP-complete [Cook, 1971].

1.3.2 First order logic

This section gives a brief refresher on the syntax of First Order Logic (FOL for short), also called predicate logic. For more details on FOL, the readers can refer to [Fitting, 1990; Huth and Ryan, 2004] for examples.

The first order logic vocabulary \mathcal{V} is built upon disjoint and finite sets N_C , N_F , N_P and N_V where N_C is a set of constant symbols (also called individuals), N_F is a set of function symbols, N_P is a set of predicate symbols (or simply predicates) and N_V is a set of variable names. Each function or predicate symbol is associated with a natural number, called arity.

Definition 1.4 (First-order terms). Given a first-order vocabulary $\mathcal{V} = (N_C, N_F, N_P, N_V)$, the set of terms is defined such that:

- if $x \in N_V$, then x is a term.
- if $a \in N_C$, then a is a term.
- Let f be an n -ary function and t_1, t_2, \dots, t_n be terms, then $f(t_1, t_2, \dots, t_n)$ is a term. Note that a 0-ary function is called a constant.

The terms are used also as arguments for predicates to form atomic formulas.

Definition 1.5 (First order atom). Let $\mathcal{V} = (N_C, N_F, N_P, N_V)$ be a first-order vocabulary. Let f be an n -ary predicate and t_1, t_2, \dots, t_n be terms. Then an expression of the form $P(t_1, t_2, \dots, t_n)$ is said to be an atom. In first order logic with equalities, expressions of the form $t_1 = t_2$ or $t_1 \neq t_2$ where t_1 and t_2 are terms, are also called atoms.

The language \mathcal{L} of a FOL is built over a set of atoms, a set of logical connectors (\neg , \wedge , \vee , \rightarrow and \leftrightarrow) as in propositional logic, the symbols (\top) and (\perp) that correspond to 0-ary predicates, the universal quantifier (\forall) and the existential quantifier (\exists).

Definition 1.6 (First order formulas). Given a first-order language \mathcal{L} , the elements of \mathcal{L} are called first-order formulas and they are formed as follows:

- \perp and \top are formulas,
- Each atom is a formula. (Atomic formula)
- If ϕ is a formula, then $(\neg\phi)$ is a formula. (Negation)
- If ϕ_1 and ϕ_2 are formulas, then $(\phi_1 \wedge \phi_2)$ is a formula. (Conjunction)
- If ϕ_1 and ϕ_2 are formulas, then $(\phi_1 \vee \phi_2)$ is a formula. (Disjunction)
- If ϕ_1 and ϕ_2 are formulas, then $(\phi_1 \rightarrow \phi_2)$ is a formula. (Implication)
- If ϕ_1 and ϕ_2 are formulas, then $(\phi_1 \leftrightarrow \phi_2)$ is a formula. (Equivalence)
- If x is a variable and ϕ is a formula, then $(\forall x.\phi)$ is a formula. (Universal quantification)
- If x is a variable and ϕ is a formula, then $(\exists x.\phi)$ is a formula. (Existential quantification)
- Let '=' be a binary predicate symbol, the formulas $=(x, y)$ and $\neg=(x, y)$ are called equalities and they are simply denoted by $x = y$ and $x \neq y$. (Equalities)

Note that the quantifiers have higher priority over all other connectors which have the same priorities as in propositional logic. A 0-ary predicate is called a proposition. Indeed, propositional logic is a fragment of first-order logic where predicates are restricted to 0-ary predicates and without using quantifications.

Definition 1.7 (Subformulas). Let ϕ be a first-order formula. The subformulas of ϕ are ϕ itself and all immediate subformulas of ϕ .

- Atomic formulas \top and \perp have no immediate subformulas.
- The only immediate subformula of $\neg\phi$ is ϕ . The immediate subformulas of $\phi_1 \vee \phi_2$ or $\phi_1 \wedge \phi_2$ or $\phi_1 \rightarrow \phi_2$ or $\phi_1 \leftrightarrow \phi_2$ are ϕ_1 and ϕ_2 . The only immediate subformula of $\forall x.\phi$ or $\exists x.\phi$ is ϕ .

Definition 1.8 (Scope, bound variable and free variable). Let ϕ be a formula, Q a quantifier, and $Qx.\varphi$ a subformula of ϕ . Then Qx is called a quantifier for x . Its scope in φ is the subformula φ except subformulas of φ that begin with a quantifier for the same variable x .

Each occurrence of the variable x in the scope of Qx is said to be *bound* in ϕ by Qx . Each occurrence of x that is not in the scope of any quantifier for x is a *free* occurrence of x in ϕ .

Namely, x is bound if it is not contained in any subformula $Qx.\varphi$ of ϕ . A formula is said to be a closed formula if it contains only bound occurrences of variables. Otherwise, it is said to be an open formula. A ground term is a term containing no variable. A ground formula is a formula containing no variable.

1.3.3 Description logics

Description logics (DLs for short) are a family of formalisms designed to represent knowledge of a particular domain, and subsequently, reason by deriving new knowledge. DLs were introduced as decidable subsets of FOL, having a precise model-theoretic semantics [Baader *et al.*, 2010]. A DL language only uses unary and binary predicates, called respectively concepts and roles. To represent generic knowledge (*i.e.* intensional level), DLs restrict forms of logical formulas (called axioms), using only concepts and role inclusions. To represent the extensional level, DLs use ground facts expressed in term of membership assertions on concepts or roles. Interestingly enough, DLs provide a good compromise between expressive power and computational complexity. Throughout this section, we present basic concepts of DLs.

A DL knowledge base is built upon a description vocabulary \mathcal{V} consisting of atomic concepts which correspond to unary predicates to denote sets of individuals, and atomic roles, which correspond to binary predicates, to denote binary relations among individuals.

Definition 1.9 (DL vocabulary). A DL description vocabulary \mathcal{V} is a triple (N_C, N_R, N_I) of pairwise disjoint sets of atomic concept names, denoted by N_C , atomic role names, denoted by N_R and individual names, denoted by N_I .

Example 1.1. For example, the atomic concept "Mother" represents the set of all mothers in a particular domain. The atomic role "marriedTo" represents the set of all married couples in a particular domain. The individuals "Marie" and "Paul" represent constants in a particular domain. ■

Syntax

A description language \mathcal{L} is characterized by a set of constructs used to form complex concepts and roles from atomic ones. It is used to structure a domain of interest through a set of logical formulas, called

axioms. Each description language allows different sets of constructs. We start with one of the most basic description languages, called \mathcal{ALC} (*Attributive Language with Complement*), which is an extension of the \mathcal{AL} language that constitutes the foundation framework of many other DLs. The \mathcal{ALC} allows the use of conjunction, disjunction, negation, universal quantification and existential quantification to form complex concepts and roles as presented in Table 1.1:

Constructor	Designation
\top	Top concept
\perp	Bottom concept
A	Atomic concept
$\neg C$	Negation
$C \sqcap D$	Conjunction
$C \sqcup D$	Disjunction
$\exists R.C$	Existential quantification
$\forall R.C$	Universal quantification

Table 1.1: Constructors of the \mathcal{ALC} logic where C and D are concepts and R is a role.

Note that $\exists R$ and $\forall R$ can be used as an abbreviation of $\exists R.\top$ and $\forall R.\top$ and any atomic concept $A \in N_C$ is also a concept.

Example 1.2. Let the following atomic concepts "Male", "Female" and "Person" be three atomic concepts and let "hasChild" be an atomic role. Using \mathcal{ALC} language, one can express the following complex concepts:

- $Person \sqcap \neg Parent$: Persons who are not parents.
- $Female \sqcap Person$: Female persons.
- $(Male \sqcap Person) \sqcap \exists hasChild.Person$: Male persons who have at least a child.
- $Person \sqcap \forall hasChild.\neg Female$: Persons who do not have a Female as child.

■

A DL knowledge base, denoted by $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ contains two distinct components: A terminological base, called TBox and denoted by \mathcal{T} , that describes the generic knowledge about the domain, and an assertional base, called ABox and denoted by \mathcal{A} , that contains the assertional facts (*i.e.* individuals or constants) that instantiate the terminological knowledge. Namely, the TBox encodes the intensional level and the ABox stores the extensional level of an ontology.

Definition 1.10 (DL axioms). Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an \mathcal{ALC} knowledge base. Then:

- The TBox contains a set of terminological axioms of the form:
 - $C \sqsubseteq D$: C is a subconcept of D .
 - $C \equiv D$: C is equivalent to D (namely, $C \sqsubseteq D$ and $D \sqsubseteq C$).
- The ABox contains a set of assertion axioms of the form:

- $C(a)$: a is an instance of C .
- $R(a, b)$: a and b are related by R .

where C and D are concepts, R is a role and a and b are individuals.

Example 1.3. Consider the following set of atomic concepts $N_C = \{Person, Female, Male\}$ and the following set of atomic roles $N_R = \{hasChild\}$. In the following, we give an example of an \mathcal{ALC} TBox:

$$\begin{aligned}
 Woman &\equiv Person \sqcap Female \\
 Male &\sqsubseteq \neg Female \\
 Man &\equiv Person \sqcap \neg Women \\
 Mother &\equiv Woman \sqcap \exists hasChild. Person \\
 Father &\equiv Man \sqcap \exists hasChild. Person \\
 Parent &\equiv Father \sqcup Mother \\
 ParentWithoutDaughter &\equiv Parent \sqcap \forall hasChild. \neg Woman
 \end{aligned}$$

In the following, we give an example of ABox:

$$\begin{aligned}
 Person(Paul) \quad & Person(Marie) \\
 Male(Bob) \quad & Female(Marie) \\
 hasChild(Marie, Paul) \quad & hasChild(Bob, Alice)
 \end{aligned}$$

■

Semantics

The semantics of \mathcal{ALC} is in the spirit of first order logic semantics and it is given in terms of interpretations.

Definition 1.11 (DL interpretation). An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of a non-empty domain, denoted by $\Delta^{\mathcal{I}}$, and an interpretation function, denoted by $\cdot^{\mathcal{I}}$, defined from N_I to $\Delta^{\mathcal{I}}$. The function $\cdot^{\mathcal{I}}$ associates with each individual a an element $a^{\mathcal{I}}$ of $\Delta^{\mathcal{I}}$, to each atomic concept A a subset $A^{\mathcal{I}}$ of $\Delta^{\mathcal{I}}$ and to each atomic role R a binary relation $R^{\mathcal{I}}$ over $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. Furthermore, the interpretation function $\cdot^{\mathcal{I}}$ is extended in a straightforward way for \mathcal{ALC} concepts and roles as follows:

$$\begin{aligned}
 A^{\mathcal{I}} &\subseteq \Delta^{\mathcal{I}} \\
 R^{\mathcal{I}} &\subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \\
 (\neg C)^{\mathcal{I}} &= \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}} \\
 (C \sqcap D)^{\mathcal{I}} &= C^{\mathcal{I}} \cap D^{\mathcal{I}} \\
 (C \sqcup D)^{\mathcal{I}} &= C^{\mathcal{I}} \cup D^{\mathcal{I}} \\
 (\exists R.C)^{\mathcal{I}} &= \{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}} \text{ such that } (x, y) \in R^{\mathcal{I}} \text{ and } y \in C^{\mathcal{I}}\} \\
 (\forall R.C)^{\mathcal{I}} &= \{x \in \Delta^{\mathcal{I}} \mid \text{if } (x, y) \in R^{\mathcal{I}} \text{ then } y \in C^{\mathcal{I}}\}
 \end{aligned}$$

Example 1.4. Let us continue Example 1.3. Assume that $\Delta^{\mathcal{I}} = \{Marie, Paul, Bob, Alice, Titi\}$. One

can consider the following interpretation:

$$\begin{aligned}
(Person)^{\mathcal{I}} &= \{Paul, Marie, Bob, Alice\} \\
(Female)^{\mathcal{I}} &= \{Marie, Alice, Titi\} \\
(Male)^{\mathcal{I}} &= \{Bob, Paul\} \\
(hasChild)^{\mathcal{I}} &= \{(Marie, Paul), (Bob, Alice)\} \\
(Woman)^{\mathcal{I}} &= \{Marie, Alice\} \\
(Man)^{\mathcal{I}} &= \{Bob, Paul\} \\
(Mother)^{\mathcal{I}} &= \{Marie\} \\
(Father)^{\mathcal{I}} &= \{Bob\} \\
(ParentWithoutDaughter)^{\mathcal{I}} &= \{Marie\} \\
(Parent)^{\mathcal{I}} &= \{Marie, Bob\}
\end{aligned}$$

■

An interpretation \mathcal{I} is said to satisfy a knowledge base $\mathcal{K}=\langle\mathcal{T}, \mathcal{A}\rangle$ if and only if \mathcal{I} satisfies every axiom in \mathcal{T} and every axiom in \mathcal{A} . Such interpretation is said to be a model of \mathcal{K} .

Definition 1.12. Let $\mathcal{K}=\langle\mathcal{T}, \mathcal{A}\rangle$ be an \mathcal{ALC} DL knowledge base. The satisfiability of axioms of \mathcal{K} with respect to an interpretation \mathcal{I} is defined as follows:

- $\mathcal{I} \models C \sqsubseteq D$ if and only if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$.
- $\mathcal{I} \models C \equiv D$ if and only if $C^{\mathcal{I}} = D^{\mathcal{I}}$.
- $\mathcal{I} \models C(a)$ if and only if $a^{\mathcal{I}} \in C^{\mathcal{I}}$.
- $\mathcal{I} \models R(a, b)$ if and only if $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$.

Example 1.5. From Example 1.4, one can check that \mathcal{I} is a model of the knowledge bases of Example 1.3

■

Basic reasoning tasks

Reasoning is a fundamental issue in DLs. It allows to derive implicit knowledge from the one explicitly represented in the knowledge base. The main standard reasoning services over a DL knowledge base are:

- **Concept satisfiability:** A concept C is said to be satisfiable (or consistent), with respect to a TBox \mathcal{T} , if there exists an interpretation \mathcal{I} that is a model of \mathcal{T} such that $C^{\mathcal{I}} \neq \emptyset$.
- **Subsumption checking:** A concept C is said to be subsumed by another concept D with respect to a TBox \mathcal{T} , denoted by $\mathcal{T} \models C \sqsubseteq D$, if for each interpretation \mathcal{I} such that $\mathcal{I} \models \mathcal{T}$, $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ holds.
- **Classification:** Given a TBox \mathcal{T} , classification consists in computing all pairs of concepts (C, D) such that $\mathcal{T} \models C \sqsubseteq D$.
- **Knowledge base satisfiability or consistency:** An ABox \mathcal{A} is said to be consistent with respect to a TBox \mathcal{T} , if there exists an interpretation \mathcal{I} such that $\mathcal{I} \models \mathcal{T}$ and $\mathcal{I} \models \mathcal{A}$.
- **Instance checking:** An individual a (resp. a and b) is said to be an instance of C (resp. are related by a role R) with respect to a knowledge base $\mathcal{K} = \langle\mathcal{T}, \mathcal{A}\rangle$, denoted by $\mathcal{K} \models C(a)$ (resp. $\mathcal{K} \models R(a, b)$), if for each interpretation \mathcal{I} such that $\mathcal{I} \models \mathcal{T}$ and $\mathcal{I} \models \mathcal{A}$, we have $a^{\mathcal{I}} \in C^{\mathcal{I}}$ (resp. $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$).

	Description	Syntax	Semantics
\mathcal{R}_+	Transitive role	$(tra R)$	$(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$ and $(b^{\mathcal{I}}, c^{\mathcal{I}}) \in R^{\mathcal{I}}$ implies $(a^{\mathcal{I}}, c^{\mathcal{I}}) \in R^{\mathcal{I}}$
\mathcal{H}	Role hierarchies	$R \sqsubseteq S$	$R^{\mathcal{I}} \subseteq S^{\mathcal{I}}$
\mathcal{I}	Inverse roles	R^-	$\{(b, a) (a, b) \in R^{\mathcal{I}}\}$
\mathcal{F}	Functional roles	$(funct R)$	$(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$ and there is no $c^{\mathcal{I}} \neq b^{\mathcal{I}}$ s.t. $(a^{\mathcal{I}}, c^{\mathcal{I}}) \in R^{\mathcal{I}}$
\mathcal{O}	Nominals	$\{a_1, \dots, a_n\}$	$\{a_1^{\mathcal{I}}, \dots, a_n^{\mathcal{I}}\}$
\circ	Role composition Complex role hierarchies	$R \circ S$ $R \circ S \sqsubseteq R$	$\{(a^{\mathcal{I}}, c^{\mathcal{I}}) \exists b \in \Delta^{\mathcal{I}} \text{ s.t. } (a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}} \text{ and } (b^{\mathcal{I}}, c^{\mathcal{I}}) \in S^{\mathcal{I}}\}$ $(R \circ S)^{\mathcal{I}} \subseteq R^{\mathcal{I}}$
\mathcal{N}	Number restrictions	$\bowtie nR$	$\{a \in \Delta^{\mathcal{I}} \text{card} \{b \in \Delta^{\mathcal{I}} (a, b) \in R^{\mathcal{I}}\} \bowtie n\}$
\mathcal{Q}	Qualified number restrictions	$\bowtie nR.C$	$\{a \in \Delta^{\mathcal{I}} \text{card} \{b \in \Delta^{\mathcal{I}} (a, b) \in R^{\mathcal{I}} \wedge b \in C^{\mathcal{I}}\} \bowtie n\}$

 Table 1.2: DLs constructors where $\text{card}(X)$ denotes the cardinality of the set X and \bowtie denotes \leq or \geq .

The knowledge base satisfiability or consistency problem is the main reasoning task in DLs. It allows to check whether the knowledge encoded in the TBox and the ABox is non-contradictory. According to [Horrocks and Patel-Schneider, 2004], all the above reasoning tasks can be reduced from subsumption test to satisfiability test, from instance checking to knowledge base consistency, etc. For instance [Baader et al., 2010]:

- C is unsatisfiable if and only if $\mathcal{T} \models C \sqsubseteq \perp$.
- $\mathcal{T} \models C \sqsubseteq D$ if and only if $C \sqcap \neg D$ is unsatisfiable.
- C is satisfiable if and only if $\{C(a)\}$ is consistent with respect to \mathcal{T} .
- $\mathcal{A} \models C(a)$ if and only if $\mathcal{A} \cup \{\neg C(a)\}$ is inconsistent with respect to \mathcal{T} .

In general, there exist several approaches of reasoning. The most widely used one is the so called tableau algorithm [Baader et al., 2010]. A tableau algorithm uses the concepts of negation to reduce subsumption to an (un)satisfiability problem. A tableau algorithm verifies whether a knowledge base contains contradictions or not by checking the existence of an interpretation that is a model of the knowledge base by constructing its finite representation, so-called tableau. Such technique decomposes axioms of the knowledge base using a set of consistency-preserving transformation rules (depending on the constructors used in the DL), called completion rules. For each application of a rule, an expression of an axiom is decomposed while preserving the semantics behind it into simple expressions. This decomposition leads to exhibit contradictory elements of the ABox. Therefore, one can check if there exists a model for the given knowledge base.

Expressive description logics

To define a DL language, one first needs to specify the set of concept and role constructors that can be used, and then what types of axioms that can be expressed in the knowledge base. In order to meet the needs of applications that require more expressiveness, the set of constructors in \mathcal{ALC} was enriched with several constructors. Table 1.2 summarizes the most used ones.

By convention, we use \mathcal{S} to denote \mathcal{ALCR}_+ , the \mathcal{ALC} constructors extended with role transitivity. A DL language is defined by a string of capital letters referring to the used constructors. There exist several DLs where the majority of them underly the ontology Web language *OWL*. For instance, in its first version *OWL1* where *OWL-Lite* is based on *SHIF* and *OWL-DL* is based on *SHOIN*, and in its second version *OWL2* where *OWL2-DL* is based on *SROIQ*, etc.

Computational complexity in description logics

The computational complexity of DLs is well studied in the literature. Given a decision problem, the complexity of DL reasoning tasks is performed around two settings.

- The combined complexity: It considers all the components of the knowledge base $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ as inputs, namely the size of the problem is equal to $|\mathcal{K}|$ with $|\mathcal{K}| = |\mathcal{T}| + |\mathcal{A}|$.
- The data complexity: It considers the TBox as fixed and only takes as input the size of the ABox $|\mathcal{A}|$.

Table 1.3 summarizes the computational complexity of consistency checking in some important DLs, The complexity results of other DLs including the ones given in Table 1.3, can be found at the following link ¹³.

DL family	Combined Complexity	Data Complexity
<i>ALC</i>	EXPTIME-complete	NP-complete
<i>SHIF</i>	EXPTIME-complete	NP-complete
<i>SHOIN</i>	NEXPTIME-complete	NP-hard
<i>SROIQ</i>	N2EXPTIME-complete	NP-hard

Table 1.3: Computational complexity of consistency checking of some expressive DLs

Description logics Reasoners

There are several implementations of reasoning task algorithms for DLs. These implementations are operated in general around ontology languages and using programming tools presented in Section 1.2. In what follows, we give a description of some well-known reasoners. ¹⁴.

- Pellet¹⁵: Pellet is an open-source reasoner for *SROIQ* DL with simple datatypes (*i.e.* *OWL2-DL*) developed in Java. It can be used in command line, OWL API, Jena API, Pellet API or with Protégé. It supports the *OWL2-DL* language and includes some support for the *OWL2* profiles. The main features of Pellet is that it incorporates optimizations for the use of nominals, conjunctive query answering, and incremental reasoning. As a reasoning technique, Pellet uses a tableau-based algorithm.

¹³<http://www.cs.man.ac.uk/~ezolin/dl/>

¹⁴A complete list of DL reasoners with their description is available at this link: <http://www.cs.man.ac.uk/~sattler/reasoners.html>

¹⁵<http://clarkparsia.com/pellet>

- Hermit¹⁶: Hermit is an open-source reasoner for *SROIQ* DL with simple datatypes (*i.e.* *OWL2-DL*) that supports description graphs. In particular, Hermit implements a reasoner based on a novel “hypertableau” calculus which provides much more efficient reasoning than any previously known algorithm. It can be used with OWL API or integrated with Protégé editor.
- FaCT++¹⁷: FaCT++ is an open-source reasoner for *SROIQ* DL with simple datatypes (*i.e.* *OWL2-DL*) implemented in C++. It implements a tableau-based algorithm for general TBoxe’s (subsumption, satisfiability, classification) and ABoxe’s (query). Now, it is used as one of the default reasoners in the Protégé 4 editor.

From Table 1.3, one can check that the classical DLs are intractable in the sense that there is no efficient (*i.e.* polynomial time) algorithm for checking satisfiability. To this end, several lightweight fragments of DLs that offer a nice tradeoff between expressivity and complexity of reasoning, were introduced. One of these lightweight fragments DLs, is the *DL-Lite* family.

According to the official documentation of *W3C* three profiles of *OWL2* are proposed as sub-languages of the full *OWL2* language, to offer important advantages in particular application scenarios. These lightweight logics are the \mathcal{EL} family [Baader *et al.*, 2005a] (underpinning *OWL2-EL*), the *DL-Lite* family [Calvanese *et al.*, 2007a; Artale *et al.*, 2009] (underpinning *OWL2-QL*), and the DL Programs [Grosz *et al.*, 2003] (underpinning *OWL2-RL*).

In this thesis, we are interested in the *DL-Lite* family of description logics. *DL-Lite* is well suitable for ontology-based data access setting.

1.4 The *DL-Lite* family

In recent years, a lot of attention was given to *DL-Lite*, a family of lightweight DLs specifically designed for applications using huge volumes of data such as Web applications where query answering is the most important reasoning task [Calvanese *et al.*, 2007a]. In particular, *DL-Lite* guarantees an efficient computational complexity of the reasoning process. In fact, the idea behind the reasoning (consistency checking and query answering) in *DL-Lite* is based on the so-called FOL-reducibility property. This latter permits to considerably reduce reasoning tasks to the evaluation of FOL queries over the set of assertions (*i.e.* data). The efficiency of reasoning in *DL-Lite* is ensured thanks to the use of relational database techniques.

1.4.1 The *DL-Lite* family and *OWL2-QL*

The knowledge representation format considered in this thesis is the one of *DL-Lite* language. We mainly consider three tractable members of the *DL-Lite* family. Namely, the *DL-Lite_{core}* which is the core fragment of all *DL-Lite* logics, *DL-Lite_F* and *DL-Lite_R* which underlies the *OWL2-QL* profile. For the sake of simplicity and when there is no ambiguity, through this section (and this thesis, in general), we use *DL-Lite* to refer to these three fragments.

Syntax The starting points are N_C , N_R and N_I , three pairwise disjoint sets of atomic concepts, atomic roles and individuals. The *DL-Lite* language uses three unary connectors: “ \neg ”, “ \exists ” and “ $-$ ” and a binary connector “ \sqsubseteq ” to define complex concepts and roles and inclusions between concepts and roles. Let $A \in N_C$, $P \in N_R$, basic concepts (*resp.* roles) B (*resp.* R) and complex concepts (*resp.* roles) C (*resp.* E) are defined in *DL-Lite* as follows:

¹⁶<http://hermit-reasoner.com>

¹⁷<http://owl.cs.manchester.ac.uk/tools/fact/>

$$\begin{array}{l} R \longrightarrow P \mid P^- \quad E \longrightarrow R \mid \neg R \\ B \longrightarrow A \mid \exists R \quad C \longrightarrow B \mid \neg B \end{array}$$

where P^- represents the inverse of P .

A *DL-Lite* knowledge base is a pair $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$. The *DL-Lite_{core}* TBox is constituted by a finite set of inclusion axioms between concepts of the form

$$B \sqsubseteq C.$$

In the original conference paper [Calvanese *et al.*, 2005], *DL-Lite_{core}* does not use negation of roles. In the journal paper [Calvanese *et al.*, 2007a], negation appears in *DL-Lite*. Here, we follow description of *DL-Lite* used in the journal paper [Calvanese *et al.*, 2007a].

The ABox contains a finite set of membership assertions (or facts) on atomic concepts and on atomic roles respectively of the form

$$A(a) \text{ and } P(a, b)$$

where $A \in N_C$, $P \in N_R$ and $a, b \in N_I$.

The *DL-Lite_F* language extends *DL-Lite_{core}* with the ability of specifying functionality on roles or on their inverses of the form:

$$(\text{funct } R)$$

The *DL-Lite_R* language extends *DL-Lite_{core}* with the ability of specifying in the TBox inclusion axioms between roles of the form:

$$R \sqsubseteq E.$$

Note that *DL-Lite* language does not allow the use of the conjunctive and the disjunctive operators. However, one can easily add conjunctions (*resp.* disjunctions) in the right-hand side (*resp.* left-hand side) of inclusion axioms. Indeed, as we will see it later, the conjunction of the form $B \sqsubseteq C_1 \sqcap C_2$ is equivalent to the pair of inclusion axioms $B \sqsubseteq C_1$ and $B \sqsubseteq C_2$, while the disjunction of the form $B_1 \sqcup B_2 \sqsubseteq C$ is equivalent to the pair of inclusion assertions $B_1 \sqsubseteq C$ and $B_2 \sqsubseteq C$.

Any *DL-Lite* knowledge base can be equivalently written as a FOL knowledge base. Table 1.4 summarizes all possible expression of axiom in *DL-Lite* and their translation from to FOL formulas. For the ABox assertions, one can easily check that they are equivalent to ground atoms in FOL setting (see Section 1.3.2).

DL-Lite axiom	FOL formula	DL-Lite axiom	FOL formula
$A_1 \sqsubseteq A_2$	$\forall x.A_1(x) \rightarrow A_2(x)$	$A_1 \sqsubseteq \neg A_2$	$\forall x.A_1(x) \rightarrow \neg A_2(x)$
$A \sqsubseteq \exists P$	$\forall x.A(x) \rightarrow \exists y.P(x, y)$	$A \sqsubseteq \neg \exists P$	$\forall x.A(x) \rightarrow \neg \exists y.P(x, y)$
$A \sqsubseteq \exists P^-$	$\forall x.A(x) \rightarrow \exists y.P(y, x)$	$A \sqsubseteq \neg \exists P^-$	$\forall x.A(x) \rightarrow \neg \exists y.P(y, x)$
$\exists P \sqsubseteq A$	$\forall x.\exists y.P(x, y) \rightarrow A(x)$	$\exists P \sqsubseteq \neg A$	$\forall x.\exists y.P(x, y) \rightarrow \neg A(x)$
$\exists P^- \sqsubseteq A$	$\forall x.\exists y.P(y, x) \rightarrow A(x)$	$\exists P^- \sqsubseteq \neg A$	$\forall x.\exists y.P(y, x) \rightarrow \neg A(x)$
$\exists P_1 \sqsubseteq \exists P_2$	$\forall x.\exists y.P_1(x, y) \rightarrow \exists z(x, z)$	$\exists P_1 \sqsubseteq \neg \exists P_2$	$\forall x.\exists y.P_1(x, y) \rightarrow \neg \exists z(x, z)$
$\exists P_1 \sqsubseteq \exists P_2^-$	$\forall x.\exists y.P_1(x, y) \rightarrow \exists z(z, x)$	$\exists P_1 \sqsubseteq \neg \exists P_2^-$	$\forall x.\exists y.P_1(x, y) \rightarrow \neg \exists z(z, x)$
$\exists P_1^- \sqsubseteq \exists P_2$	$\forall x.\exists y.P_1(y, x) \rightarrow \exists z(x, z)$	$\exists P_1^- \sqsubseteq \neg \exists P_2$	$\forall x.\exists y.P_1(y, x) \rightarrow \neg \exists z(x, z)$
$\exists P_1^- \sqsubseteq \exists P_2^-$	$\forall x.\exists y.P_1(y, x) \rightarrow \exists z(z, x)$	$\exists P_1^- \sqsubseteq \neg \exists P_2^-$	$\forall x.\exists y.P_1(y, x) \rightarrow \neg \exists z(z, x)$
$P_1 \sqsubseteq P_2$	$\forall x, y.P_1(x, y) \rightarrow P_2(x, y)$	$P_1 \sqsubseteq \neg P_2$	$\forall x, y.P_1(x, y) \rightarrow \neg P_2(x, y)$
$P_1^- \sqsubseteq P_2^-$	$\forall x, y.P_1(x, y) \rightarrow P_2(x, y)$	$P_1^- \sqsubseteq \neg P_2^-$	$\forall x, y.P_1(x, y) \rightarrow \neg P_2(x, y)$
$P_1 \sqsubseteq P_2^-$	$\forall x, y.P_1(x, y) \rightarrow P_2(y, x)$	$P_1 \sqsubseteq \neg P_2^-$	$\forall x, y.P_1(x, y) \rightarrow \neg P_2(y, x)$
$P_1^- \sqsubseteq P_2$	$\forall x, y.P_1(x, y) \rightarrow P_2(y, x)$	$P_1^- \sqsubseteq \neg P_2$	$\forall x, y.P_1(x, y) \rightarrow \neg P_2(y, x)$
$(\text{funct } P)$	$\forall x, y, z.P(x, y) \wedge P(x, z) \rightarrow y = z$	$(\text{funct } P^-)$	$\forall x, y, z.P(y, x) \wedge P(z, x) \rightarrow y = z$

 Table 1.4: The equivalence of the *DL-Lite* axioms in FOL.

Example 1.6. Let *Teacher* and *Student* be two atomic concepts and *TeachesTo* and *HasSupervisor* be two atomic roles. In the following, we give an example of *DL-Lite_{core}* TBox:

$$\begin{aligned}
 & \textit{Teacher} \sqsubseteq \neg \textit{Student} \\
 & \textit{Teacher} \sqsubseteq \exists \textit{TeachesTo} \\
 & \exists \textit{TeachesTo}^- \sqsubseteq \textit{Student} \\
 & \textit{Student} \sqsubseteq \exists \textit{HasSupervisor} \\
 & \exists \textit{HasSupervisor}^- \sqsubseteq \textit{Teacher}
 \end{aligned}$$

To obtain a *DL-Lite_R* TBox, one can extend the above *DL-Lite_{core}* TBox with the following axiom:

$$\textit{HasSupervisor}^- \sqsubseteq \textit{TeachesTo}$$

To obtain a *DL-Lite_F* TBox, one can extend the *DL-Lite_{core}* TBox with the following axiom:

$$(\text{funct } \textit{HasSupervisor})$$

Finally, a *DL-Lite* ABox can be expressed as follows:

$$\textit{Student}(\textit{Paul}) \quad \textit{HasSupervisor}(\textit{Paul}, \textit{Alice}) \quad \textit{TeachesTo}(\textit{Alice}, \textit{Bob})$$

■

Semantics The semantics is given in terms of interpretations where as usual an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of a non-empty domain $\Delta^{\mathcal{I}}$ and an interpretation function $\cdot^{\mathcal{I}}$ that assigns each $a \in N_I$ to an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, each $A \in N_C$ to a subset $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and each $P \in N_R$ to $P^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$.

Furthermore, the interpretation function $\cdot^{\mathcal{I}}$ is extended in a straightforward way for $DL-Lite_{core}$ concepts and roles as follows:

$$\begin{aligned}(P^-)^{\mathcal{I}} &= \{(y, x) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid (x, y) \in P^{\mathcal{I}}\} \\ (\exists R)^{\mathcal{I}} &= \{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}} \text{ such that } (x, y) \in R^{\mathcal{I}}\} \\ (\neg B)^{\mathcal{I}} &= \Delta^{\mathcal{I}} \setminus B^{\mathcal{I}} \\ (\neg R)^{\mathcal{I}} &= \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \setminus R^{\mathcal{I}}\end{aligned}$$

An interpretation \mathcal{I} is said to be a model of a concept (*resp.* role) inclusion axiom, denoted by $\mathcal{I} \models B \sqsubseteq C$ (*resp.* $\mathcal{I} \models R \sqsubseteq E$), if and only if $B^{\mathcal{I}} \subseteq C^{\mathcal{I}}$ (*resp.* $R^{\mathcal{I}} \subseteq E^{\mathcal{I}}$). Similarly, we say that an interpretation \mathcal{I} is a model of a membership assertion $A(a)$ (*resp.* $P(a, b)$), denoted by $\mathcal{I} \models A(a)$ (*resp.* $\mathcal{I} \models P(a, b)$), if and only if $a^{\mathcal{I}} \in A^{\mathcal{I}}$ (*resp.* $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in P^{\mathcal{I}}$).

Regarding $DL-Lite_F$, we say that an interpretation \mathcal{I} is a model of an axiom (*funct* R) if and only if $R^{\mathcal{I}}$ is a function, i.e., if $(c, c') \in R^{\mathcal{I}}$ and $(c, c'') \in R^{\mathcal{I}}$ implies $c' = c''$. Notice that we only consider $DL-Lite$ with unique name assumption.

Note that the interpretation function $\cdot^{\mathcal{I}}$ is extended for \sqcap and \sqcup constructors respectively as follows: $(C_1 \sqcap C_2)^{\mathcal{I}} = C_1^{\mathcal{I}} \cap C_2^{\mathcal{I}}$ and $(C_1 \sqcup C_2)^{\mathcal{I}} = C_1^{\mathcal{I}} \cup C_2^{\mathcal{I}}$.

An interpretation \mathcal{I} is said to satisfy a knowledge base $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ if and only if \mathcal{I} satisfies every axiom in \mathcal{T} and every axiom in \mathcal{A} . Such interpretation is said to be a model of \mathcal{K} .

Incoherence and inconsistency Two kinds of inconsistency can be distinguished in DL-based knowledge bases: *incoherence* and *inconsistency* [Baader *et al.*, 2010]. The former is considered as a kind of inconsistency in the TBox, *i.e.* the terminological part of a knowledge base. The latter is the standard notion of inconsistency of knowledge bases. A knowledge base is said to be inconsistent if and only if it does not admit any model and it is said to be incoherent if there exists at least a non-satisfiable concept (*i.e.* no individual can belong to the concept). More formally:

Definition 1.13. A $DL-Lite$ terminological base \mathcal{T} is said to be incoherent if there exists a concept C such that for each interpretation \mathcal{I} which is a model of \mathcal{T} , we have $C^{\mathcal{I}} = \emptyset$.

Example 1.7. An example of incoherent TBox is the one composed of the two inclusion axioms $\mathcal{T} = \{B_1 \sqsubseteq B_2, B_1 \sqsubseteq \neg B_2\}$. One can easily check that for all models \mathcal{I} of \mathcal{T} we have $B_1^{\mathcal{I}} = \emptyset$. ■

In a propositional setting the counterpart of incoherence is a so-called potential inconsistency, as defined for instance in [Nonfjall and Larsen, 1992].

The concept of knowledge base inconsistency is defined by:

Definition 1.14. A $DL-Lite$ knowledge base $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ is said to be inconsistent if it does not admit any model.

FOL-reducibility An important property, called FOL-reducibility, has been established in [Calvanese *et al.*, 2007a] for consistency checking and query answering in $DL-Lite$. This property reduces reasoning tasks in $DL-Lite$ knowledge base $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ to the evaluation over the ABox of FOL queries obtained from \mathcal{T} . Clearly, such a property separates the TBox and the ABox when reasoning. Namely, the reasoning tasks are done in two steps: The first one consists in producing FOL queries using axioms of the TBox. The second step consists in evaluating the obtained queries over the ABox that can be stored in a relational database, and thus, one can use SQL engines.

It is important to note that for other $DL-Lite$ members that allow more expressivity (presented in Section 1.4.2) than $DL-Lite_{core}$, $DL-Lite_F$ and $DL-Lite_R$, the FOL-reducibility property is not always guaranteed [Artale *et al.*, 2009].

Consistency checking in *DL-Lite* In *DL-Lite* a TBox $\mathcal{T} = \{\mathcal{T}_p, \mathcal{T}_n\}$ can be viewed as composed of a set of positive inclusion axioms (denoted \mathcal{T}_p) and a set of negative inclusion axioms (denoted \mathcal{T}_n). A positive inclusion axiom (PI) is of the form $B_1 \sqsubseteq B_2$ and a negative inclusion axiom (NI) is of the form $B_1 \sqsubseteq \neg B_2$. Intuitively \mathcal{T}_p specifies inclusion dependencies, while \mathcal{T}_n defines integrity constraints.

The *DL-Lite* logics, and in particular *DL-Lite_{core}*, *DL-Lite_F* and *DL-Lite_R*, enjoy the canonical model property [Calvanese et al., 2007a]. This property states that given a consistent *DL-Lite* knowledge base \mathcal{K} , one can construct a single model \mathcal{I}_c of \mathcal{K} so that any other model \mathcal{I} of \mathcal{K} can be obtained from \mathcal{I}_c . This model is called canonical model and defined through the notion of *Chase* [Abiteboul et al., 1995]. Using the notion of canonical interpretation, it was shown that a knowledge base that only contains PIs in its ABox is always consistent [Calvanese et al., 2007a]. Inconsistency is caused by NI axioms. Note that in query answering, the canonical interpretation allows to find the correct answers of queries.

***DL-Lite* deductive closure** The negative closure of \mathcal{T} , denoted by $cln(\mathcal{T})$, performs interaction between positive and negative axioms. It represents the propagation of the negative axioms using both positive axioms and negative axioms in the TBox. For *DL-Lite_{core}*, the $cln(\mathcal{T})$ is obtained using the following rules repeatedly until reaching a fixed point (see [Calvanese et al., 2007a] for more details):

1. All negative axioms in \mathcal{T} are in $cln(\mathcal{T})$.
2. If $B_1 \sqsubseteq B_2$ is in \mathcal{T} and $B_2 \sqsubseteq \neg B_3$ is in $cln(\mathcal{T})$, then $B_1 \sqsubseteq \neg B_3$ is in $cln(\mathcal{T})$.
3. If $B_1 \sqsubseteq B_2$ is in \mathcal{T} and $B_3 \sqsubseteq \neg B_2$ is in $cln(\mathcal{T})$, then $B_1 \sqsubseteq \neg B_3$ is in $cln(\mathcal{T})$.

For the *DL-Lite_R* and *DL-Lite_F* logics, we need the following additional rules:

4. All functionality axioms in \mathcal{T} are also in $cln(\mathcal{T})$.
5. If $R_1 \sqsubseteq R_2$ is in \mathcal{T} and $\exists R_2 \sqsubseteq \neg B$ or $B \sqsubseteq \neg \exists R_2$ is in $cln(\mathcal{T})$, then $\exists R_1 \sqsubseteq \neg B$ is in $cln(\mathcal{T})$;
6. If $R_1 \sqsubseteq R_2$ is in \mathcal{T} and $\exists R_2^- \sqsubseteq \neg B$ or $B \sqsubseteq \neg \exists R_2^-$ is in $cln(\mathcal{T})$, then $\exists R_1^- \sqsubseteq \neg B$ is in $cln(\mathcal{T})$;
7. If $R_1 \sqsubseteq R_2$ is in \mathcal{T} and $R_2 \sqsubseteq \neg R_3$ or $R_3 \sqsubseteq \neg R_2$ is in $cln(\mathcal{T})$, then $R_1 \sqsubseteq \neg R_3$ is in $cln(\mathcal{T})$;
8. (a) in the case where \mathcal{T} is a *DL-Lite_R* TBox, if one of the axioms $\exists R \sqsubseteq \neg \exists R$, $\exists R^- \sqsubseteq \neg \exists R^-$ or $R \sqsubseteq \neg R$ is in $cln(\mathcal{T})$, then all these three axioms are in $cln(\mathcal{T})$.
 (b) in the case where \mathcal{T} is a *DL-Lite_F* TBox, if one of the axioms $\exists R \sqsubseteq \neg \exists R$, $\exists R^- \sqsubseteq \neg \exists R^-$ is in $cln(\mathcal{T})$, then both such axioms are in $cln(\mathcal{T})$.

Example 1.8. From the *DL-Lite_{core}* TBox given in Example 1.6, one can derive the following negated closure:

$$\begin{aligned} Teacher &\sqsubseteq \neg Student \\ \exists TeachesTo^- &\sqsubseteq \neg Teacher \\ \exists HasSupervisor^- &\sqsubseteq \neg Student \end{aligned}$$

The negated closure of the *DL-Lite_R* TBox is constituted by adding the following axioms to the *DL-Lite_{core}* negated closure:

$$\begin{aligned} \exists TeachesTo &\sqsubseteq \neg Student \\ \exists HasSupervisor &\sqsubseteq \neg Teacher \end{aligned}$$

The negated closure of the $DL-Lite_F$ TBox is obtained by adding the following axiom to the $DL-Lite_{core}$ negated closure :

$$(funct HasSupervisor)$$

■

Formally, $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ is consistent if and only if $\langle \text{cln}(\mathcal{T}), \mathcal{A} \rangle$ is consistent [Calvanese et al., 2007a]. In fact, this is a consequence of the property of FOL reducibility. Namely, it has been shown in [Calvanese et al., 2007a] that consistency checking can be reduced to evaluating FOL queries (called Unsat queries) over the ABox which may be considered as a relational database. Table 1.5 summarizes transformations from NI axioms to Unsat queries.

NI axiom	Unsat query
$A_1 \sqsubseteq \neg A_2$	$\exists x. A_1(x) \wedge A_2(x)$
$A \sqsubseteq \neg \exists P$ or $\exists P \sqsubseteq \neg A$	$\exists x. A(x) \wedge \exists y. P(x, y)$
$A \sqsubseteq \neg \exists P^-$ or $\exists P^- \sqsubseteq \neg A$	$\exists x. A(x) \wedge \exists y. P(y, x)$
$\exists P_1 \sqsubseteq \neg \exists P_2$	$\exists x. \exists y. P_1(x, y) \wedge \exists z. P_2(x, z)$
$\exists P_1 \sqsubseteq \neg \exists P_2^-$ or $\exists P_1^- \sqsubseteq \neg \exists P_2$	$\exists x. \exists y. P_1(x, y) \wedge \exists z. P_2(z, x)$
$\exists P_1^- \sqsubseteq \neg \exists P_2^-$	$\exists x. \exists y. P_1(y, x) \wedge \exists z. P_2(z, x)$
$P_1 \sqsubseteq \neg P_2$ or $P_1^- \sqsubseteq \neg P_2^-$	$\exists x, y. P_1(x, y) \wedge P_2(x, y)$
$P_1 \sqsubseteq \neg P_2^-$ or $P_1^- \sqsubseteq \neg P_2$	$\exists x, y. P_1(x, y) \wedge P_2(y, x)$
$(funct P)$	$\exists x, y, z. P(x, y) \wedge P(x, z) \wedge y \neq z$
$(funct P^-)$	$\exists x, y, z. P(y, x) \wedge P(z, x) \wedge y \neq z$

Table 1.5: Transformation of the negative inclusion axioms to unsat queries

Example 1.9. From Example 1.8, the set of queries associated with the $DL-Lite_{core}$ negated closure is as follows:

$$\begin{aligned} q_1(x) &= \exists x. Teacher(x) \wedge Student(x) \\ q_2(x) &= \exists x. \exists y. TeachesTo(y, x) \wedge Professor(x) \\ q_3(x) &= \exists x. \exists y. HasSupervisor(y, x) \wedge Student(x) \end{aligned}$$

For the $DL-Lite_R$ negated closure, we add the following queries:

$$\begin{aligned} q_4(x) &= \exists x. \exists y. TeachesTo(x, y) \wedge Student(x) \\ q_5(x) &= \exists x. \exists y. HasSupervisor(x, y) \wedge Teacher(x) \end{aligned}$$

For the $DL-Lite_F$ negated closure, we add the following query:

$$q_4(x) = \exists x, y, z. HasSupervisor(x, y) \wedge HasSupervisor(x, z) \wedge y \neq z$$

■

Queries and certain answers over *DL-Lite* An n -ary query is an open formula of first-order logic with equality of the form

$$q = \{\vec{x} \mid \phi(\vec{x})\},$$

where $\phi(\vec{x})$ is a FOL formula with free variables $\vec{x} = x_1, \dots, x_n$ (called also answer variables) and the arity n of q is the number of its free variables. When $n = 0$, the query is said to be a boolean or ground query. A boolean query of the form $q = \{ \mid \phi \}$ is a query that does not involve free variables (*i.e.* with no answer variables).

Given an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, a boolean query is either interpreted as true in \mathcal{I} if $[\phi]^{\mathcal{I}} = true$ or false if $[\phi]^{\mathcal{I}} = false$. Indeed, the answer to such a query is either "yes" or "no". When $n > 0$, a non-boolean query q is interpreted as the set of tuples of the domain elements, called answer sets with respect to \mathcal{I} , such that if we substitute \vec{x} by an answer set \vec{a} the query q will be evaluated to true in \mathcal{I} . Namely $q^{\mathcal{I}} = \{\vec{a}_i \in (\Delta^{\mathcal{I}})^n \mid [\phi(\vec{a}_i)]^{\mathcal{I}} = true\}$. An interpretation that evaluates a boolean query (*resp.* non-boolean query) to true (*resp.* to a non empty answer set), is said to be a model of that query, written $\mathcal{I} \models q$.

In *DL-Lite*, the more interesting queries are the class of conjunctive queries and the class of union of conjunctive queries. A Conjunctive Query (CQ) is a query of the form:

$$q = \{\vec{x} \mid \exists \vec{y}. conj(\vec{x}, \vec{y})\},$$

where \vec{x} are free variables called distinguished or answer variables, \vec{y} are existentially quantified variables called non-distinguished or bounded variables, and $conj(\vec{x}, \vec{y})$ is a conjunction of atoms of the form $A(t_i)$ or $P(t_i, t_j)$ and equalities, where the predicates A and P are respectively an atomic concept name and an atomic role name appearing in \mathcal{K} , and t_i, t_j are terms, *i.e.* constants (individuals) in \mathcal{A} or variables in \vec{x} or \vec{y} . Notice that we call instance query the one consisting of a single atom with no free variable, namely an ABox assertion. A Union of Conjunctive Query (UCQ) denoted by Q is simply an expression of the form:

$$Q = \{\vec{x} \mid \bigvee_{i=1, \dots, n} \exists \vec{y}_i. conj(\vec{x}, \vec{y}_i)\}.$$

where each $conj(\vec{x}, \vec{y}_i)$ is a conjunction of atoms and equalities with answer variables \vec{x} and bound variable \vec{y}_i . Obviously, the class of UCQ contains the one of conjunctive queries.

Given $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ a *DL-Lite* knowledge base and a CQ q , we write $\mathcal{K} \models q$ when $\mathcal{I} \models q$ for all models \mathcal{I} of \mathcal{K} , otherwise $\mathcal{K} \not\models q$. The answer to q over \mathcal{K} , denoted $ans(q, \mathcal{K})$, is the set of tuples of constants appearing in \mathcal{K} such that $\forall \vec{a}_i: \vec{a}_i^{\mathcal{I}} \in q^{\mathcal{I}}$, for every model \mathcal{I} of \mathcal{K} . Namely $ans(q, \mathcal{K}) = \{\vec{a}_i \in (\mathcal{K})^n \mid \mathcal{K} \models q(\vec{a}_i)\}$ where $q(\vec{a}_i)$ is the closed formula obtained by replacing the answer variables \vec{x} in q by an answer set \vec{a}_i , and $\mathcal{K} \models q(\vec{a}_i)$ means that every model of \mathcal{K} is also model of $q(\vec{a}_i)$. This corresponds to the well-known certain answers semantics defined in [Artale *et al.*, 2009; Calvanese *et al.*, 2007a]. Given $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ a *DL-Lite* knowledge base and a CQ q , a certain answer to q over \mathcal{K} is an answer that holds in all the models satisfying \mathcal{K} .

It is important to note that CQ answering can be reduced to boolean query answering. Namely, given a CQ q with free variables $\vec{x} = \{x_1, \dots, x_n\}$, an answer set $\vec{a} = \{a_1, \dots, a_n\}$ is a certain answer for q over \mathcal{K} if the boolean query $q(\vec{a})$ obtained by replacing each variable x_i by a_i in $q(\vec{x})$, evaluates to true for every model of \mathcal{K} . Lastly, if \mathcal{K} is inconsistent, then $ans(q, \mathcal{K})$ is trivially the set of all possible answer sets, denoted $AllUp(q, \mathcal{K})$.

DL-Lite Reasoner QuOnto¹⁸ is a free reasoner for *DL-Lite* developed in Java. It implements a query rewriting algorithm for both consistency checking and query answering for unions of conjunctive queries over *DL-Lite* knowledge bases where the ABox is managed using a relational database.

1.4.2 The extended *DL-Lite* family

We now introduce the extended *DL-Lite* family of description logics proposed with the aim of capturing typical conceptual modeling formalisms, such as UML class diagrams and Entity-Relation models, while maintaining good computational properties of standard DL reasoning tasks [Artale *et al.*, 2009]. For more details, see the original paper [Artale *et al.*, 2009].

As usual, let N_C , N_R and N_I respectively be pairwise disjoint sets of concepts, roles and individuals names. Let $A \in N_C$, $P \in N_R$ and $a \in N_I$. The syntax of the extended family of *DL-Lite* is composed of $DL-Lite_\alpha^\beta$ logics where
 $\alpha = \{core, krom, horn, bool\}$ and
 $\beta = \{-, \mathcal{H}, \mathcal{F}, \mathcal{N}, \mathcal{HF}, \mathcal{HN}, (\mathcal{HF})^+, (\mathcal{HN})^+, (\mathcal{HF})^+, (\mathcal{HN})^+\}$,
 is defined using the following syntax:

$$\begin{aligned} R &\longrightarrow P \mid P^- \\ B &\longrightarrow \perp \mid A_n \mid \geq zR \\ C &\longrightarrow B \mid \neg C \mid C_1 \sqcap C_2 \end{aligned}$$

with $z \in \mathbb{N}$, and $\top = \neg\perp$, $C_1 \sqcup C_2 = \neg(\neg C_1 \sqcap \neg C_2)$, $\exists R = \exists R.\top = (\geq 1R)$, $\leq zR = \neg(\geq z+1R)$. \mathcal{H} denotes role hierarchies (*i.e.* role inclusion axioms), \mathcal{F} denotes functionality (*i.e.* $(Funct R)$), and \mathcal{N} denotes number restriction $\geq zR$. The semantics of \mathcal{N} , \mathcal{H} and $(Funct R)$ is given in Table 1.2.

Let \mathcal{L}_α be *DL-Lite* $_\alpha$ language where $\alpha = \{core, krom, horn, bool\}$. Table 1.6 gives the forms of concept inclusion axioms and assertions that can be allowed in a *DL-Lite* knowledge base $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ expressed using \mathcal{L}_α .

	$DL-Lite_{bool}$	$DL-Lite_{krom}$	$DL-Lite_{horn}$	$DL-Lite_{core}$
TBox	$C_1 \sqsubseteq C_2$	$B_1 \sqsubseteq B_2, B_1 \sqsubseteq \neg B_2, \neg B_1 \sqsubseteq B_2$	$\sqcap_n B_n \sqsubseteq B$	$B_1 \sqsubseteq B_2, B_1 \sqsubseteq \neg B_2$
ABox	$A(a), \neg A(a), P(a, b), \neg P(a, b)$			

Table 1.6: Concept inclusion axioms and assertions in extended *DL-Lite* logics.

There is a tight relationship between the different *DL-Lite* members. The $DL-Lite_{bool}^{HN}$ logic is considered as the supremum (most expressive) of all above logics. The most basic one (least expressive) is the $DL-Lite_{core}$ logic (presented in Section 1.4.1). Indeed, $DL-Lite_{core}$ is situated in the intersection of $DL-Lite_{krom}$ and $DL-Lite_{horn}$ since $B_1 \sqsubseteq \neg B_2$ is equivalent to $B_1 \sqcap B_2 \sqsubseteq \perp$. Moreover, $DL-Lite_\alpha^H$ logics are considered as fragments of $DL-Lite_\alpha^{HF}$ where these latter are considered as fragments of $DL-Lite_\alpha^{HN}$ (since from $\geq zR$, one can express $\exists R$ for $z = 1$ and $(funct R)$ (*resp.* $(funct R^-)$) as $\geq 2R \sqsubseteq \perp$ (*resp.* $\geq 2R^- \sqsubseteq \perp$) for $z = 2$). Lastly, the TBox's of $DL-Lite_\alpha^{(HN)}$, $DL-Lite_\alpha^{(HN)^+}$, $DL-Lite_\alpha^{(HF)}$ and $DL-Lite_\alpha^{(HF)^+}$ can contain role constraints of the form:

¹⁸<http://www.dis.uniroma1.it/~quonto/>

- Transitivity ($tra(R)$): $\mathcal{I} \models tra(R)$ if and only if $(x, y) \in R^{\mathcal{I}} \wedge (y, z) \in R^{\mathcal{I}} \rightarrow (x, z) \in R^{\mathcal{I}}$.
- Disjointness ($dis(R, S)$): $\mathcal{I} \models dis(R_1, S_2)$ if and only if $R_1^{\mathcal{I}} \cap R_2^{\mathcal{I}} = \emptyset$.
- Reflexivity ($ref(R)$): $\mathcal{I} \models ref(R)$ if and only if $(x, x) \in R^{\mathcal{I}}$ for all $x \in \Delta^{\mathcal{I}}$.
- Irreflexivity ($irr(R)$): $\mathcal{I} \models irr(R)$ if and only if $(x, x) \notin R^{\mathcal{I}}$ for all $x \in \Delta^{\mathcal{I}}$.
- Symmetry ($sym(R)$): $\mathcal{I} \models sym(R)$ if and only if $R^{\mathcal{I}} = (R^{-})^{\mathcal{I}}$.
- Asymmetry ($asy(R)$): $\mathcal{I} \models asy(R)$ if and only if $R^{\mathcal{I}} \cap (R^{-})^{\mathcal{I}} = \emptyset$.

Table 1.7 summarizes the different members of the extended *DL-Lite* family.

Role inclusion		Number restriction	Constraints on roles
No	Yes		
$DL-Lite_{\alpha}$	$DL-Lite_{\alpha}^H$	$\exists R$	No
$DL-Lite_{\alpha}^F$	$DL-Lite_{\alpha}^{HF}$	$\exists R / funct$	No
$DL-Lite_{\alpha}^N$	$DL-Lite_{\alpha}^{HN}$	$\geq zR$	No
	$DL-Lite_{\alpha}^{(HF)}$	$\exists R.D / funct (c)$	dis, (a)sym, (ir)ref
	$DL-Lite_{\alpha}^{(HN)}$	$\geq zR.D (c)$	dis, (a)sym, (ir)ref
	$DL-Lite_{\alpha}^{(HF)^+}$	$\exists R.D / funct$	dis, (a)sym, (ir)ref, tra
	$DL-Lite_{\alpha}^{(HN)^+}$	$\geq zR.D$	dis, (a)sym, (ir)ref, tra

Table 1.7: The extended *DL-Lite* family

Note that in Table 1.7, only $DL-Lite_{\alpha}^{(HN)^+}$ and $DL-Lite_{\alpha}^{(HF)^+}$ allow role transitivity constraint. Moreover $DL-Lite_{\alpha}^{(HN)}$ and $DL-Lite_{\alpha}^{(HF)}$ are restricted by the following constraints [Artale et al., 2009]:

- \mathcal{T} may contain only positive occurrences of qualified number restrictions $\geq zR.C$, where C is a conjunction of concepts allowed in the right-hand side of $DL-Lite_{\alpha}$ concept inclusions.
- if $\geq zR.C$ occurs in \mathcal{T} , then \mathcal{T} does not contain negative occurrences of number restrictions $\geq z'R$ or $\geq z'R^{-}$ with $z' \geq 2$;
- if R has a proper sub-role in \mathcal{T} , then \mathcal{T} does not contain negative occurrences of $\geq zR$ or $\geq zR^{-}$ with $z \geq 2$.

In fact, the above restrictions limit the interaction between role inclusions and number restrictions in each $DL-Lite_{\alpha}$ TBox in order to reduce the complexity of reasoning and allow the use of the above role constraints which increase the expressive power of the logics but do not affect their computational properties.

Table 1.9 reviews main computational complexity results of the different logics of the *DL-Lite* family. For a more detailed description on *DL-Lite* family, see [Artale et al., 2009].

Languages	Complexity		
	Combined Complexity	Data Complexity	
	Satisfiability	Instance checking	Query Answering
$DL-Lite_{core}^{[H]}$	NLogSpace	AC^0	AC^0
$DL-Lite_{horn}^{[H]}$	P	AC^0	AC^0
$DL-Lite_{krom}^{[H]}$	NLogSpace	AC^0	coNP
$DL-Lite_{bool}^{[H]}$	NP	AC^0	coNP
$DL-Lite_{core}^{[F N](HF) (HN)}$	NLogSpace	AC^0	AC^0
$DL-Lite_{horn}^{[F N](HF) (HN)}$	P	AC^0	AC^0
$DL-Lite_{krom}^{[F N](HF) (HN)}$	NLogSpace	AC^0	coNP
$DL-Lite_{bool}^{[F N](HF) (HN)}$	NP	AC^0	coNP
$DL-Lite_{core/horn}^{HF}$	EXPTIME	P	P
$DL-Lite_{krom/bool}^{HF}$	EXPTIME	coNP	coNP
$DL-Lite_{core/horn}^{HN}$	EXPTIME	coNP	coNP
$DL-Lite_{krom/bool}^{HN}$	EXPTIME	coNP	coNP

Table 1.8: Complexity of reasoning in $DL-Lite$ logics (with unique name assumption)

Note that $DL-Lite_{core}^H$ (resp. $DL-Lite_{core}^F$) language is the $DL-Lite_R$ (resp. $DL-Lite_F$) language presented in section 1.4. The following table summarizes their complexity.

Languages	Complexity		
	Combined Complexity	Data Complexity	
	Satisfiability	Instance checking	Query Answering
$DL-Lite_{core}$	NLogSpace	AC^0	AC^0
$DL-Lite_R$	NLogSpace	AC^0	AC^0
$DL-Lite_F$	NLogSpace	AC^0	AC^0

Table 1.9: Complexity of reasoning in $DL-Lite_{core}$, $DL-Lite_R$ and $DL-Lite_F$

Recall that, in this thesis we only consider $DL-Lite_{core}$, $DL-Lite_F$ and $DL-Lite_R$ logics. For the sake of simplicity and when there is no ambiguity, we use $DL-Lite$ to refer to these three fragments.

1.5 Conclusion

In this chapter, we presented description logics, as decidable fragments of first order logics, that offer a nice logical framework to serve ontologies. We focused on three main members of the $DL-Lite$ family

investigated in this thesis: $DL-Lite_{core}$, $DL-Lite_F$ and $DL-Lite_R$ which underly the OWL2-QL profile especially dedicated for applications using large data.

In real world applications, knowledge and data are usually affected with uncertainty and imprecision. Moreover, knowledge evolves from a situation to another or may be issued from different information sources. As pointed out in Section 1.2, merging, evolution and inconsistency and uncertainty management in ontologies are recognized as challenging problems. Next chapter will focus on these issues and will provide an overview on different techniques and tools proposed in the literature, especially in a propositional logic setting, to handle them.

BELIEF CHANGE AND UNCERTAINTY MANAGEMENT

2.1 Introduction

Originally ontologies have been proposed to represent the knowledge of a domain of interest [Baader *et al.*, 2010] in a static form. However, in some applications (like Web-based ones), the knowledge may be non static and may evolve and change from one situation to another in order to take into account and integrate the changes that occur over time. This dynamic aspect of ontologies is closely related to the belief revision problem studied within propositional logic frameworks (*e.g.* [Alchourrón *et al.*, 1985; Katsuno and Mendelzon, 1991]). Moreover, in some Web applications, knowledge may come from different and often conflicting sources of information where aggregating them in order to provide a global point of view, is required. Merging different pieces of information is also largely studied within a propositional logic setting (*e.g.* [Bloch *et al.*, 2001; Konieczny and Pino Pérez, 2002]).

In the artificial intelligence community, it is well-known that nonmonotonic reasoning and revision are considered as the two sides of the same coin [Makinson and Gärdenfors, 1989]. In general, revision or merging of different information sources may lead to inconsistency problems where several approaches based on either restoring the consistency of the knowledge base in order to exploit it and perform inferences (*e.g.* [Benferhat *et al.*, 1997a]), or analyzing inconsistency using different measures (*e.g.* [Hunter and Konieczny, 2005]), or using argumentation framework (*e.g.* [Besnard and Hunter, 2008]) to make decisions, were proposed.

Regarding the quality of the information, it can be affected with uncertainty and imprecision. This is due for instance to the reliability of sources that provide them. In general, information qualified as imperfect may be of different forms: incomplete information, heterogeneous information, incommensurable information, imprecise information, uncertain information, etc. Faithfully handling such pieces of information and taking them into account when reasoning is an important issue that arises in many Web applications. Probability theory is the oldest and the most widely used theory for handling uncertain information. This latter is suitable especially within a frequentist setting. Moreover, several non-classical probabilistic and non-probabilistic theories for handling uncertainties and imprecisions have been proposed. The most well-known ones are fuzzy set theory [Zadeh, 1965; Zadeh, 1988], Dempster-Shafer theory of evidence (or belief functions) [Shafer, 1976], the Spohn's ordinal conditional functions [Spohn, 1988], and possibility theory [Zadeh, 1978; Dubois and Prade, 1988b].

Choosing the right and appropriate framework to represent and reason under imperfect information is closely related to the context of the applications. In this thesis, we focus on possibility theory [Dubois and Prade, 1988b] which is a very natural framework to deal with ordinal and qualitative uncertainty. It deals with non-probabilistic information and it is particularly appropriate when the uncertainty scale only reflects a priority relation between different pieces of information. For instance, the choice of possibility theory in our context, can be justified in an Ontology-Based Data Access setting in which as presented in Section 1.2 data may be provided by different sources which can have different levels of priority.

This chapter aims to provide an overview about belief change and uncertainty management from a

propositional logic point of view while giving the related works done in description logics in order to situate our works.

The rest of the chapter is organized as follows: Section 2.2 reviews the notion of uncertainty and imprecision and recalls basic concepts of probability theory and provides a refresher on possibility theory. Section 2.3 recalls the context of merging, revision and uncertainty handling and presents relevant works done in description logics. Section 2.4 concludes this chapter.

2.2 Uncertainty management

An information, in a broad sense, refers to any collection of symbols or signs produced either through the observation of natural or artificial phenomena or by cognitive human activity [Dubois and Prade, 2010]. A piece of information, can be of different forms (objective, subjective, quantitative, qualitative, singular, generic, etc) [Dubois and Prade, 2011] and can be affected with different kinds of imperfection (incompleteness, confusion, irrelevance, imprecision, vagueness, etc) which are considered as forms of ignorance [Parsons and Hunter, 1998].

Throughout this section, we use the following notations.

- $\Omega = \{\omega_1, \dots, \omega_n\}$: Denotes the set of the states of the world, called the universe of discourse.
- $\omega_i \in \Omega$: Denotes a state of the world, called an interpretation or elementary event.
- A, B, \dots, E : Capital letters denote subsets of Ω . A subset $A \subseteq \Omega$ is called an event.
- v : Denotes a vector of variables where Ω is its domain.

A subset A of Ω is considered as a disjunctive set and it is viewed as a proposition that asserts a variable v in A . However, the propositions expressible on Ω may be attached with imperfections as said before. In the following, we present pieces of information qualified as incomplete.

Incomplete information. A piece of information is said to be incomplete (or partial) in a given context, if it is not sufficient enough to answer a relevant question asked in the same context. For instance, one can consider the following examples:

Example 2.1. The following gives examples of incomplete information.

1. Consider the following question: "What is exactly the age of Paul?". Let "Paul was born between 1980 and 1984" be an information that one knows. Such information does not allow to answer the above question.
2. Consider the following question: "What is the temperature of the patient ?". Let "The temperature of the patient is high" be an information that one has. Such information does not allow to precisely answer the question about the exact temperature.
3. Consider the following question: "Does student Paul succeed this year?". Let "The success rate for this year is about 50%" be an information about the known success rate. Such information is not enough to answer the asked question.

■

From Example 2.1, one can see that the nature of an incomplete information is not the same. Incompleteness can be, in general, imprecision, fuzziness, uncertainty, etc. For more details, see [Parsons and Hunter, 1998; Dubois and Prade, 2011] for example.

Imprecise information Given a proposition that asserts a variable v in $A \subseteq \Omega$, a piece of information is said to be imprecise, if it is insufficient to give the current value of the variable v in A . Imprecision is related to the content of the information. Said differently, the number of elements in A , that may correspond to v is greater than 1. Note that v takes only one value from A at a given time, and thus, elements in A are considered as possible values of v and they are mutually exclusive. Let us consider the following example.

Example 2.2. From Example 2.1 item 1, the quantity $v = \text{birthyear}(\text{Paul}) \in \{1980, 1981, 1982, 1983, 1984\}$ which states that "Paul" was born between 1980 and 1984". This leads to consider that $v = 1980$ or $v = 1981$ or $v = 1982$ or $v = 1983$ or $v = 1984$. ■

Fuzzy information A fuzzy piece of information (or gradual linguistic piece of information) is considered as a subtype of imprecision. It represents a proposition asserting a variable v in $A \subseteq \Omega$ where one can not claim if it is totally true or totally false. Namely, the proposition is not boolean.

Example 2.3. From Example 2.1 item 2, the proposition stating that "The temperature of the patient is high" is fuzzy since it does not give the exact value of the temperature. Said differently, we only know that it is "high". Indeed, a temperature equal to 42 looks more credible than a temperature equal to 40 which is itself more credible than a temperature equal to 39. However, saying that "the temperature is equal to 37" is completely false referring to the context of the information that said that the temperature is high. In this case, one would rather say that the temperature of the patient is normal. ■

In fact, a fuzzy piece of information ranks values in A in terms of their relevance to give the current value of v [Dubois and Prade, 2011]. Note that the meaning of a fuzzy information may be altered using linguistic quantifiers expressing intensity, for instance, consider the proposition "the temperature is very high" or "the temperature is slightly high", etc.

Uncertain information A piece of information is said to be uncertain when one can not decide if the information is completely true or completely false (for instance the question asked in item 3 of Example 2.1). Uncertainty is due either to variability (randomness) or lack of information about the real world, and it is in general related to the source providing the information. An uncertain piece of information is attached with a certainty qualifier which can be numerical (e.g. a probability) or symbolic (e.g. plausible).

Example 2.4. Let us consider the following information pieces:

- The probability that the task takes more than one hour is about 0.7.
- It is very possible that it will rain tomorrow.
- It is not absolutely certain that Paul will come to the meeting tomorrow.

■

Usually, an uncertain piece of information is represented by attaching (using a function f defined over Ω) to each event $A \subseteq \Omega$ a number $f(A)$ belonging to the unit interval $[0, 1]$ which evaluates the likelihood of A with respect to a proposition asserting $v \in A$. In other words, $f(A)$ is the confidence of the agent in the truth of $v \in A$. Note that, contrary to fuzzy information, the proposition is boolean, namely it only takes true or false. When dealing with uncertainty, the following requirements are needed:

1. $f(\Omega) = 1$ and $f(\emptyset) = 0$.
2. $\forall A \subseteq \Omega, \forall B \subseteq \Omega : \text{if } A \subseteq B \text{ then } f(A) \leq f(B)$ (Monotonicity).

$$3. \forall A \subseteq \Omega, \forall B \subseteq \Omega : g(A \cap B) \leq \min(f(A), f(B)).$$

$$4. \forall A \subseteq \Omega, \forall B \subseteq \Omega : g(A \cup B) \geq \max(f(A), f(B)).$$

Example 2.5. From Example 2.1, one can be completely certain that the birth year of Paul is between 1980 and 1984, but less certain that Paul's birth date is in $\{1980, 1981, 1982\}$, even less certain that it is in 1981. ■

A piece of information could be at the same time:

- Precise and certain: "Paul was born in 1983".
- Precise and uncertain: "It is probable that Paul was born in 1984, with a confidence degree of 30%".
- Imprecise and certain: "It is certain that Paul was born between 1982 and 1984".
- Imprecise and uncertain: "It is possible that Paul was born between 1981 and 1984".

Lastly, there are other kinds of imperfect pieces of information such as heterogeneous information (*i.e.* information having different nature or expressed differently), incommensurable information (*i.e.* information expressed over different scales), multiple source information (*i.e.* information provided by different sources), etc.

2.2.1 Probability theory

Probability theory is the oldest and the most widely acknowledged among uncertainty theories. This section recalls basic concepts of probability theory.

Basic notions

The notion of probability distribution is the central element of probability theory. A probability distribution, denoted by p , is defined over the universe of discourse Ω such that $\forall \omega_i \in \Omega, p(\omega_i) \in [0, 1]$. A probability measure $P : 2^\Omega \rightarrow [0, 1]$ is a function that assigns to each event $A \subseteq \Omega$ a degree belonging to the unit interval $[0, 1]$. This degree reflects the chance or the likelihood of the occurrence or the realization of A . A probability measure and probability distribution satisfy the following requirements:

- Positivity: $\forall A \subseteq \Omega, P(A) \geq 0$.
- Normalization : $P(\Omega) = 1$
- Additivity: $P(A \cup B) = P(A) + P(B)$ (if A and B are disjoint, namely $A \cap B = \emptyset$).

The first axiom states that an event of Ω may be (in the worst case) impossible, *i.e.* $P(A) = 0$. The second states that the universe of discourse Ω is certain. Finally, the third axiom states that the probability of the union of two disjoint events is equal to the sum of the probabilities of the two events separately. As consequence of the above requirements, we have:

- $\sum_{i=1}^n p(\omega_i) = 1$,
- $P(\emptyset) = 0$
- $P(A) = 1 - P(\bar{A})$ (where \bar{A} is the complementary of A in Ω , namely $\bar{A} = \Omega \setminus A$)

- $P(A \cup B) = P(A) + P(B) - P(A \cap B)$ where $A \in \Omega$ and $B \in \Omega$.

Given a probability distribution p defined on Ω , one can derive the probability measure of a subset $A \subseteq \Omega$ as follows:

$$P(A) = \sum_{\omega \in A} p(\omega)$$

Let B be an event of Ω . The probability of the realization of an event A can be updated according to the probability of the realization of the event B . Let $P(A|B)$ denote the probability of the event A knowing the event B . The transformation from $P(A)$ to $P(A|B)$ is called probabilistic conditioning and it is computed as follows:

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

The following property is called product rule and it is defined by (it is a consequence of the conditioning rule):

$$P(A \cap B) = P(A) * P(B|A) = P(B) * P(A|B)$$

Bayes theorem states that the conditional probability of an event A given B is related to the converse conditional probability of B given A . This permits to compute the probability of B if we know A as follows:

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)}$$

Bayes rule is very useful when performing inference like in classification problems, or learning from statistical data.

2.2.2 Possibility theory

Possibility theory, introduced first by Zadeh [Zadeh, 1999] and then developed by Dubois and Prade [Dubois and Prade, 1988b] and many other researchers, is a very natural framework to deal with ordinal and qualitative uncertainty. It deals with non-probabilistic information and it is particularly appropriate when the uncertainty scale only reflects a priority relation between different pieces of information. There are several interpretations of possibility degrees. The most supported ones are as follows [Dubois and Prade, 2011] :

- The feasibility or realizability, for instance "it is possible to repair the old car".
- The plausibility which refers to the degree to which an event can occur, for instance "it is possible that it will snow tomorrow".
- Consistency or compatibility which refers to a logical view of possibility and concerns the available information itself, for instance "it is impossible that Paul votes", knowing that "Paul is two years old".

Basic concepts

This section introduces basic concepts of possibility theory. We first recall the notion of possibility distribution.

Possibility distribution. A possibility distribution, denoted by π , is a mapping from the universe of discourse Ω to a totally ordered scale O . This scale may often be a finite set of integers or the unit interval $[0, 1]$ and encodes our knowledge on the real world. In general, it is the interval $[0, 1]$ and it could be interpreted in two ways: a numerical interpretation when values have a real sense and an ordinal interpretation when values only reflect a total pre-order between the different states of the world. We further explain these two settings in Section 2.2.2.

The degree $\pi(\omega)$ is called possibility degree and represents the plausibility or compatibility of ω with available knowledge encoded by π . By convention, when $\pi(\omega) = 1$, we say that ω is a totally possible state, and when $\pi(\omega) = 0$, we say that ω is an impossible state. Given two states of the world ω and ω' , if $\pi(\omega) > \pi(\omega')$, we say that ω is more preferred than ω' or more plausible. Possibility theory can capture the two extreme forms of knowledge, namely:

1. Total ignorance when $\forall \omega \in \Omega, \pi(\omega) = 1$.
2. Complete knowledge when $\exists \omega' \in \Omega, \pi(\omega') = 1$ and $\forall \omega \in \Omega, \omega' \neq \omega, \pi(\omega) = 0$.

Example 2.6. The following possibility distributions give situations of total certainty, partial ignorance and total ignorance.

ω_i	$\pi(\omega_i)$
ω_1	1
ω_2	0
ω_3	0

(a) Total certainty

ω_i	$\pi(\omega_i)$
ω_1	1
ω_2	1
ω_3	.5

(b) Partial ignorance

ω_i	$\pi(\omega_i)$
ω_1	1
ω_2	1
ω_3	1

(c) Total ignorance

■

A possibility distribution is said to be normalized if it admits at least one totally possible state, namely $\exists \omega \in \Omega$ such that $\pi(\omega) = 1$. Otherwise the possibility distribution is said to be sub-normalized. In this case, the inconsistency degree of the possibility distribution π , denoted $Inc(\pi)$, is defined as follows:

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

The concept of sub-normalization reflects the presence of contradictions in the set of available knowledge encoded by π . Lastly, possibility theory is driven by the principle of minimal specificity that states that any hypothesis not known to be impossible cannot be ruled out [Yager, 1992]. Given two possibility distributions π and π' , π is said to be more specific than π' if and only if $\forall \omega \in \Omega, \pi(\omega) \leq \pi'(\omega)$.

Example 2.7. Consider the following two possibility distributions:

ω_i	$\pi_1(\omega_i)$
ω_1	1
ω_2	.1
ω_3	0

ω_i	$\pi_2(\omega_i)$
ω_1	1
ω_2	.5
ω_3	.8

One can check that π_1 is more specific than π_2 .

■

Possibility and Necessity measures. Possibility theory offers two measures to assess the possibility (or the plausibility) and the necessity (or the certainty) of an event.

Possibility measure Given a possibility distribution π , a possibility measure, denoted by Π , of an event $A \subseteq \Omega$ is defined as follows:

$$\Pi(A) = \max_{\omega \in A} (\pi(\omega))$$

Intuitively, $\Pi(A)$ evaluates to what extent A is plausible or compatible with the available knowledge expressed by π . We have:

- if $\Pi(A) = 1$ and $\Pi(\bar{A}) = 0$: this means that A is certain.
- if $\Pi(A) = 1$ and $0 < \Pi(\bar{A}) < 1$: this means that A is somewhat certain.
- if $\Pi(A) = 1$ and $\Pi(\bar{A}) = 1$: this means that there is total ignorance about A .
- if $\Pi(A) > \Pi(B)$: this meaning that A is more plausible than B .

In the following, we give some properties of Π when the possibility distribution π is normalized.

- $\max(\Pi(A), \Pi(\bar{A})) = 1$: Meaning that A and \bar{A} cannot be both somewhat impossible (consequence of the normalization axiom)
- $\Pi(A \cap B) \leq \min(\Pi(A), \Pi(B))$
- $\Pi(A \cup B) = \max(\Pi(A), \Pi(B))$ (Maximitivity axiom)

Necessity Measure The necessity measure, denoted by N , of an event $A \subseteq \Omega$ is the dual of the possibility measure and it is defined follows:

$$N(A) = 1 - \Pi(\bar{A}) = \min_{\omega \notin A} (1 - \pi(\omega))$$

Intuitively, $N(A)$ defines the certainty degree associated to an event A . Namely, it evaluates to what extent A is certainly implied from the available knowledge encoded by π . It is important to note that in a possibility theory setting, in order for an event A to have a certainty degree greater than zero, it must be totally possible. In other words, A must be completely possible before being somewhat certain. This fact ensures that $N(A) \leq \Pi(A)$. We have:

- if $N(A) = 1$ and $N(\bar{A}) = 0$: this means that A is certain,
- if $N(A) \in]0, 1[$ and $N(\bar{A}) = 0$: this means that A is somewhat certain,
- if $N(A) = 0$ and $N(\bar{A}) = 0$: this means that there is a total ignorance about A ,

The following gives some properties of N when the possibility distribution π is normalized,

- $\min(N(A), N(\bar{A})) = 0$: Meaning that A and \bar{A} can not be both somewhat certain.
- $N(A \cap B) = \min(N(A), N(B))$
- $N(A \cup B) \geq \max(N(A), N(B))$

Example 2.8. Let us consider the following possibility distribution defined over to binary variables X and Y :

ω_i	X	Y	$\pi(XY)$
ω_1	x_1	y_1	1
ω_2	x_2	y_1	.8
ω_3	x_1	y_2	.5
ω_4	x_2	y_2	.1

One can check that:

- i) $\Pi(x_1)=1$ and $N(x_1)=.2$
- ii) $\Pi(y_1)=1$ and $N(y_1)=.5$ ■

Quantitative and qualitative settings

Contrary to many uncertainty frameworks (like probability theory, belief functions, etc.), possibilities could be expressed either using numeric values or using a ranking relation. These two kinds of interpretations correspond respectively to the quantitative setting (*i.e.* numerical interpretation of possibilities) and the qualitative setting (*i.e.* an ordinal interpretation of the possibility scale) [Dubois and Prade, 1998].

Quantitative Setting. The quantitative setting of possibility theory refers to the case where possibility degrees are real numbers in the unit interval $[0, 1]$. In such setting, possibility degrees have precise signification and must be a priori justified. Indeed, one can check that there are links between possibilities and probabilities. In fact, a degree of possibility can be considered as the upper probability bound [Dubois and Prade, 1992], and a possibility distribution can be viewed as a likelihood function [Dubois et al., 1997] where a possibility measure is also considered as a special case of plausibility function of Dempster-Shafer theory of evidence, etc.

However, in some situations, it remains difficult to assign exact numerical values for possible states of the world. It seems to be more flexible, in this case, to consider that a state ω of the world is more plausible than another one ω' instead of attaching to each state a numerical value. Hence, the idea to use a ranking relation over possible states of the universe of discourse Ω .

Qualitative possibility theory. The possibilistic qualitative setting refers to the case where the possibility distribution is a mapping from a universe of discourse Ω to a totally pre-ordered scale that ranks possible states of Ω . The idea of ranking the different states of the universe of discourse, was first introduced in [Spohn, 1988] through the so-called Spohn's ordinal conditional functions (OCF), well-known as kappa functions which map states of Ω into ordinals belonging to $[0, +\infty]$.

The idea behind qualitative settings is that the universe of discourse Ω is equipped with a total pre-order, denoted by \geq_π , which corresponds to a plausibility relation on Ω allowing to affirm that a state ω is more plausible than another one ω' . Given two possible states ω and ω' , when:

- $\omega =_\pi \omega'$, we say that ω is as plausible as ω' ,
- $\omega <_\pi \omega'$, we say that ω is less plausible than ω' ,
- $\omega >_\pi \omega'$, we say that ω is more plausible than ω' .

The pre-order \geq_π leads to induce a well-ordered partition of Ω , namely $\Omega = \{S_1, \dots, S_n\}$. In the qualitative setting, the ordinal scale O is of the form: $O = \{1, \alpha_1, \alpha_2, \dots, \alpha_n, 0\}$ where $1 > \alpha_1 > \alpha_2 > \dots > \alpha_n > 0$. A possibility distribution that maps a universe of discourse Ω to a totally ordered scale O , is called qualitative possibility distribution. It is important to note that the possibility scale can be numerical, namely of the form $O = \{0, 0.1, 0.3, \dots, 1\}$ where only the order relation between the values is significant, and not the real numerical values.

Possibilistic conditioning

Given a possibility distribution π , conditioning comes down to revise the available knowledge encoded in π , when a new piece of information (*i.e.* an evidence) is available [Dubois and Prade, 1988a; Dubois and Prade, 1990]. Conditioning the original possibility distribution π by an event B takes as input π and B and transforms π to a new possibility distribution denoted by $\pi' = \pi(.|B)$. Depending on the framework that we use (qualitative or quantitative), there are two main definitions of conditioning:

Quantitative setting In a quantitative setting, the widely used method of conditioning is called product-based conditioning. It uses the Dempster's rule of conditioning of belief functions, specialized to possibility measures which states that the conditional measure $\pi(.|B)$ by an event B is such that (we assume that $\Pi(B) > 0$):

$$\Pi(A|B) * \Pi(B) = \Pi(A \cap B)$$

Therefore, the impact of the event B on the available knowledge associated with an event A is given as follows (we assume that $\Pi(B) > 0$):

$$\Pi(A|_p B) = \frac{\Pi(A \cap B)}{\Pi(B)}$$

Given a possibility distribution π , the presence of the new evidence B alters π , by first declaring all states outside B as impossible, and then, proportionally changing all the states with respect to B . More formally:

$$\pi(\omega|_p B) = \begin{cases} \frac{\pi(\omega)}{\Pi(B)} & \text{if } \omega \in B \\ 0 & \text{otherwise} \end{cases}$$

It is important to note that there exist other ways for conditioning in the quantitative setting. For more details, see (*e.g.* [Fonck, 1997; Baets *et al.*, 1999; Bouchon-Meunier *et al.*, 2002]).

Qualitative setting Within a qualitative setting, the so-called min-based conditioning is the widely used method for conditioning. This latter is based on the qualitative counterpart of the Bayesian rule [Hisdal, 1978; Dubois and Prade, 1988b] which states that:

$$\Pi(A \cap B) = \min(\Pi(A|B), \Pi(B))$$

The min-based conditioning respects the minimum specificity principle which consists in assigning to the best element of the event B , the highest possibility degree (namely, 1). More formally, the min-based conditioning is performed as follows:

$$\Pi(A|_m B) = \begin{cases} 1 & \text{if } \Pi(A \cap B) = \Pi(B) \\ \Pi(A \cap B) & \text{if } \Pi(A \cap B) < \Pi(B) \end{cases}$$

Therefore, the min-based conditioning defined on all the states of a possibility distribution is given as follows:

$$\pi(\omega|_m B) = \begin{cases} 1 & \text{if } \pi(\omega) = \Pi(B) \text{ and } \omega \in B \\ \pi(\omega) & \text{if } \pi(\omega) < \Pi(B) \text{ and } \omega \in B \\ 0 & \text{otherwise} \end{cases}$$

Example 2.9. Let π be a possibility distribution over two binary variables X and Y and let y_2 be a new certain piece of information (observed information). One can compute the new possibility distribution $\pi' = \pi(\cdot|_m y_2)$, using the qualitative or quantitative conditioning as follows:

X	Y	$\pi(XY)$
x_1	y_1	1
x_2	y_1	.8
x_1	y_2	.5
x_2	y_2	.1

(a) Initial possibility distribution

X	Y	$\pi(XY _p x_2)$
x_1	y_1	0
x_2	y_1	0
x_1	y_2	1
x_2	y_2	.2

(b) Quantitative conditioning

X	Y	$\pi(XY _m b_2)$
x_1	y_1	0
x_2	y_1	0
x_1	y_2	1
x_2	y_2	.1

(c) Qualitative conditioning

Possibilistic logic

Using a possibility theory framework, the knowledge of an agent can be compactly encoded using different ways. One can either use logic-based formalisms (for instance, the ones presented in Section 1.3) which lead to obtain a possibilistic belief (or knowledge) base or a graphical-based formalism which leads to define a possibilistic graph or network (*e.g.* [Dubois and Prade, 1991b]). In this thesis, we use logic-based formalisms to encode possibilistic knowledge. This section recalls standard possibilistic logic [Dubois *et al.*, 1994], an extension of propositional logic within a possibility theory setting. Next chapter (Chapter 3) is dedicated to the extension of *DL-Lite* within a possibility theory framework.

Syntax. Let $\mathcal{B}^* = \{\phi_i : i = 1, \dots, n\}$ be a propositional knowledge base composed of a finite set (more precisely, a conjunction) of propositional formulas. A possibilistic knowledge base or belief base¹ \mathcal{B} , consists of a finite set of possibilistic formulas (ϕ_i, α_i) of the form:

$$\mathcal{B} = \{(\phi_i, \alpha_i) : i = 1, \dots, n\}$$

where ϕ_i is a propositional formula and α_i is its certainty degree, meaning that $N(\phi_i) > \alpha_i$. Note that formulas with α_i 's equal to '0' are not explicitly represented in the knowledge base. Moreover, when all α_i 's are equal to 1, \mathcal{B} coincides with a standard propositional knowledge base \mathcal{B}^* .

Semantics. The semantics of a possibilistic knowledge base \mathcal{B} is given by a possibility distribution, denoted by $\pi_{\mathcal{B}}$, defined over the set of propositional interpretations, namely $\Omega = \{I_1, \dots, I_n\}$. The possibility distribution $\pi_{\mathcal{B}}$ attaches to each interpretation $I \in \Omega$ a possibility degree reflecting to what extent this latter satisfies² formulas of the knowledge base. The possibility degree of an interpretation $\pi(I)$ depends on the maximum weight of formulas falsified by the interpretation I .

¹Note that throughout this Chapter and this thesis in general, and when there is no ambiguity, we do not make difference between belief base and knowledge base.

²The satisfaction relation is recalled in Section 1.3.1

$$\forall I \in \Omega, \pi_{\mathcal{B}}(I) = \begin{cases} 1 & \text{if } \forall (\phi_i, \alpha_i) \in \mathcal{B}, I \models \phi_i, \\ 1 - \max\{\alpha_i : (\phi_i, \alpha_i) \in \mathcal{B}, I \not\models \phi_i\} & \text{otherwise} \end{cases}$$

It is important to note that a possibilistic knowledge base is considered as a compact representation of a possibility distributions. Namely, from each possibilistic knowledge base \mathcal{B} , one can generate its possibility distribution $\pi_{\mathcal{B}}$.

By referring to classical logic, when the formulas of the knowledge base are completely certain (namely, having weights equal to 1 in a possibilistic setting), then the knowledge base will contain only models (i.e. $\pi_{\mathcal{B}}(I) = 1$) or countermodels ($\pi_{\mathcal{B}}(I) = 0$). As a consequence, the consistency of the knowledge base is binary, namely \mathcal{B} is consistent or inconsistent. This is not the case in possibilistic knowledge bases. As the possibility distribution allows to attribute to the countermodels a degree of compatibility with the available knowledge. In this case, the consistency of an interpretation with respect to the available knowledge is not binary. Therefore, one can associate to a possibilistic knowledge base a degree of inconsistency between 0 and 1.

Example 2.10. Let $\mathcal{B} = \{(a, .6), (a \wedge b, .1), (c \vee b, .4)\}$ be a possibilistic knowledge base. The joint possibility distribution $\pi_{\mathcal{B}}$ of \mathcal{B} is as follows:

I	$\pi_{\mathcal{B}}(I)$
abc	1
$ab\bar{c}$	1
$a\bar{b}c$.9
$a\bar{b}\bar{c}$.6
$\bar{a}bc$.4
$\bar{a}b\bar{c}$.4
$\bar{a}\bar{b}c$.4
$\bar{a}\bar{b}\bar{c}$.4

Let $\mathcal{B} = \{(\phi_i, \alpha_i) : i = 1, \dots, n\}$ be a possibilistic knowledge base, the inconsistency degree of \mathcal{B} , denoted by $Inc(\mathcal{B})$, is defined semantically and syntactically as follows:

- Semantically using the induced possibility distribution: $Inc(\mathcal{B}) = 1 - \max_{I \in \Omega}(\pi_{\mathcal{B}}(I))$ where $\pi_{\mathcal{B}}$ is its possibility distribution.
- Syntactically using the concepts of α -cut: $Inc(\mathcal{B}) = \max\{\alpha_i : \mathcal{B}_{\geq \alpha} \text{ is inconsistent}\}$ where $\mathcal{B}_{\geq \alpha}$, is called the α -cut of \mathcal{B} and it is the subbase of \mathcal{B} composed by formulas having weights greater than or equal to α . If $\mathcal{B}_{\geq 0}$ is consistent then $Inc(\mathcal{B}) = 0$.

It was shown in [Dubois *et al.*, 1994] that the computational complexity of computing the inconsistency degree of a possibilistic knowledge base is in $(\log_2(n) * \text{SAT})$ where n is the number of different weights in the knowledge base and SAT is the complexity of the propositional satisfiability problem. Namely, computing inconsistency degrees needs $\log_2(n)$ calls to a SAT solver. Contrary to classical logic, using the notion of inconsistency degree, possibilistic logic allows reasoning from an inconsistent knowledge base.

Let $\mathcal{B} = \{(\phi_i, \alpha_i) : i = 1, \dots, n\}$ be a possibilistic knowledge base, possibilistic entailments are defined semantically as follows:

- A formula is a logical consequence of a possibilistic knowledge base, denoted by $\pi_{\mathcal{B}} \models \phi_i$ if and only if $N(\phi_i) > 0$ where $N(\phi_i)$ is the necessity degree of ϕ_i computed from $\pi_{\mathcal{B}}$.

- A formula is a logical consequence of a possibilistic knowledge base with a certainty degree α_i , denoted by $\pi_{\mathcal{B}} \models (\phi_i, \alpha_i)$ if and only if $N(\phi_i) \geq \alpha_i > 0$ where $N(\phi_i)$ is the necessity degree of ϕ_i computed from $\pi_{\mathcal{B}}$.

The above reasoning tasks can be done syntactically as follows:

- A formula ϕ_i is said to be a plausible conclusion of \mathcal{B} , denoted by $\mathcal{B} \models_P \phi_i$ if and only if $\mathcal{B}_{>inc(\mathcal{B})} \models_P \phi_i$.
- A formula ϕ_i is said to be a possibilistic conclusion of \mathcal{B} , denoted by $\mathcal{B} \models_{\pi} (\phi_i, \alpha_i)$ if and only if $\mathcal{B}_{\geq\alpha_i}$ is consistent, $\mathcal{B}_{\geq\alpha_i} \models \phi_i$ and $\forall \beta > \alpha_i, \mathcal{B}_{\geq\beta} \not\models \phi_i$.

Note that the above inferences can be reduced to computing the inconsistency degree of the possibilistic knowledge base. More formally:

- $\mathcal{B} \models_P \phi_i$ if and only if $Inc(\mathcal{B} \cup \{(\neg\phi_i, 1)\}) > Inc(\mathcal{B})$.
- $\mathcal{B} \models_{\pi} (\phi_i, \alpha_i)$ if and only if $Inc(\mathcal{B} \cup \{(\neg\phi_i, 1)\}) > \alpha_i$.

Another compact representation of possibility distributions is the one of possibilistic networks.

Possibilistic networks

Possibilistic networks (e.g. [Fonck, 1994; Gebhardt and Kruse, 1996; Benferhat *et al.*, 2001; Benferhat *et al.*, 2002a]) are frameworks used for representing and reasoning with uncertain information. Comparing with possibilistic logic, possibilistic networks explicit relationships between different variables of the domain while possibilistic logic only ranks formulas according to their certainty level.

A possibilistic network $\mathcal{G} = \langle G, \Theta \rangle$ is specified by:

- i) A *graphical component* G consisting of a directed acyclic graph (DAG) where vertices represent the variables and edges represent direct *dependence* relationships between variables. Each variable A_i is associated with a domain D_i containing the values a_i taken by the variable A_i .
- ii) A *numerical component* Θ allowing to assess the uncertainty relative to each variable using local possibility tables. The possibilistic component consists in a set of local possibility tables $\Theta_i = \{\theta_{a_i|u_i}\}$ where $a_i \in D_i$ and u_i is an instance of U_i denoting the parent variables of A_i in the network \mathcal{G} .

Note that all the local possibility distributions Θ_i must be normalized, namely

$$\forall i = 1..n, \forall u_i \in D_{U_i}, \max_{a_i \in D_i} (\theta_{a_i|u_i}) = 1.$$

Example 2.11. Figure 2.1 gives an example of a possibilistic network over four boolean variables A , B , C and D .

The structure of \mathcal{G} encodes a set of independence relationships $I_r = \{I_r(A_i, U_i, Y)\}$ where each variable A_i in the context of its parents U_i is independent of its non descendants Y . For example, in the network of Figure 2.1, variable C is independent of B and D in the context of A .

In possibilistic networks, the joint possibility distribution is factorized using the possibilistic chain rule:

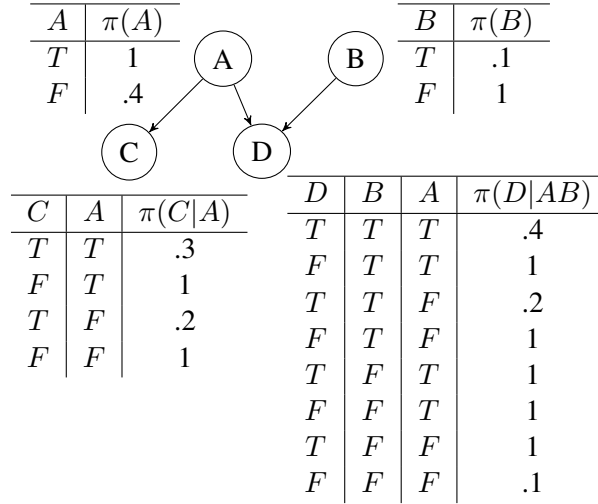


Figure 2.1: Example of a possibilistic network

$$\pi(a_1, a_2, \dots, a_n) = \bigotimes_{i=1}^n (\pi(a_i | u_i)).$$

where \bigotimes is either the product-based or min-based operator.

Lastly, there exists a translation from possibilistic networks to possibilistic knowledge bases [Benferhat *et al.*, 2002a]. For reasoning in possibilistic networks, see [Fonck, 1994; Benamor, 2002; Ayachi *et al.*, 2013] for instance.

2.2.3 Uncertainty management in description logics

Uncertainty reasoning for the World Wide Web has received in recent years a lot of attention³. Several approaches are proposed and they are based on the extension of DLs within uncertainty theories. In the following, we recall the main proposed approaches.

Probabilistic description logics

Probabilistic Description Logics (e.g. [Giugno and Lukasiewicz, 2002; Lukasiewicz, 2002; Lukasiewicz and Straccia, 2008; Lukasiewicz *et al.*, 2012a]) is an extension of standard DLs with probabilistic terminological axioms and probabilistic assertional facts in order to manage uncertainty. Probabilistic knowledge in probabilistic description logics is modeled using the notion of probabilistic conditional constraints [Lukasiewicz, 1999]. A conditional constraint of the form $(A|B) [l, u]$ is expressed by attaching a belief interval reflecting the lower bound l and the upper bound u of the probability of concluding A given an evidence B where A and B are two concepts. Intuitively, a conditional constraint represents a concept inclusion relation between two concepts A and B of the form $B \sqsubseteq A$, with a probability degree between l and u . This permits to model the fact that "generally, if an individual belongs to B , then it belongs to A with a probability lying between l and u ". Similarly, for probabilistic assertions, a conditional constraint $(A(a)|\top) [l, u]$ is used to express the fact that a is an instance of the concept A with a probability degree that lies between l and u ".

³<http://www.w3.org/2005/Incubator/urw3/XGR-urw3-20080331/>

In a probabilistic knowledge base, generic knowledge is encoded in a probabilistic TBox, denoted by PTBox, and assertions are stored in a probabilistic ABox, denoted by PABox. A PTBox $PT = (T, P)$ contains a set of standard DL axioms, *i.e.* a standard TBox \mathcal{T} (expressed using the used DL language), and a set of conditional constraints P defined over the concepts of the domain. A PABox P_o is a finite set of conditional constraints defined on probabilistic individuals $o \in I_p$. A probabilistic knowledge base is a triple $PK = (T, P, (P_o)_{o \in I_p})$ relative to I_p . It is important to note that the set of individuals is partitioned into a set of standard individuals I_s and a set of probabilistic individuals I_p .

From a reasoning point of view, consistency checking in a probabilistic knowledge and entailment are based respectively on the notions of consistency and lexicographic entailment proposed in probabilistic default reasoning [Lukasiewicz, 2002].

Fuzzy description logics

Fuzzy Description Logics (*e.g.* [Straccia, 1998; Straccia, 2001; Bobillo and Straccia, 2007; Bobillo and Straccia, 2012; Lukasiewicz and Straccia, 2009]) are extensions of DLs within fuzzy sets theory. The aim of such extension is to model fuzziness attached to the elements of the domain. From a syntactic point of view, a fuzzy description logic uses first fuzzy concepts and weighted ABox assertions of the form $(A(a), n)$ where $A(a)$ is an ABox assertion and $n \in [0, 1]$ is its membership degree to the fuzzy concept A .

The semantics of fuzzy DL relies on the fuzzy set semantics [Zadeh, 1965]. Recall that, a fuzzy set S is defined with respect to a set S' by a membership function $\mu_S : S' \rightarrow [0, 1]$, that assigns to each element in S' a membership degree in $[0, 1]$. Within a description logic setting, a fuzzy interpretation is a pair $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ where $\Delta^{\mathcal{I}}$ is the domain of the interpretation, defined as in the standard description logic semantics, and $\cdot^{\mathcal{I}}$ is an interpretation function that maps i) each individual as in the standard case, ii) each concept into a membership function $\Delta^{\mathcal{I}} \rightarrow [0, 1]$ iii) and each role into membership function $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \rightarrow [0, 1]$.

Given an interpretation \mathcal{I} , a concept A is interpreted as a membership function and $A^{\mathcal{I}}(a)$ with a an individual (*i.e.* $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$) is interpreted as the truth degree of the object a being an element of A under the interpretation \mathcal{I} . The fuzzy interpretation is extended to DL constructs following a fuzzy semantics aggregation modes.

Lastly, it is important to note that this representation can be handled efficiently using possibility theory. For more details about fuzzy description logics, see [Lukasiewicz and Straccia, 2008].

Possibilistic description logics

Possibilistic Description Logics are frameworks introduced to deal with uncertainty and to ensure reasoning under inconsistent knowledge bases. The use of possibility theory to extend DLs has been proposed in [Hollunder, 1995] and discussed in [Dubois *et al.*, 2006]. In [Hollunder, 1995] a possibilistic DL knowledge base was defined syntactically by attaching to every terminological axiom or assertion a necessity degree. However there is no formal foundation of the semantic counterpart of this extension. In addition, only some standard inference services have been defined. From an algorithmic point of view, Hollunder's method [Hollunder, 1995] is based on an instantiation of possibilistic entailment with a classical inference algorithm for DLs.

In [Qi *et al.*, 2007b; Qi *et al.*, 2007a], the authors go one step further in the definition of possibilistic DL. A possibilistic DL knowledge base has been defined syntactically by equipping every axiom with a confidence degree to encode its certainty. This confidence degree is simply the necessity value of an axiom and it reflects to what extent this latter can be considered as certain (priority, importance, etc) with respect to the available knowledge. These degrees are then used to determine the inconsistency degree

of a knowledge base and to ensure inference services. From a computational point of view, an algorithm to compute inconsistency degrees and possibilistic inference services has been provided. In general, it has been shown [Qi *et al.*, 2007b; Qi *et al.*, 2007a] that checking the consistency degree and several inference services can be done with classical DLs reasoning services through consistent sub-sets of the Possibilistic DL knowledge base. Clearly, computing inconsistency degrees comes down to perform a dichotomie search among the certainty scale while calling a standard DL reasoner is closely related to the method proposed in [Dubois *et al.*, 1994] for computing inconsistency degrees of a possibilistic propositional knowledge base.

An implementation of a reasoner called “DL-Poss”, has been provided in [Qi *et al.*, 2010a] (see also [Qi *et al.*, 2011] for a discussion on Possibilistic DLs). Finally, another method has been introduced in [Couchariere *et al.*, 2008a; Qi *et al.*, 2008b; Zhu *et al.*, 2013] for checking the inconsistency of a possibilistic DL base as a direct extension of the tableau algorithm [Baader *et al.*, 2010].

In Chapter 3, we more discuss these works and we follow another direction to extend *DL-Lite* within a possibility theory setting. The main feature of this extension is that it is done by slightly modifying the reasoning method proposed in standard *DL-Lite* by propagating the uncertainty degrees associated with formulas in the knowledge base. Compared to the existing works, this extension allows to equip *DL-Lite* with many other reasoning capabilities like merging (Chapter 4), inconsistency handling (Chapter 7) when the assertional base is prioritized.

2.3 Belief change

As pointed out in Section 1.2, there are several lines of research that aim to equip ontologies with additional reasoning abilities (in addition to classical ones). In this thesis, we consider the problem of ontology evolution, ontology merging and ontology repairing, with a focus on the context of Ontology-Based Data Access. These problems are respectively closely related to belief revision, belief merging and inconsistency handling in a propositional logic setting. This section gives a brief overview on these topics.

2.3.1 Belief revision

Originally, description logics have been introduced to represent the static aspects of a domain of interest [Baader *et al.*, 2010]. However, for some applications, knowledge may not be static and evolves from a situation to another in order to cope with changes that occur over time. Such dynamic aspects have been recognized as important problems (*e.g.* [Qi *et al.*, 2006c; Calvanese *et al.*, 2010; Wang *et al.*, 2010; Kharlamov *et al.*, 2013]) and often concern the situation where new information should be taken into account requiring to modify the old one while ensuring the consistency of the result. Such problem is well-known as belief revision.

Belief revision problem in a propositional logic setting

Belief revision has been defined as knowledge change and was characterized for instance by the well-known AGM postulates [Alchourrón *et al.*, 1985]. These postulates are based on the following three main ideas:

1. The principle of *priority*⁴ which states that the priority between beliefs is given to the new pieces

⁴Note that the notion of priority here refers to the fact that the new information is more reliable than the old one while in a prioritized setting two pieces of information may have different priority levels reflecting their plausibility with respect to available knowledge. Indeed, the priority of the new information refers to the dynamic aspect while in a static context, one can

of information,

2. The principle of *consistency* which states that the result of the revision operation must be a consistent set of beliefs, and
3. The principle of *minimal* change which states that as less as possible initial beliefs should be changed in the revision operation.

Belief revision has been largely considered in the literature when knowledge bases are encoded using a propositional language. In a propositional logic setting, the AGM postulates were equivalently presented in [Katsuno and Mendelzon, 1991]. Given a knowledge base and a new information expressed respectively using two propositional formulas ϕ and φ and a revision operator, denoted by \circ , then $\phi \circ \varphi$ should satisfy the following postulates, well-known as KM postulates [Katsuno and Mendelzon, 1991]:

- (R1)** $\phi \circ \varphi$ implies φ
- (R2)** if $\phi \wedge \varphi$ is satisfiable, then $\phi \circ \varphi \equiv \varphi \wedge \phi$
- (R3)** if φ is satisfiable, then $\phi \circ \varphi$ is satisfiable
- (R4)** if $\phi_1 \equiv \phi_2$ and $\varphi_1 \equiv \varphi_2$, then $\phi_1 \circ \varphi_1 \equiv \phi_2 \circ \varphi_2$.
- (R5)** $(\phi \circ \varphi) \wedge \psi$ implies $\phi \circ (\varphi \wedge \psi)$
- (R6)** if $(\phi \circ \varphi) \wedge \psi$ is satisfiable, then $\phi \circ (\varphi \wedge \psi)$ implies $(\phi \circ \varphi) \wedge \psi$ (where ψ is a propositional formula).

Intuitively, postulate (R1) states that the models of the revised formula with the new information φ are also models of φ , or simply the new information should be entailed from the result of revision. (R2) says that if the new information is consistent with the initial one, then the result of revision of ϕ with φ is made by their intersection. (R3) indicates that the result of revision is satisfiable if the new information is satisfiable. (R4) expresses the syntax independence of the revision operator. (R5) and (R6) together ensure closeness, *i.e.* the minimal change principle.

It is important to note that the KM postulates are equivalent to AGM postulates [Katsuno and Mendelzon, 1991] in a propositional logic setting. Moreover, a representation theorem, based on the notion of *syncretic assignment* was proposed. Recall that a *faithful assignment* is a function that defines a total pre-order \leq_ϕ over the set of interpretations that represents the formula ϕ . Let $mod(\phi)$ be the set of models of ϕ , the pre-order \leq_ϕ associated to ϕ is a *faithful assignment* if and only if:

- if $I \in mod(\phi)$ and $I' \in mod(\phi)$, then $I =_\phi I'$,
- if $I \in mod(\phi)$ and $I' \notin mod(\phi)$, then $I <_\phi I'$,
- if $\phi \equiv \varphi$, then $\leq_\phi = \leq_\varphi$.

A revision operator satisfies postulates (R1)-(R6) if there exist a *faithful assignment* that associates to ϕ a total pre-order \leq_ϕ such that:

$$mod(\phi \circ \varphi) = \min(mod(\varphi), \leq_\phi)$$

Note that there exists other representation theorems used to define revision operators in addition of the representation theorem based on *faithful assignment*. Based on these works, several belief revision approaches are proposed. One can classify these approaches in two main classes:

have different pieces of information with different priority/uncertainty levels.

Semantics approaches : well-known as model-based approaches, they are based on the interpretations of the formulas such as Grove’s approach based on spheres [Grove, 1988], revision operators based on distance between interpretations (e.g. [Borgida, 1985; Dalal, 1988; Satoh, 1988]), etc.

Syntactic approaches : well-known as formula-based approaches, they are based on formulas such as semi-revision [Hansson, 1997], selective revision [Fermé and Hansson, 1999], removed set revision [Papini, 1992; Würbel *et al.*, 2000], etc.

Note that AGM postulates were defined for revising belief sets, i.e deductively closed sets of formulas, possibly infinite. Besides, an axiomatic characterization for revising belief bases, namely finite set of formulas was given in [Fuhrmann, 1997; Hansson, 1998].

There is an extended approach of revision, called iterative revision [Lehmann, 1995; Darwiche and Pearl, 1997] that permits revision of a knowledge base with a sequence of pieces of information, namely $(\varphi_1, \dots, \varphi_n)$ where as usual each piece of information φ_i has the priority over the set of initial beliefs and φ_j is more preferred than φ_i for $1 < i < j < n$. A logical characterization of iterative revision was given by Darwiche and Pearl in [Darwiche and Pearl, 1997]. There are several operators for iterative revision such as: revision proposed by Boutilier [Boutilier, 1993], possibilistic revision [Benferhat *et al.*, 2002c], revision approach based on polynomials [Benferhat *et al.*, 2002b], etc.

Revision within description logic settings

Recently, several works have been proposed for revising DLs knowledge bases. In [Flouris *et al.*, 2004; Flouris *et al.*, 2005] an adaptation of the AGM postulates was discussed in order to generalize it to DLs. The authors in [Qi *et al.*, 2006c] focused on revising a finite representation of belief sets. They used a semantic reformulation of AGM postulates, done by Katsuno and Mendelzon [Katsuno and Mendelzon, 1991], to extend it to DLs knowledge bases. Recently, several works were proposed to define revision operators for description logics. In [Halaschek-wiener *et al.*, 2006; Ribeiro and Wassermann, 2007; Qi *et al.*, 2008a] an extension of kernel-based revision and semi-revision operators to DLs frameworks has been proposed. It is closely related to the one proposed by [Hansson, 1997] in a propositional logic setting. In [Qi and Yang, 2008; Qi and Du, 2009; Wang *et al.*, 2010], model-based approaches for revising DLs have also been proposed.

However, as pointed out in [Calvanese *et al.*, 2010] model-based approaches of revision are not expressible in *DL-Lite* in the sense that the result of revision is not expressible in the language in which the initial knowledge base is expressed. Moreover, most of the approaches are restricted to the revision of the TBox [Qi and Du, 2009; Zhuang *et al.*, 2014] or the ABox (e.g. [Liu *et al.*, 2006; Gao *et al.*, 2012]) but not both. Regarding *DL-Lite* knowledge bases, few works have been proposed for the revision problem. In [Calvanese *et al.*, 2010; Kharlamov and Zheleznyakov, 2011], a formula-based approach for revising *DL-Lite* knowledge bases has been presented. Two algorithms have been proposed: one for revising the TBox, and the other for revising the ABox. Another operator for ABox revision in *DL-Lite* based on graph structure has been introduced in [Gao *et al.*, 2012]. In this work, the new information is restricted to ABox assertions. In [Zhuang *et al.*, 2014] a revision approach based on propositional logic reduction was proposed to revise a TBox.

In Chapter 6, we investigate a formula-based approach for revising *DL-Lite* with either a TBox axiom or an ABox assertion. We consider the case of a *DL-Lite* knowledge base where the ABox is prioritized. In such setting, a new TBox axiom can only expand (enrich) generic knowledge while revision process comes down to throw out some assertional facts in order to restore consistency.

2.3.2 Belief merging

Knowledge bases merging or belief merging (e.g. [Bloch *et al.*, 2001; Everaere *et al.*, 2010; Everaere *et al.*, 2012]) is an important problem addressed in many application areas such as multi-agent systems, distributed databases, etc. It focuses on aggregating pieces of information issued from distinct, and may be/potentially conflicting or inconsistent, sources of information. It leads to a global point of view of the considered problems by taking advantage of pieces of information provided by each source. Merging pieces of information requires to use some fusion operators that permit to combine them while respecting different constraints between sources. In the literature, several fusion operators have been proposed which depend on the nature and the representation of knowledge such as merging propositional knowledge bases (e.g. [Konieczny and Pino Pérez, 2002]), prioritized knowledge bases (e.g. [Delgrande *et al.*, 2006]) or weighted logical knowledge bases (e.g. [Benferhat *et al.*, 1993a]).

The next two sections introduce merging in the framework of propositional logic and possibilistic logic.

Merging within a propositional logic setting

In propositional logic frameworks, a belief base denoted by K_i , is constituted by a finite set of propositional formulas. Let us use $\mathcal{E} = \{K_i, \dots, K_n\}$ to denote a multi-set, called belief profile, to represent the belief bases to be merged. Assume that each belief base is consistent. Merging multiple consistent belief bases may lead to conflicts between bases. Hence, the aggregation process requires to perform suitable merging operators.

Let us use Δ to denote a merging operator and $\Delta(\mathcal{E})$ to denote the result of this merging which is equal to a propositional formula (*i.e.* a knowledge base). In some cases, a merging operator is submitted to integrity constraints, denoted by ρ , generally expressed by a set of propositional formulas. Hence, a merging operation with integrity constraints, denoted by $\Delta_\rho(\mathcal{E})$, must retain the integrity constraints explicitly represented in the merging result and not simply implicitly consistent with the merging result. A logical characterization of integrity constraints merging operators has been proposed in [Konieczny and Pino Pérez, 2002] through a set of rational postulates extended for the one proposed for belief revision. We recall rational postulates proposed to characterize the ideal behavior of a fusion operator.

$$(IC0) \quad \Delta_\rho(\mathcal{E}) \models \rho$$

$$(IC1) \quad \text{If } \rho \text{ is consistent, then } \Delta_\rho(\mathcal{E}) \text{ is consistent.}$$

$$(IC2) \quad \text{If } \bigwedge_{K \in \mathcal{E}} K \text{ is consistent with } \rho, \text{ then } \Delta_\rho(\mathcal{E}) = \bigwedge_{K \in \mathcal{E}} K \wedge \rho$$

$$(IC3) \quad \text{If } \mathcal{E}_1 \equiv \mathcal{E}_2 \text{ and } \rho_1 \equiv \rho_2, \text{ then } \Delta_{\rho_1}(\mathcal{E}_1) \equiv \Delta_{\rho_2}(\mathcal{E}_2)$$

$$(IC4) \quad \text{If } K \models \rho \text{ and } K' \models \rho, \text{ then } \Delta_\rho(\{K, K'\}) \wedge K \text{ is consistent if and only if } \Delta_\rho(\{K, K'\}) \wedge K' \text{ is consistent.}$$

$$(IC5) \quad \Delta_\rho(\mathcal{E}_1) \wedge \Delta_\rho(\mathcal{E}_2) \models \Delta_\rho(\mathcal{E}_1 \sqcup \mathcal{E}_2)$$

$$(IC6) \quad \text{If } \Delta_\rho(\mathcal{E}_1) \wedge \Delta_\rho(\mathcal{E}_2) \text{ is consistent, then } \Delta_\rho(\mathcal{E}_1 \sqcup \mathcal{E}_2) \models \Delta_\rho(\mathcal{E}_1) \wedge \Delta_\rho(\mathcal{E}_2)$$

$$(IC7) \quad \Delta_{\rho_1}(\mathcal{E}) \wedge \rho_2 \models \Delta_{\rho_1 \wedge \rho_2}(\mathcal{E})$$

$$(IC8) \quad \text{if } \Delta_{\rho_1}(\mathcal{E}) \wedge \rho_2 \text{ is consistent, then } \Delta_{\rho_1 \wedge \rho_2}(\mathcal{E}) \models \Delta_{\rho_1}(\mathcal{E}) \wedge \rho_2$$

Intuitively, these postulates seek for a set of propositional formulas that represents the fusion result in the most faithful way to merge belief bases while respecting the integrity constraints. Indeed, the first

postulate **(IC0)** states that the fusion result must satisfy the integrity constraints. **(IC1)** ensures that if the set of integrity constraints is consistent then the fusion result must also be consistent. **(IC2)** means that whenever is possible, the fusion result is simply the conjunction of the merged beliefs bases and the integrity constraint. **(IC3)** simply refers to the syntax-irrelevancy principle and it states that if two belief profiles are equivalent and that two integrity constraint sets are logically equivalent then the resulting belief base of each belief profile are logically equivalent. **(IC4)** assures that when merging a pair of belief bases then the merging operator must not give any preference to one of them. **(IC5)** and **(IC6)** together affirm that if possible one can find two subgroups of sources which are in agreement on at least one interpretation, then the fusion result is exactly the interpretations on which the two groups are in agreement. Finally, **(IC7)** and **(IC8)** are introduced to preserve the notion of closeness (early presented in KM postulates [Katsuno and Mendelzon, 1991] through the postulates R5 and R6).

In [Konieczny and Pino Pérez, 2002] merging operators were classified in two major subclasses, namely majority merging operators and arbitration merging operators. Formally, the family of majority merging operators is logically characterized by the following postulate:

$$\text{(Maj)} \quad \exists n \in \mathbb{N} \text{ such that } \Delta_{\rho}(\mathcal{E}_1 \sqcup \mathcal{E}_2^n) \models \Delta_{\rho}(\mathcal{E}_2)$$

This postulate affirms that if a particular set of beliefs is repeated quite enough in the whole set of belief profile then this particular set of beliefs must prevail in the fusion result. Hence, it is obvious that majority operators behavior is sensitive to redundancy.

Contrary to majority merging operators that take into account the opinion of the majority about a situation, arbitration merging operators try to better satisfy opinions as many opinions as possible among integrity constraints [Konieczny and Pino Pérez, 2002]. Namely, if a set of preferred opinions among a set of integrity constraints ρ_1 for a belief base K corresponds to the preferred opinions among the set of integrity constraints ρ_2 of another base K' and if the opinions that belong to a set of integrity constraints but not to the other are equally preferred for the whole groups $(\{K, K'\})$, then the subset of preferred opinions among the disjunction of integrity constraints will coincide with the preferred opinions of each base among their respective integrity constraints. Formally, the arbitration merging operators are logically characterized by the following postulate:

$$\text{(Arb)} \quad \left. \begin{array}{l} \Delta_{\rho_1}(K_1) \equiv \Delta_{\rho_2}(K_2) \\ \Delta_{\rho_1 \leftrightarrow \neg \rho_2}(\{K_1, K_2\}) (\rho_1 \leftrightarrow \neg \rho_2) \\ \rho_1 \not\equiv \rho_2 \\ \rho_2 \not\equiv \rho_1 \end{array} \right\} \Rightarrow \Delta_{\rho_1 \vee \rho_2}(\{K_1, K_2\}) \equiv \Delta_{\rho_1}(K_1)$$

Now, according to rational postulates **IC0-IC8**, an integrity constraint merging operator is logically defined by a representation theorem according to the notion of *syncretic assignment* [Konieczny and Pino Pérez, 2002]. The *syncretic assignment* tries to build up a pre-order on interpretations and it is defined as an extension *faithful assignment* defined for belief revision [Katsuno and Mendelzon, 1991]. Note that a *majority syncretic assignment* and *arbitration syncretic assignment* are also defined by adding some conditions to those of *syncretic assignment*.

According to the notion of *syncretic assignment*, the representation theorem for integrity constraint merging operators states that the result of merging the belief sets with the merging operator Δ_{ρ} is simply represented by the pre-order $\leq_{\mathcal{E}}$ on the interpretations that consist in computing minimal interpretations to the pre-order associated to the belief bases to be merged. Namely, an integrity constraint merging operator satisfies postulates (IC1)-(IC8) if and only if there exists a *syncretic assignment* that associates to \mathcal{E} a total pre-order $\leq_{\mathcal{E}}$ such that:

$$\text{mod}(\Delta_{\rho}(\mathcal{E})) = \min(\text{mod}(\rho), \leq_{\mathcal{E}})$$

Generally, pieces of information provided by each source are represented, on one hand, syntactically using a propositional set of formulas, and on the other hand, semantically through a set of interpretations. Thus, as for belief revision approaches, one can identify two categories of merging operators.

Semantic merging operators well-known as model-based merging operators, they are parametrized by a distance and an aggregation function. Merging process consists first in ranking interpretations using some distance measures, then combining them using an aggregation function to generally obtain a unique ordering on interpretations for all sources of information. There are several model-based operators: The *Sum* merging operator which is considered as majority merging operator and satisfies the postulate (IC0)-(IC8), the *Max* merging operator which is considered as an approximation of arbitration merging operator, the *GMax* operator [Konieczny and Pino Pérez, 2002], the DA^2 operators which are based on a distance (D) and two aggregation function (A^2), the disjunctive operators [Everaere et al., 2010], etc.

Syntactic merging operators well-known as formula-based merging operators, they consist to merge formulas of each belief base to obtain a unique consistent belief base that represents all sources of information. The fusion result which must be a consistent set of formulas depends on the syntactic representation of the merged belief bases. This family of merging operators tries to find from the union of merged belief bases, the consistent and maximal subsets of formulas. Note that the maximality criterion here is in the sense of set inclusion (it can also be defined in terms of cardinality). Formally, let us use $MaxCons(K, \rho)$ to denote the collection of the maximal consistent subsets from $K \cup \rho$ which necessarily satisfies the integrity constraints ρ . Namely, a maximal consistent subset $M \in MaxCons(K, \rho)$ satisfies the following requirements: i) $M \subseteq K \cup \rho$, ii) $\rho \in M$, and iii) if $M \subset M' \subseteq \{K \cup \rho\}$ then M' is inconsistent.

Let $MaxCons(\mathcal{E}, c) = MacCons(\bigcup_{K_i \in \mathcal{E}} K_i, \rho)$. When the maximality criterion is in the sense of cardinality criterion, we will use $MaxCons_{card}(\mathcal{E}, \rho)$ as notation. One can define the combination operators as follows: Let \mathcal{E} be a belief set and ρ be an integrity constraint:

$$\Delta_{\rho}^{C_1}(\mathcal{E}) = \bigvee MaxCons(\mathcal{E}, \rho)$$

$$\Delta_{\rho}^{C_3}(\mathcal{E}) = \bigvee \{M : M \in MaxCons(\mathcal{E}, \top) \text{ and } M \cup \{\rho\} \text{ is consistent}\}$$

$$\Delta_{\rho}^{C_4}(\mathcal{E}) = \bigvee MaxCons_{card}(\mathcal{E}, \rho)$$

$$\Delta_{\rho}^{C_5}(\mathcal{E}) = \bigvee \{M \cup \{\rho\} : M \in MaxCons(\mathcal{E}, \top) \text{ and } M \cup \{\rho\} \text{ is consistent}\} \text{ if this set is not empty, otherwise } \rho.$$

As pointed out in [Konieczny and Pino Pérez, 2002], the combination operators are rather similar to the techniques dedicated to inconsistency-tolerant reasoning from an inconsistent belief bases, than merging operators. Hence, combination operators do not exploit the repartition of the information between sources. Besides, they generally lose the original information provided by merged information sources. To overcome this inconvenient behavior of combination operators, one can define selection functions in the same way as the belief revision. Those selection functions try to select from the whole maximal consistent sets a group in order to attenuate as much possible the fusion result produced by combination operators.

Merging in a possibilistic logic setting

In a possibility theory framework, several fusion operators (e.g. [Dubois et al., 1992; Benferhat et al., 1997b; Benferhat and Kaci, 2003]) have been proposed for merging pieces of information issued from different and potentially conflicting or inconsistent sources. In general, the possibilistic fusion process first consists in combining a set of possibility distributions that encode the information in order to obtain

a unique possibility distribution that represents the global point of view of the available information, and then, it provides for each fusion operator used to combine possibility distributions its syntactic counterpart when merging possibilistic knowledge bases.

Let $\mathcal{E} = \{\mathcal{B}_1, \dots, \mathcal{B}_n\}$ be n possibilistic knowledge bases to be merged where each of them is associated to a possibility distribution, denoted by π_i . Let $V(I) = \langle \pi_1(I), \dots, \pi_n(I) \rangle$ be a vector that groups for each interpretation all possibility degree $\pi_i(I)$. When aggregating possibility distributions, two main properties are required:

- If $\forall i, \pi_i(I) = 1$ then $\pi_{\odot} = 1$ and,
- If $\forall i, \pi_i(I') \leq \pi_i(I)$ then $\pi_{\odot}(I') \leq \pi_{\odot}(I)$.

The first property is called consistency property and it states that if the different sources agree that nothing prevents I to be the real world, then the fusion result must confirm the total possibility of I . Intuitively, this property ensures the consistency of the fusion result if the union of the merged belief bases is consistent. The second property is called monotonicity property and it affirms that if all information sources are agree that an interpretation I is at least as preferred as another interpretation I' , then the fusion result must conserve this preference.

According to the properties presented above, a possibilistic merging approach first introduces a semantic merging operator to combine possibility distributions, represented by $V(I)$, to obtain a unique possibility distribution, denoted by π_{\odot} . Then it provides the syntactic counterpart for this operator used to combine possibility distributions for merging possibilistic knowledge bases and obtain a unique base, denoted by \mathcal{B}_{\odot} ⁵ such that $\pi_{\mathcal{B}_{\odot}} = \pi_{\odot}$.

The basic aggregation modes proposed for possibilistic merging are the conjunction and the disjunction modes.

Conjunctive operators When all the information sources are considered as equally and fully reliable, then one can exploit the complementarity between the difference sources. Namely, when the available information is inconsistent from one source to another. Thus all values that are considered as impossible by one source but possible by the others are rejected. A conjunctive merging operator, denoted by \oplus is defined as follows:

$$\forall a \in [0, 1], \oplus(a, 1) = \oplus(1, a) = a$$

The result of the syntactic counterpart of conjunctive merging operator is a possibilistic knowledge base obtained as follows:

$$\mathcal{B}_{\oplus} = \mathcal{B}_1 \cup \mathcal{B}_2 \cup \{(\varphi_i \vee \phi_j, 1 - \oplus(1 - a_i, 1 - b_j))\}$$

In this case, possibilistic fusion preserves all the available information when the union of merged sources is consistent.

Now, when the information sources are considered as dependent, one can see the redundancy as a way of strengthening or confirmation. Whereas when the information sources are assumed to be independent, the redundancy may be ignored. Indeed, the conjunctive aggregation modes may be adapted to these two kinds of situations. In the first case (dependent information sources) the conjunctive aggregation modes are called idempotent aggregation operators and they satisfy the following property:

⁵For the sake of simplicity and to respect notations proposed in the original papers on possibilistic merging, we used \mathcal{B}_{\odot} instead of Δ_{\odot} .

$$\forall a \in [0, 1], \oplus(a, a) = a$$

It is obvious that idempotent operators ignore redundancy. Namely, if two different sources provide the same possibility distribution, then the result of their aggregation is simply the same possibility distribution. As an example of idempotent conjunctive operator, the well-known minimum (Min) operator defined as follows:

$$\pi_{min}(I) = \min(\pi_1(I), \dots, \pi_n(I))$$

As syntactic counterpart associated to $\pi_{min}(I)$, we have the following possibilistic knowledge base:

$$\mathcal{B}_{\oplus} = \mathcal{B}_1 \cup \mathcal{B}_2$$

In [Benferhat *et al.*, 2000], it was shown that, in some conditions, that the conjunctive operator may be considered as a majority merging operator. Formally, there exists n such that $(\mathcal{B}_1 \oplus \mathcal{B}_2^n) \models \mathcal{B}_2$ where \mathcal{B}_2^n is the aggregation of \mathcal{B}_2 with \oplus for n iterations.

Disjunctive operators When the different sources are conflicting and having the same reliability, it seems to be unsafe to privilege one source to another. Namely, if the union of two distinct knowledge bases \mathcal{B}_1 and \mathcal{B}_2 is inconsistent, then the fusion result should neither imply \mathcal{B}_1 nor \mathcal{B}_2 . Such situation requires to use a disjunctive operator, denoted by \otimes , and defined as follows:

$$\forall a \in [0, 1], \otimes(a, 1) = \otimes(1, a) = 1$$

As syntactic counterpart, one can associate to \otimes the following possibilistic knowledge base:

$$\mathcal{B}_{\otimes} = \{(\varphi_i \vee \phi_j, 1 - \oplus(1 - a_i, 1 - b_j))\}$$

According to [Benferhat *et al.*, 2000], a disjunctive operator may lead to a situation of total ignorance. To this end, another class of disjunctive operators, called regular operators was introduced. This class satisfies the following condition.

$$\forall a \neq 1, \forall b \neq 1, \otimes(a, b) \neq 1$$

Intuitively, if the different sources consider an information as somewhat certain, then when aggregating them this information must be also somewhat certain. Clearly, in the presence of inconsistency the fusion result recovers the common set between belief bases. Formally, a regular disjunctive operator is as follows:

$$\mathcal{B}_{\oplus}^* = \mathcal{B}_1^* \vee \mathcal{B}_2^*$$

where \mathcal{B}_i^* is the standard knowledge bases associated to \mathcal{B} by ignoring all the weights in \mathcal{B} .

Finally, note that the disjunctive regular operators are not appropriate when the sources are assumed to be consistent since these operators promote less informative beliefs.

Ontology merging Regarding ontology merging, there are few approaches on belief merging for description logics in the spirit of standard belief merging. However, the existing approaches (*e.g.* [Noy and Musen, 2000; Kotis *et al.*, 2006; Moguillansky and Falappa, 2007]) are mainly based on techniques proposed in ontology debugging which is closely related to inconsistency handling in propositional logic. Moreover, there is to the best of our knowledge no approach for merging prioritized DLs knowledge bases.

2.3.3 Inconsistency handling

In the artificial intelligence community, it is well-known that nonmonotonic reasoning and revision are considered as the two sides of the same coin [Makinson and Gärdenfors, 1989]. In general, revision or merging of different information sources may lead to inconsistency problems. Several approaches were proposed to deal with inconsistency in flat propositional logic knowledge bases (by a flat propositional logic knowledge base, we mean a base where all the formulas have the same priority) and prioritized knowledge bases. This is generally done through the definition of many suitable notions for consequence relations. There exist several attitudes when faced to inconsistency. In what follows, we recall the main approaches of inconsistency handling:

Restoring consistency Consists in getting rid of inconsistency by first computing the set of maximal or not maximally consistent subsets that restore consistency of the initial base, then using them to perform inference. Let K be an inconsistent knowledge base. Inference relation comes down first to compute maximally consistent subbases of K . A maximally consistent subset, denoted by K' , is defined as follows:

- $K' \subseteq K$
- K' is consistent
- if $K' \subset K'' \subseteq K$, then K'' is inconsistent

Let $MC(K)$ denote the set of maximally consistent subsets of K . Many inference strategies are proposed to select bases from $MC(K)$ in order to perform inference. The well-known inconsistency-tolerant inference relations are: the universal inference [Rescher and Manor, 1970], existential inference [Rescher and Manor, 1970], argued inference [Benferhat *et al.*, 1993b; Amgoud, 2005; Amgoud and Prade, 2009], cardinality-based inference [Benferhat *et al.*, 1997a], safe inference [Benferhat *et al.*, 1992]. An extension of these inference strategies is proposed when the knowledge base is layered or stratified (*i.e.* the formulas are attached with priorities) [Brewka, 1989; Benferhat *et al.*, 1998a]. In (*e.g.* [Lang and Marquis, 2002; Lang and Marquis, 2010; Konieczny *et al.*, 2005]) another approach, based on the notion of variable forgetting, is proposed.

Paraconsistent logics This consists in accepting inconsistency while coping with it by weakening inference relations. This can be done either by weakening logical connectors (see for instance [Besnard and Hunter, 1995; Hunter, 1998; Dubois *et al.*, 2003] for more details) or localizing inconsistency using, for instance, richer semantics (*e.g.* the multi-valued-semantics) (see for instance [Konieczny and Marquis, 2002; Konieczny *et al.*, 2008] for more details).

Inconsistency analysis : A standard knowledge base is either considered as completely consistent or completely inconsistent. In case of inconsistency, one cannot deduce meaningful conclusions. As mentioned in Section 2.2.2, one way for reasoning is to use the subset of formulas induced from the inconsistency degree to do reasoning. In the same way, many inconsistency approaches were proposed for analysing and measuring the inconsistency of a knowledge base in order to make inference (for more details, see [Hunter and Konieczny, 2005; Hunter and Konieczny, 2010; Grant and Hunter, 2011; Grant and Hunter, 2013] for example).

Argumentation framework : Argumentation frameworks [Besnard and Hunter, 2008] offers a good way to reason and make decision from conflicting pieces of information. Given a set of conflicting information, argumentation helps to identify pros and cons argument for a particular conclusion [Amgoud and Cayrol, 2002; Hunter and Konieczny, 2005; Bertossi *et al.*, 2005b].

Ontology debugging In the context of ontologies, several approaches are proposed to handle inconsistency. As explained above, there are works that dealt with inconsistency by: i) restoring consistency (e.g. [Kalyanpur *et al.*, 2005; Flouris *et al.*, 2006a; Lam *et al.*, 2008; Corcho *et al.*, 2009]); ii) using paraconsistent semantics (e.g. [Ma *et al.*, 2011; Zhou *et al.*, 2012]); and iii) measuring inconsistency (e.g. [Qi and Hunter, 2007; Ma and Hitzler, 2010]); and using argumentation (e.g. [Zhang and Lin, 2013; Croitoru and Vesic, 2013])

A crucially important problem that arises in Ontology-Based Data Access is how to manage inconsistency; otherwise the knowledge base is meaningless and useless. In such setting, inconsistency is defined with respect to some assertions that contradict the terminology. Typically, a *TBox* is usually verified and validated while the assertions can be provided in large quantities by various and unreliable sources and may contradict the *TBox*. Moreover, it is often too expensive to manually check and validate all the assertions. This is why it is very important in OBDA to reason in the presence of inconsistency. This consists in accepting and keeping the inconsistencies in the knowledge base while coping with them when performing inference (*i.e.* while answering queries).

Regarding *DL-Lite*, and especially within OBDA setting, existing works [Giacomo *et al.*, 2007; Lembo *et al.*, 2010; Bienvenu, 2012; Bienvenu and Rosati, 2013], basically inspired by the approaches proposed in the database area, tried to deal with inconsistency in *DL-Lite* by proposing and adapting several inconsistency-tolerant inference methods. All the proposed approaches are based on the notion of repair (restoring consistency) which is closely related to the notion of database repair defined in order to answer queries raised to inconsistent databases. A repair of a database contradicting a set of integrity constraints is a database obtained by applying a minimal set of changes in order to restore consistency. This notion of repair was extended to the *DL-Lite* setting by defining assertional-based repair for *DL-Lite* knowledge bases which is simply a maximal assertional subbase consistent with the terminology. In the *DL-Lite* framework, the notion of assertional-based reparation is in the spirit of maximal consistent subsets defined in propositional logic. Clearly an assertional-based reparation is a maximal assertional subbase of the *ABox* that is consistent with the *TBox*.

In this thesis, we are interested in repairing or restoring consistency for *DL-Lite* knowledge bases. Many inference strategies will be discussed in Chapter 5, within an OBDA setting, and in Chapter 7 when the *ABox* is prioritized.

2.4 Conclusion

In the first part of this chapter, we recalled the basic concepts of possibility theory which offers a natural way to deal with ordinal and qualitative uncertainty. We then reviewed the main extensions of description logics to handle uncertainty/imprecision when specifying ontologies. The second part of this chapter introduced the belief change problem (merging and revision) and the different techniques for handling inconsistency in propositional logic settings while recalling relevant related works done in the description logics area. The rest of this thesis investigates these issues within *DL-Lite* frameworks, especially when the knowledge base is prioritized. To this end, we need to extend the *DL-Lite* languages to support uncertainty or priority between axioms that represent a domain of interest. Next chapter studies this extension using a possibility theory setting.

PART II

On the possibilistic extension of *DL-Lite*

MIN-BASED POSSIBILISTIC *DL-Lite*

3.1 Introduction

Nowadays, in real world applications, knowledge is usually affected with uncertainty and imprecision. Recently, several works have been proposed to deal with probabilistic and non-probabilistic uncertainty [Dubois *et al.*, 2006; Lukasiewicz *et al.*, 2012a] on the one hand and to deal with fuzzy information [Bobillo and Straccia, 2012; Lukasiewicz and Straccia, 2009] on the other hand. A particular attention was given to fuzzy extensions of DLs (e.g. [Bobillo *et al.*, 2012; Bobillo *et al.*, 2013; Pan *et al.*, 2007; Straccia, 2006b; Straccia, 2013]). Besides, some works are devoted to possibilistic extensions of DLs (e.g. [Dubois *et al.*, 2006; Hollunder, 1995; Qi *et al.*, 2011]) which are basically based on standard reasoning services.

This chapter concerns the development of uncertainty-based *DL-Lite* using possibility theory. We focus on main fragments of *DL-Lite*. Namely *DL-Lite_{core}* which is the simplest *DL-Lite* language, *DL-Lite_F* and *DL-Lite_R* which underlie the *OWL2-QL* language. Indeed, we first develop our study on the extension of *DL-Lite_{core}* within a possibility theory setting. The restriction to *DL-Lite_{core}* is mainly motivated by the clarity and lightness of the language on which we can easily show how the extension of *DL-Lite* to the possibility theory can be achieved. We then extend our approach to richer *DL-Lite* languages such as *DL-Lite_F* or *DL-Lite_R*.

An important question addressed in this chapter is: “can one extend the expressive power of *DL-Lite*, to deal with possibilistic uncertain information, without increasing the computational cost?” This chapter provides a positive answer to this question.

Note first that some existing extensions of possibilistic DLs [Qi *et al.*, 2007b; Qi *et al.*, 2007a] may need some extra computational costs (although their inference process is still in P). For instance, in some existing approaches (e.g. [Qi *et al.*, 2007b; Qi *et al.*, 2007a]), computing inconsistency degrees comes down to achieve $\log_2(n)$ calls where n is the size of the uncertainty scale¹) to the inconsistency checking in standard (without uncertainty) DLs. This method, based on a dichotomy search, is closely related to the method proposed in [Dubois *et al.*, 1994] for computing inconsistency degrees of a possibilistic propositional knowledge base (see Section 2.2.2).

This chapter departs from several approaches for computing the inconsistency degrees of a knowledge base and follows another direction to achieve reasoning tasks in possibilistic *DL-Lite*. The idea is to slightly modify the algorithm for checking the inconsistency of a knowledge base used in standard *DL-Lite* by propagating the uncertainty degrees associated with axioms. The uncertainty propagation does not generate any extra computational cost.

Note that a tool for representing and reasoning in possibilistic *DL-Lite* framework is implemented. A description of this tool is provided in the appendix A.

The rest of this chapter is organized as follows: Section 3.2 rephrases the possibility theory framework over *DL-Lite* interpretations. Section 3.3 discusses the possibilistic extension of *DL-Lite_{core}*, denoted π -*DL-Lite_{core}*, where we present its syntax and its semantics. Section 3.4 introduces the so-called π -negated closure of a π -*DL-Lite_{core}* knowledge base. Section 3.5 gives a method to compute

¹number of levels in the used scale

the inconsistency degree of a π -*DL-Lite_{core}* knowledge base using query evaluations. Section 3.6 extends possibilistic *DL-Lite_{core}* to *DL-Lite_F* and *DL-Lite_R*, two important fragments of *DL-Lite* family. Section 3.7 studies different standard possibilistic inferences. Section 3.8 addresses the problem of query answering within π -*DL-Lite* framework. Section 3.9 deals with related works and Section 3.10 concludes the chapter.

3.2 Possibility distribution over *DL-Lite* interpretations

In this section, we rephrase the semantics of possibility theory over *DL-Lite* interpretations. Let \mathcal{L} be a finite *DL-Lite* description language as defined in Section 1.4, Ω be a universe of discourse (here represented by a set of *DL-Lite* interpretations) and $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}}) \in \Omega$ be a *DL-Lite* interpretation.

3.2.1 Possibility distribution

In the context of possibilistic *DL-Lite*, a possibility distribution is a mapping, denoted by π , from the universe of discourse Ω to the unit interval $[0, 1]$. It assigns to each interpretation $\mathcal{I} \in \Omega$ a possibility degree $\pi(\mathcal{I}) \in [0, 1]$ that represents its compatibility or consistency degree with respect to the set of available knowledge. When $\pi(\mathcal{I}) = 1$, we say that \mathcal{I} is totally possible and it is fully consistent with the set of available knowledge. When $\pi(\mathcal{I}) = 0$, we say that \mathcal{I} is impossible and it is fully inconsistent with the set of available knowledge. Two special cases exist:

1. Total ignorance when $\forall \mathcal{I} \in \Omega, \pi(\mathcal{I}) = 1$.
2. Complete knowledge when $\exists \mathcal{I}' \in \Omega, \pi(\mathcal{I}') = 1$ and $\forall \mathcal{I} \in \Omega, \mathcal{I}' \neq \mathcal{I}, \pi(\mathcal{I}) = 0$.

A possibility distribution π is said normalized if there exists at least one totally possible interpretation, namely $\exists \mathcal{I} \in \Omega$ such that $\pi(\mathcal{I}) = 1$, otherwise, we say that π is sub-normalized. The concept of normalization reflects the presence of conflicts in the set of available information. For two interpretations \mathcal{I} and \mathcal{I}' , we say that \mathcal{I} is more consistent or more compatible than \mathcal{I}' (with respect to the available knowledge) if $\pi(\mathcal{I}) > \pi(\mathcal{I}')$.

3.2.2 Possibility and necessity measures

Let us consider M to be a subset of Ω . Let \overline{M} be the complementary of M , namely $\overline{M} = \Omega \setminus M$. In a standard possibility theory, given a possibility distribution π , one can define two measures from 2^Ω to the interval $[0, 1]$ which discriminate between the plausibility and the certainty regarding the subset M . These two measures are:

Possibility measure A possibility measure, denoted by Π , is a function that assigns to each $M \subseteq \Omega$ a degree between $[0, 1]$. Given a possibility distribution π , $\Pi(M)$ is defined as:

$$\Pi(M) = \sup\{\pi(\mathcal{I}) : \mathcal{I} \in M\}.$$

$\Pi(M)$ evaluates to what extent the subset M is compatible with the available knowledge encoded by the possibility distribution π . When $\Pi(M)=1$, we say that M is totally possible if $\Pi(\overline{M}) = 0$ and M is somewhat possible if $\Pi(\overline{M}) \in]0, 1[$. When $\Pi(M)=1$ and $\Pi(\overline{M})=1$, we say that there is a total ignorance about M . A possibility measure Π satisfies the following properties for normalized possibility distributions:

$$\forall M \subseteq \Omega, \forall L \subseteq \Omega, \Pi(M \cup L) = \max(\Pi(M), \Pi(L)),$$

and

$$\forall M \subseteq \Omega, \forall L \subseteq \Omega, \Pi(M \cap L) \leq \min(\Pi(M), \Pi(L)).$$

Necessity measure A necessity measure, denoted by N , is a function dual to Π . It is defined from Π as follows:

$$N(M) = 1 - \Pi(\overline{M}).$$

$N(M)$ evaluates to what extent M is certainty entailed from available knowledge encoded by π . When $N(M)=1$, we say that M is certain. When $N(M) \in]0, 1[$, we say that φ is somewhat certain. When $N(M) = 0$ and $N(\overline{M}) = 0$, we say that there is a total ignorance about M .

A necessity measure N satisfies the following properties for normalized possibility distributions:

$$\forall M \subseteq \Omega, \forall L \subseteq \Omega, N(M \cap L) = \min(N(M), N(L)),$$

and

$$\forall M \subseteq \Omega, \forall L \subseteq \Omega, N(M \cup L) \geq \max(N(M), N(L)).$$

Remark 3.1. Clearly not all subsets of Ω , the universe of discourse defined over a set of interpretations, represent axioms of a *DL-Lite* language. Namely, if M is a subset of Ω , then it may happen that there is no ϕ , an axiom of a *DL-Lite* language, such that $M = [\phi]$ where $[\phi]$ denotes the models of ϕ . This is due to the fact that *DL-Lite* is not a very expressive language. For instance, assume that our vocabulary is composed of one concept A and two individuals a_1 and a_2 . Assume that we have two interpretations $\mathcal{I}_1 = (\Delta^{\mathcal{I}} = \{a_1, a_2\}, \cdot^{\mathcal{I}_1})$ and $\mathcal{I}_2 = (\Delta^{\mathcal{I}} = \{a_1, a_2\}, \cdot^{\mathcal{I}_2})$ such that $A^{\mathcal{I}_1} = \{a_1\}$ and $A^{\mathcal{I}_2} = \{a_2\}$. Clearly, $\{\mathcal{I}_1, \mathcal{I}_2\}$ does not correspond to any axiom of our *DL-Lite* language, since $\{\mathcal{I}_1, \mathcal{I}_2\}$ intuitively encodes the formula $A(a_1) \vee A(a_2)$, while the disjunction operator between two assertions is not allowed in the *DL-Lite* language.

In the following, possibility and necessity measures are assumed to be only defined over a *DL-Lite* language. If ϕ is an axiom of the *DL-Lite* language, we use $[\phi]$ to denote the set of models of ϕ , we define its associated possibility measure and its associated necessity measure respectively as follows:

$$\Pi([\phi]) = \sup_{\mathcal{I} \in \Omega} \{\pi(\mathcal{I}) : \mathcal{I} \models \phi\},$$

and

$$N([\phi]) = 1 - \sup_{\mathcal{I} \in \Omega} \{\pi(\mathcal{I}) : \mathcal{I} \not\models \phi\}.$$

where $\mathcal{I} \not\models \phi$ means that \mathcal{I} is not a model of ϕ .

Remark 3.2. In standard propositional possibilistic logic, the necessity measure is the dual of the possibility measure and it is defined by $N([\psi]) = 1 - \Pi([\neg\psi])$ where ψ is a propositional formula and $[\psi]$ is its associated propositional models (see Section 2.2.2). In possibilistic *DL-Lite*, the necessity measure cannot be defined as the dual of the possibility measure because the negation of a *DL-Lite* axiom is not allowed (except if ϕ denotes basic facts, i.e membership assertions of the form $A(a)$ or $P(a, b)$).

3.3 Possibilistic $DL-Lite_{core}$

In this section, we provide a possibilistic extension of $DL-Lite_{core}$, denoted by $\pi-DL-Lite_{core}$. We first present the syntax of $\pi-DL-Lite_{core}$. We then show how to generate the possibility distribution associated with a $\pi-DL-Lite_{core}$ knowledge base. The section also contains some properties of $\pi-DL-Lite_{core}$.

3.3.1 Syntax of $\pi-DL-Lite_{core}$

Let us consider \mathcal{L}_{core} a $DL-Lite_{core}$ description language recalled in Section 1.4. A $\pi-DL-Lite_{core}$ knowledge base is defined as follows:

Definition 3.1. A $\pi-DL-Lite_{core}$ knowledge base $\mathcal{K} = \{\langle \phi_i, \alpha_i \rangle : 1, \dots, n\}$ is a finite set of possibilistic axioms of the form $\langle \phi, \alpha \rangle$ where ϕ is an axiom expressed in \mathcal{L}_{core} language and $\alpha \in]0, 1]$ is the necessity (i.e. certainty) degree of ϕ .

Only somewhat certain information (namely $\alpha > 0$) is explicitly represented in a $\pi-DL-Lite$ knowledge base. A weighted axiom $\langle \phi, \alpha \rangle$ means that the certainty degree of ϕ is at least equal to α . The higher the degree α the more certain is the axiom or the fact. The degree α can be associated either with an inclusion assertion between concepts (TBox), or with a membership assertion (ABox). A $\pi-DL-Lite$ knowledge base \mathcal{K} will also be represented by a couple $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ where both elements in \mathcal{T} and \mathcal{A} are at least somewhat certain. Note that in the definition of possibilistic-DL knowledge base proposed by Hollunder in [Hollunder, 1995], a possibilistic axiom is attached with a possibility value or a necessity value. Here, we only represent certainty using necessity values.

Note that, if we consider $\forall \alpha_i, \alpha_i = 1$ then we represent a classical $DL-Lite$ knowledge base: $\mathcal{K}^* = \{\phi_i : \langle \phi_i, \alpha_i \rangle \in \mathcal{K}\}$.

Example 3.1. Let *Loyal*, *Discount*, *NeedBased*, *Impulse* and *Wandering* be five atomic concepts that represent different types of customers. Let *FidelityService* be an atomic concept that represents fidelity service to be accorded to loyal customers. Let *satisfiedBy* be an atomic role that represents whether a loyal customer is satisfied by a fidelity service. We consider the following $\pi-DL-Lite$ knowledge base where we use an ordinal uncertainty scale $\{0, \gamma_1, \dots, \gamma_n, 1\}$ with $0 < \gamma_1 < \gamma_2 < \dots < \gamma_n < 1$ and its possibilistic TBox \mathcal{T} contains the following axioms:

$$\begin{array}{ll} \langle Discount \sqsubseteq Loyal, \gamma_6 \rangle & \langle Impulse \sqsubseteq Discount, \gamma_3 \rangle \\ \langle NeedBased \sqsubseteq \neg Loyal, \gamma_2 \rangle & \langle NeedBased \sqsubseteq Discount, \gamma_8 \rangle \\ \langle Impulse \sqsubseteq \neg NeedBased, \gamma_{10} \rangle & \langle Wandering \sqsubseteq Impulse, \gamma_5 \rangle \\ \langle \exists satisfiedBy^- \sqsubseteq FidelityService, 1 \rangle & \langle \exists satisfiedBy \sqsubseteq Loyal, 1 \rangle \end{array}$$

and its possibilistic ABox \mathcal{A} contains the following axioms:

$$\begin{array}{ll} \langle Wandering(John), \gamma_4 \rangle & \langle Loyal(Mary), \gamma_1 \rangle \\ \langle NeedBased(John), \gamma_9 \rangle & \langle satisfiedBy(John, Gifts), \gamma_{11} \rangle \end{array}$$

This running example will be used in the rest of the chapter. ■

In a $\pi-DL-Lite$ knowledge base, the necessity degree attached with an axiom reflects its confidence and evaluates to what extent this axiom is considered as certain. For instance in Example 3.1, the axiom $\langle Discount \sqsubseteq Loyal, \gamma_6 \rangle$ states that “a Discount customer may be a Loyal customer with a certainty degree equal or greater than γ_6 ”. The degree 1 is used to represent fully certain pieces of information.

\mathcal{I}	\mathcal{I}	$\pi_{\mathcal{K}}$
\mathcal{I}_1	$Loyal=\{John, Mary\}, Discount=\{\}$ $Impulse=\{\}, NeedBased=\{John\}, Wandering=\{John\},$ $FidelityService=\{Gifts\}, satisfiedBy=\{(John, Gifts)\}$	$1-\gamma_8$
\mathcal{I}_2	$Loyal=\{John, Mary\}, Discount=\{John\}$ $Impulse=\{\}, NeedBased=\{John\}, Wandering=\{\}$ $FidelityService=\{Gifts\}, satisfiedBy=\{(John, Gifts)\}$	$1-\gamma_4$
\mathcal{I}_3	$Loyal=\{John, Mary\}, Discount=\{John\}$ $Impulse=\{John\}, NeedBased=\{John\}, Wandering=\{John\}$ $FidelityService=\{Gifts\}, satisfiedBy=\{(John, Gifts)\}$	$1-\gamma_{10}$

Table 3.1: Example of a possibility distribution $\pi_{\mathcal{K}}$ computed using Definition 3.2.

3.3.2 From a π -DL-Lite_{core} knowledge base to a π -DL-Lite_{core} possibility distribution

The semantics of π -DL-Lite_{core} is given by a possibility distribution, denoted by $\pi_{\mathcal{K}}$, defined over the set of all interpretations $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ of a DL-Lite language \mathcal{L}_{core} (see Section 3.2). As in standard possibilistic logic [Dubois et al., 1994], given a π -DL-Lite_{core} knowledge base \mathcal{K} , the possibility distribution induced by \mathcal{K} is defined as follows:

Definition 3.2. For every $\mathcal{I} \in \Omega$,

$$\pi_{\mathcal{K}}(\mathcal{I}) = \begin{cases} 1 & \text{if } \forall \langle \phi_i, \alpha_i \rangle \in \mathcal{K}, \mathcal{I} \models \phi_i \\ 1 - \max\{\alpha_i : \langle \phi_i, \alpha_i \rangle \in \mathcal{K}, \mathcal{I} \not\models \phi_i\} & \text{otherwise} \end{cases}$$

where \models is the satisfaction relation of DL-Lite formulas recalled in Section 1.4. $\langle \phi_i, \alpha_i \rangle \in \mathcal{K}$ means that $\langle \phi_i, \alpha_i \rangle$ either belongs to the TBox \mathcal{T} or to the ABox \mathcal{A} of \mathcal{K} .

Example 3.2. [Example 3.1 continued] Using Definition 3.2, Table 3.1 below gives possibility degree of three interpretations $\mathcal{I}_1, \mathcal{I}_2$ and \mathcal{I}_3 . We assume that $\Delta^{\mathcal{I}} = \{John, Mary, Gifts\}$ is the same for the three interpretations:

In this example, we can see that the interpretation \mathcal{I}_1 does not satisfy $\langle NeedBased \sqsubseteq \neg Loyal, \gamma_2 \rangle$, $\langle NeedBased \sqsubseteq Discount, \gamma_8 \rangle$ and $\langle Wandering \sqsubseteq Impulse, \gamma_5 \rangle$. The interpretation \mathcal{I}_2 does not satisfy $\langle NeedBased \sqsubseteq \neg Loyal, \gamma_2 \rangle$ and $\langle Wandering(John), \gamma_4 \rangle$. The interpretation \mathcal{I}_3 does not satisfy $\langle NeedBased \sqsubseteq \neg Loyal, \gamma_2 \rangle$ and $\langle Impulse \sqsubseteq \neg NeedBased, \gamma_{10} \rangle$. Hence, none of these interpretations is a model of \mathcal{K} . ■

A π -DL-Lite_{core} knowledge base \mathcal{K} is said to be consistent if its associated possibility distribution $\pi_{\mathcal{K}}$ is normalized, namely there exists an interpretation \mathcal{I} such that $\pi_{\mathcal{K}}(\mathcal{I})=1$. Otherwise, \mathcal{K} is said to be inconsistent and its inconsistency degree is defined semantically as follows:

Definition 3.3. The inconsistency degree of a π -DL-Lite_{core} knowledge base \mathcal{K} , denoted by $Inc(\mathcal{K})$, is semantically defined as follows:

$$Inc(\mathcal{K}) = 1 - \max_{\mathcal{I} \in \Omega} \{\pi_{\mathcal{K}}(\mathcal{I})\}.$$

If $Inc(\mathcal{K}) = 1$ then \mathcal{K} is fully inconsistent and if $Inc(\mathcal{K})=0$ then it is simply said to be consistent. One can easily check that $Inc(\mathcal{K}) = 1 - \Pi([\top])$, where Π is the possibility measure defined in Section 3.2 and \top is a tautology.

Example 3.3. [Example 3.2 continued] One can check that the inconsistency degree of \mathcal{K} according to $\pi_{\mathcal{K}}$ is: $Inc(\mathcal{K}) = 1 - \max_{\mathcal{I} \in \Omega} \{\pi_{\mathcal{K}}(\mathcal{I})\} = \gamma_4$, and hence \mathcal{K} is inconsistent (in fact, there is no way to find an interpretation that satisfies \mathcal{K} with a degree greater than γ_4). ■

The inconsistency degree allows to define different inference processes as follows:

Definition 3.4 (Flat inference). Let \mathcal{K} be a π -*DL-Lite_{core}* knowledge base, $\pi_{\mathcal{K}}$ be the possibility distribution associated with \mathcal{K} and ϕ be a *DL-Lite* axiom. $\mathcal{K} \models_{\pi} \phi$ if and only if $N_{\pi}(\phi) > Inc(\mathcal{K})$ where N_{π} is the necessity measure induced by $\pi_{\mathcal{K}}$.

Here, what is important is just to know whether the conclusion is plausible or not.

The following definition extends Definition 3.4, by requiring that a conclusion should be entailed with some degree.

Definition 3.5 (Weighted inference). Let \mathcal{K} be a π -*DL-Lite_{core}* knowledge base, $\pi_{\mathcal{K}}$ be the possibility distribution associated with \mathcal{K} and ϕ be a *DL-Lite* axiom. $\mathcal{K} \models_{\pi} (\phi, \alpha)$ if and only if $N_{\pi}(\phi) = \alpha > Inc(\mathcal{K})$ where N_{π} is the necessity measure induced by $\pi_{\mathcal{K}}$.

A method based on inconsistency computation for implementing inferences of Definitions 3.4 and Definition 3.5 is given in Section 3.7. Besides, the two kinds of inference detailed in this chapter are :

- Flat entailment, from which one is only interested to know whether an axiom is entailed or not from a possibilistic *DL-Lite* knowledge base,
- Weighted entailment, where given a weight α , one is interested whether an axiom can be inferred with this specified degree.

Now if one is interested to compute the maximal degree of entailment of an axiom, we can use dichotomic search. The dichotomic search is applied over the set of all degrees used in the knowledge base (in both weighed TBox and weighted ABox).

Remark 3.3. In a propositional possibilistic logic setting, each possibilistic knowledge base induces a unique joint possibility distribution and each possibility distribution can be represented by a possibilistic knowledge base. Although each π -*DL-Lite* knowledge base induces a unique joint possibility distribution, the converse does not always hold.

Consider again the example where we only have one concept A and two individuals a_1 and a_2 . Consider four interpretations $\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3$ and \mathcal{I}_4 having the same domain $\Delta^{\mathcal{I}} = \{a_1, a_2\}$ where $(A)^{\mathcal{I}_1} = \{a_1\}$, $(A)^{\mathcal{I}_2} = \{a_2\}$, $(A)^{\mathcal{I}_3} = \{a_1, a_2\}$ and $(A)^{\mathcal{I}_4} = \emptyset$. Assume that $\pi(\mathcal{I}_1) = \pi(\mathcal{I}_2) = 1$ and $\pi(\mathcal{I}_3) = \pi(\mathcal{I}_4) = .5$. One can check that there is no π -*DL-Lite* knowledge base such that $\pi_{\mathcal{K}} = \pi$. This remark has no incidence on the results of this chapter. It simply points out some differences between standard propositional possibilistic logic and possibilistic *DL-Lite_{core}*.

3.3.3 Logical properties of π -*DL-Lite_{core}*

In the following, we present some properties of π -*DL-Lite_{core}*. These properties simply show that one can add conjunctions (*resp.* disjunctions) in the right side (*resp.* left side) of weighted inclusion axioms. Such results already hold in standard *DL-Lite_{core}* [Calvanese et al., 2007a] and in standard possibilistic logic [Dubois et al., 1994]. For the sake of clarity, we rephrase them for possibilistic π -*DL-Lite_{core}*.

Proposition 3.1 shows that a complex inclusion axiom of the form $\langle B_1 \sqsubseteq C_1 \sqcap C_2, \alpha \rangle$ can be splitted into two elementary inclusion axioms that can be added to the π -*DL-Lite_{core}* knowledge base without modifying its possibility distribution.

Proposition 3.1. *Let $\mathcal{K} = \{\langle B \sqsubseteq C_1 \sqcap C_2, \alpha \rangle, \mathcal{A}\}$ and $\mathcal{K}' = \{\langle B \sqsubseteq C_1, \alpha \rangle, \langle B \sqsubseteq C_2, \alpha \rangle, \mathcal{A}\}$ be two π -DL-Lite_{core} knowledge bases. Then \mathcal{K} and \mathcal{K}' induce the same possibility distribution, namely $\forall \mathcal{I} \in \Omega, \pi_{\mathcal{K}}(\mathcal{I}) = \pi_{\mathcal{K}'}(\mathcal{I})$.*

Proof of Proposition 3.1. The proof is immediate. Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be an interpretation. Assume that $\mathcal{I} \models \langle B \sqsubseteq C_1 \sqcap C_2, \alpha \rangle$. By definition of the satisfaction relation, this means that $(B)^{\mathcal{I}} \subseteq ((C_1)^{\mathcal{I}} \cap (C_2)^{\mathcal{I}})$. Hence $(B)^{\mathcal{I}} \subseteq (C_1)^{\mathcal{I}}$ and $(B)^{\mathcal{I}} \subseteq (C_2)^{\mathcal{I}}$, which means that $\mathcal{I} \models \langle B \sqsubseteq C_1, \alpha \rangle$ and $\mathcal{I} \models \langle B \sqsubseteq C_2, \alpha \rangle$. Therefore $\pi_{\mathcal{K}}(\mathcal{I}) = \pi_{\mathcal{K}'}(\mathcal{I})$.

Conversely, assume that $\mathcal{I} \models \langle B \sqsubseteq C_1, \alpha \rangle$ and $\mathcal{I} \models \langle B \sqsubseteq C_2, \alpha \rangle$. By definition of the satisfaction relation, this again means that $(B)^{\mathcal{I}} \subseteq (C_1)^{\mathcal{I}}$ and $(B)^{\mathcal{I}} \subseteq (C_2)^{\mathcal{I}}$. Hence $(B)^{\mathcal{I}} \subseteq ((C_1)^{\mathcal{I}} \cap (C_2)^{\mathcal{I}})$ which means that $\mathcal{I} \models \langle B \sqsubseteq C_1 \sqcap C_2, \alpha \rangle$. Therefore $\pi_{\mathcal{K}}(\mathcal{I}) = \pi_{\mathcal{K}'}(\mathcal{I})$.

The other cases, where $\mathcal{I} \not\models \langle B \sqsubseteq C_1 \sqcap C_2, \alpha \rangle$ or $\mathcal{I} \not\models \langle B \sqsubseteq C_1, \alpha \rangle$ and $\mathcal{I} \not\models \langle B \sqsubseteq C_2, \alpha \rangle$ follow similarly. \square

Proposition 3.2 shows that a complex inclusion axiom of the form $\langle B_1 \sqcup B_2 \sqsubseteq C, \alpha \rangle$ can be splitted into two elementary inclusion axioms that can be added to a π -DL-Lite_{core} knowledge base without modifying its possibility distribution.

Proposition 3.2. *Let $\mathcal{K} = \{\langle B_1 \sqcup B_2 \sqsubseteq C, \alpha \rangle, \mathcal{A}\}$ and $\mathcal{K}' = \{\langle B_1 \sqsubseteq C, \alpha \rangle, \langle B_2 \sqsubseteq C, \alpha \rangle, \mathcal{A}\}$ be two π -DL-Lite_{core} knowledge bases. Then \mathcal{K} and \mathcal{K}' induce the same possibility distribution, namely $\forall \mathcal{I} \in \Omega, \pi_{\mathcal{K}}(\mathcal{I}) = \pi_{\mathcal{K}'}(\mathcal{I})$.*

Proof of Proposition 3.2. The proof of Proposition 3.2 is similar to the one of Proposition 3.1. \square

3.4 Possibilistic negated closure in π -DL-Lite_{core}

The aim of this section is to define the so-called π -negated closure of a π -DL-Lite_{core} knowledge base. This notion is crucial for characterizing the concepts of consistency and inference from a π -DL-Lite_{core} knowledge base.

3.4.1 Rules used to obtain π -negated closure

A possibilistic π -DL-Lite_{core} TBox $\mathcal{T} = \{\mathcal{T}_p, \mathcal{T}_n\}$ can be viewed as composed of positive inclusion axioms of the form $\langle B_1 \sqsubseteq B_2, \alpha \rangle$ and negative inclusion axioms of the form $\langle B_1 \sqsubseteq \neg B_2, \alpha \rangle$. The possibilistic negated closure, denoted by π -neg(\mathcal{T}), will contain all the possibilistic negated axioms of the form $\langle B_1 \sqsubseteq \neg B_2, \alpha \rangle$ that can be derived from \mathcal{T} . Roughly speaking, the set π -neg(\mathcal{T}) is obtained by applying a set of three rules that extend the ones defined in standard DL-Lite_{core} when axioms are weighted with certainty degrees.

At the beginning π -neg(\mathcal{T}), is set to an empty set.

Rule 3.1. *Let $\mathcal{T} = \{\mathcal{T}_p, \mathcal{T}_n\}$ then add all negated axioms of \mathcal{T} to π -neg(\mathcal{T}).*

Example 3.4. [Example 3.1 continued] Using Rule 3.1, we add $\langle \text{NeedBased} \sqsubseteq \neg \text{Loyal}, \gamma_2 \rangle$ and $\langle \text{Impulse} \sqsubseteq \neg \text{NeedBased}, \gamma_{10} \rangle$ to π -neg(\mathcal{T}). \blacksquare

The first rule states that negative axioms that are explicitly stated in \mathcal{K} are trivially entailed from \mathcal{K} , and hence can be added to π -neg(\mathcal{T}).

Rule 3.2. *If $\langle B_1 \sqsubseteq B_2, \alpha_1 \rangle \in \mathcal{T}$ and $\langle B_2 \sqsubseteq \neg B_3, \alpha_2 \rangle \in \pi$ -neg(\mathcal{T}) then add $\langle B_1 \sqsubseteq \neg B_3, \min(\alpha_1, \alpha_2) \rangle$ to π -neg(\mathcal{T}).*

The second rule expresses transitivity relation induced by the inclusion assertion relation.

Rule 3.3. *If $\langle B_1 \sqsubseteq B_2, \alpha_1 \rangle \in \mathcal{T}$ and $\langle B_3 \sqsubseteq \neg B_2, \alpha_2 \rangle \in \pi\text{-neg}(\mathcal{T})$ then add $\langle B_1 \sqsubseteq \neg B_3, \min(\alpha_1, \alpha_2) \rangle$ to $\pi\text{-neg}(\mathcal{T})$.*

Remark 3.4. Note that instead of Rule 3.3 one can define the following rule:

If $\langle B_1 \sqsubseteq \neg B_2, \alpha \rangle \in \pi\text{-neg}(\mathcal{T})$ then add $\langle B_2 \sqsubseteq \neg B_1, \alpha \rangle \in \pi\text{-neg}(\mathcal{T})$ and then re-use Rule 3.2.

As we will see in Proposition 3.3, these rules (Rules 3.1-3.3) are enough for $\pi\text{-DL-Lite}_{core}$. In particular, these rules will be useful to equivalently define an efficient inference using directly $\pi\text{-neg}(\mathcal{T})$. As it will be shown later, the minimum operation used in the rules for propagating certainty degrees is justified by the fact that the joint possibility distribution will not be affected if the derived inclusion relations are added to the knowledge base. Lastly, when the degrees α_i 's are equal to 1, then $\pi\text{-neg}(\mathcal{T})$ simply collapses with the standard negated closure defined for standard $DL\text{-Lite}_{core}$ knowledge bases. In fact, $\pi\text{-neg}(\mathcal{T})$ extends standard $DL\text{-Lite}_{core}$ when one only deals with fully certain pieces of information.

Example 3.5. [Example 3.1 continued] Using $\pi\text{-neg}(\mathcal{T})$ of Example 3.4, Rules 3.2-3.3 allow to derive the following negative axioms:

- $\langle Discount \sqsubseteq Loyal, \gamma_6 \rangle$ and $\langle NeedBased \sqsubseteq \neg Loyal, \gamma_2 \rangle$ lead to adding $\langle Discount \sqsubseteq \neg NeedBased, \gamma_2 \rangle$ to $\pi\text{-neg}(\mathcal{T})$.
- $\langle Impulse \sqsubseteq Discount, \gamma_3 \rangle$ and $\langle Discount \sqsubseteq \neg NeedBased, \gamma_2 \rangle$ lead to adding $\langle Impulse \sqsubseteq \neg NeedBased, \gamma_2 \rangle$ to $\pi\text{-neg}(\mathcal{T})$.
- $\langle NeedBased \sqsubseteq Discount, \gamma_8 \rangle$ and $\langle Discount \sqsubseteq \neg NeedBased, \gamma_2 \rangle$, lead to adding $\langle NeedBased \sqsubseteq \neg NeedBased, \gamma_2 \rangle$ to $\pi\text{-neg}(\mathcal{T})$. ■

Next lemma deals with redundancy and simply states that an axiom does not need to appear several times in a knowledge base. It is enough to keep the one having the highest degree.

Lemma 3.1. *Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a $\pi\text{-DL-Lite}_{core}$ knowledge base. Let $B_1 \sqsubseteq B_2$ (or $B_1 \sqsubseteq \neg B_2$) be a TBox axiom and X be an ABox assertion.*

1. *Assume that $\langle B_1 \sqsubseteq B_2, \alpha_1 \rangle$ and $\langle B_1 \sqsubseteq B_2, \alpha_2 \rangle$ belong to \mathcal{T} . Then \mathcal{K} and $\mathcal{K}' = \langle \mathcal{T}', \mathcal{A} \rangle$ where $\mathcal{T}' = (\mathcal{T} \setminus \{\langle B_1 \sqsubseteq B_2, \alpha_1 \rangle, \langle B_1 \sqsubseteq B_2, \alpha_2 \rangle\}) \cup \{\langle B_1 \sqsubseteq B_2, \max(\alpha_1, \alpha_2) \rangle\}$ are equivalent in the sense that for all $\mathcal{I} \in \Omega$, $\pi_{\mathcal{K}}(\mathcal{I}) = \pi_{\mathcal{K}'}(\mathcal{I})$.*
2. *Similarly, assume that $\langle X, \alpha_1 \rangle$ and $\langle X, \alpha_2 \rangle$ belong to \mathcal{A} . Then \mathcal{K} and $\mathcal{K}' = \langle \mathcal{T}, \mathcal{A}' \rangle$ where $\mathcal{A}' = (\mathcal{A} \setminus \{\langle X, \alpha_1 \rangle, \langle X, \alpha_2 \rangle\}) \cup \{\langle X, \max(\alpha_1, \alpha_2) \rangle\}$ are equivalent in the sense that for all $\mathcal{I} \in \Omega$, $\pi_{\mathcal{K}}(\mathcal{I}) = \pi_{\mathcal{K}'}(\mathcal{I})$.*

Proof of Lemma 3.1. The proof of the lemma immediately follows from the definition of the possibility distribution associated with a $\pi\text{-DL-Lite}_{core}$ knowledge base. □

Example 3.6. [Example continued] In the $\pi\text{-neg}(\mathcal{T})$ of Examples 3.4 and 3.5, we observe that we derive both $\langle Impulse \sqsubseteq \neg NeedBased, \gamma_{10} \rangle$ and $\langle Impulse \sqsubseteq \neg NeedBased, \gamma_2 \rangle$. Using Lemma 3.1, we only keep $\langle Impulse \sqsubseteq \neg NeedBased, \gamma_{10} \rangle$ in $\pi\text{-neg}(\mathcal{T})$. ■

Example 3.7. [Example continued] We now give the $\pi\text{-neg}(\mathcal{T})$ of the Example 3.1 using Rules 3.1-3.3 and Lemma 3.1:

$$\begin{array}{ll}
 \langle NeedBased \sqsubseteq \neg Loyal, \gamma_2 \rangle & \langle NeedBased \sqsubseteq \neg NeedBased, \gamma_2 \rangle \\
 \langle Impulse \sqsubseteq \neg NeedBased, \gamma_{10} \rangle & \langle Wandering \sqsubseteq \neg NeedBased, \gamma_5 \rangle \\
 \langle Discount \sqsubseteq \neg NeedBased, \gamma_2 \rangle & \langle \exists satisfiedBy \sqsubseteq \neg NeedBased, \gamma_2 \rangle
 \end{array}$$

■

3.4.2 Properties of π -negated closure

This subsection gives some properties of the π -negated closure of a π -DL-Lite_{core}. Given $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ a π -DL-Lite_{core} knowledge base, we define the α -cut of \mathcal{K} (resp. \mathcal{T} and \mathcal{A}), denoted by $\mathcal{K}_{\geq \alpha}$ (resp. $\mathcal{T}_{\geq \alpha}$ and $\mathcal{A}_{\geq \alpha}$), the subbase of \mathcal{K} (resp. \mathcal{T} and \mathcal{A}) composed of axioms having weights α_i that are at least equal to α and the strict α -cut of \mathcal{K} (resp. \mathcal{T} and \mathcal{A}), denoted by $\mathcal{K}_{> \alpha}$ (resp. $\mathcal{T}_{> \alpha}$, $\mathcal{A}_{> \alpha}$), as a subbase of \mathcal{K} (resp. \mathcal{T} and \mathcal{A}) composed of axioms having weights α_i strictly greater than α .

Lemma 3.2. *Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a π -DL-Lite_{core} knowledge base. Let α_1 and α_2 be two degrees in $[0, 1]$ such that $\alpha_1 > \alpha_2$. Then:*

$$\pi\text{-neg}(\mathcal{T}_{\geq \alpha_1}) \subseteq \pi\text{-neg}(\mathcal{T}_{\geq \alpha_2})$$

Proof of Lemma 3.2. The proof is immediate. Indeed, to obtain $\pi\text{-neg}(\mathcal{T}_{\geq \alpha_2})$, one may apply Rules 3.1-3.3 to all axioms of $\mathcal{T}_{\geq \alpha_2}$ and implicitly $\mathcal{T}_{\geq \alpha_1}$, since $\mathcal{T}_{\geq \alpha_1} \subseteq \mathcal{T}_{\geq \alpha_2}$. This leads to trivially obtain $\pi\text{-neg}(\mathcal{T}_{\geq \alpha_1})$ by re-applying again Rules 3.1-3.3 on remaining weighted axioms of $\mathcal{T}_{\geq \alpha_1}$. \square

The following lemma states that deriving negative axioms with weights greater or equal to α can be equivalently done either by deriving all weighted negative axioms then select those having a weight greater or equal to α , or select initial axioms from \mathcal{K} having a weight greater or equal to α then apply the negative closure of this subbase of \mathcal{K} .

Lemma 3.3. *Let \mathcal{K} be a π -DL-Lite_{core} knowledge base. Let $\pi\text{-neg}(\mathcal{T})$ be the possibilistic negative closure of \mathcal{K} . Then:*

$$\pi\text{-neg}(\mathcal{T})_{\geq \alpha} = \pi\text{-neg}(\mathcal{T}_{\geq \alpha})$$

Proof of Lemma 3.3. The proof of this lemma can be obtained by first noticing that to compute $\pi\text{-neg}(\mathcal{T})$ one may start with only axioms having a weight greater or equal to α . This leads to $\pi\text{-neg}(\mathcal{T}_{\geq \alpha})$. Now, applying Rules 3.1-3.3 to axioms with weights strictly less than α leads to derive negative axioms with weights also less than α . Hence, they will not belong to $\pi\text{-neg}(\mathcal{T})_{\geq \alpha}$. \square

Proposition 3.3 states that adding all negative axioms of $\pi\text{-neg}(\mathcal{T})$ to \mathcal{T} does not change the induced possibility distribution.

Proposition 3.3. *Let $\mathcal{T} = \{\mathcal{T}_p, \mathcal{T}_n\}$ and $\pi\text{-neg}(\mathcal{T})$ be the negated closure of \mathcal{T} obtained using Rules (3.1-3.3). Then $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ and its $\mathcal{K}' = \langle \mathcal{T} \cup \pi\text{-neg}(\mathcal{T}), \mathcal{A} \rangle$ induce the same possibility distribution, namely $\forall \mathcal{I} \in \Omega : \pi_{\mathcal{K}}(\mathcal{I}) = \pi_{\mathcal{K}'}(\mathcal{I})$.*

Proof of Proposition 3.3. It is sufficient to show that one application of Rule 3.2 (or Rule 3.3) does not modify the possibility distribution. It is enough then to repeat the application of Rule 3.2 (or Rule 3.3) on the obtained and derived negative inclusion axioms. Assume that $\langle B_1 \sqsubseteq B_2, \alpha_1 \rangle \in \mathcal{T}$ and $\langle B_2 \sqsubseteq \neg B_3, \alpha_2 \rangle \in \mathcal{T}$. Let us show that the result of applying Rule 3.2, which leads to add $\langle B_1 \sqsubseteq \neg B_3, \min(\alpha_1, \alpha_2) \rangle$ does not modify the possibility distribution. Namely, $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ and $\mathcal{K}' = \langle \mathcal{T} \cup \{\langle B_1 \sqsubseteq \neg B_3, \min(\alpha_1, \alpha_2) \rangle\}, \mathcal{A} \rangle$ are equivalent. Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be an interpretation. We consider three cases:

1. $\mathcal{I} \models B_1 \sqsubseteq B_2$ and $\mathcal{I} \models B_2 \sqsubseteq \neg B_3$. By definition of the satisfaction relation, this means that: $(B_1)^{\mathcal{I}} \subseteq (B_2)^{\mathcal{I}}$ and $(B_2)^{\mathcal{I}} \subseteq (\neg B_3)^{\mathcal{I}}$. Hence, $(B_1)^{\mathcal{I}} \subseteq (\neg B_3)^{\mathcal{I}}$, which means that $\mathcal{I} \models B_1 \sqsubseteq \neg B_3$. Therefore $\pi_{\mathcal{K}}(\mathcal{I}) = \pi_{\mathcal{K}'}(\mathcal{I})$.

2. $\mathcal{I} \models B_1 \sqsubseteq B_2$ and $\mathcal{I} \models B_2 \sqsubseteq \neg B_3$ (the other case, where $\mathcal{I} \not\models B_1 \sqsubseteq B_2$ and $\mathcal{I} \models B_2 \sqsubseteq \neg B_3$, follows similarly). Let $\mathcal{K}'' = \mathcal{K} \setminus \{\langle B_1 \sqsubseteq B_2, \alpha_1 \rangle, \langle B_2 \sqsubseteq \neg B_3, \alpha_2 \rangle\}$. We have:

$$\begin{aligned} \pi_{\mathcal{K}}(\mathcal{I}) &= \min(\pi_{\mathcal{K}''}(\mathcal{I}), 1 - \alpha_2) \\ &= \min(\pi_{\mathcal{K}''}(\mathcal{I}), 1 - \alpha_2, 1 - \min(\alpha_1, \alpha_2)) \\ &= \pi_{\mathcal{K}'}(\mathcal{I}) \end{aligned}$$

3. $\mathcal{I} \not\models B_1 \sqsubseteq B_2$ and $\mathcal{I} \not\models B_2 \sqsubseteq \neg B_3$. Again let $\mathcal{K}'' = \mathcal{K} \setminus \{\langle B_1 \sqsubseteq B_2, \alpha_1 \rangle, \langle B_2 \sqsubseteq \neg B_3, \alpha_2 \rangle\}$. We have

$$\begin{aligned} \pi_{\mathcal{K}}(\mathcal{I}) &= \min(\pi_{\mathcal{K}''}(\mathcal{I}), 1 - \alpha_1, 1 - \alpha_2) \\ &= \min(\pi_{\mathcal{K}''}(\mathcal{I}), 1 - \alpha_1, 1 - \alpha_2, 1 - \min(\alpha_1, \alpha_2)) \\ &= \pi_{\mathcal{K}'}(\mathcal{I}). \end{aligned}$$

□

3.5 Checking inconsistency degrees

In this section, we show how to compute the inconsistency degree of a π -*DL-Lite_{core}* knowledge base using query evaluations.

3.5.1 Additional properties of π -*neg*(\mathcal{T})

In the previous section (Section 3.4), we showed that adding π -*neg*(\mathcal{T}) to \mathcal{T} does not modify the joint possibility distribution (Proposition 3.3). This subsection shows that computing the inconsistency degree of $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ comes down to compute the inconsistency degree of $\mathcal{K}' = \langle \pi$ -*neg*(\mathcal{T}), $\mathcal{A} \rangle$.

We first introduce the two following technical lemmas.

Lemma 3.4. *Let \mathcal{K} be a π -DL-Lite_{core} knowledge base. Let \mathcal{K}^* be the standard DL-Lite_{core} knowledge base obtained from \mathcal{K} by ignoring the weights associated with axioms of \mathcal{K} . Let $cln(\mathcal{T}^*)$ be the negated closure (NI-closure) defined in [Calvanese et al., 2007a]. Then:*

$$cln(\mathcal{T}^*) = \{B_1 \sqsubseteq \neg B_2 : \langle B_1 \sqsubseteq \neg B_2, \alpha \rangle \in \pi$$
-*neg*($\mathcal{T})\}$

This lemma states that our definition of π -*neg*(\mathcal{T}) recovers the one used in standard DL-Lite_{core}. Namely, we derive the same set of negative axioms as in standard DL-Lite_{core} knowledge base. However, in our approach the negative axioms are attached with certainty degrees.

Proof of Lemma 3.4. The proof of Lemma 3.4 follows from the fact that removing weights from Rules 3.1-3.3 gives exactly the same rules used in [Calvanese et al., 2007a] for deriving the negated closure, denoted by $cln(\mathcal{T})$, of a standard DL-Lite_{core} knowledge base $\mathcal{B} = \langle \mathcal{T}, \mathcal{A} \rangle$. □

Lemma 3.5. *Let \mathcal{K} be a π -DL-Lite_{core} knowledge base. Then*

$$cln(\mathcal{T}_{\geq \alpha}^*) = \{B_1 \sqsubseteq \neg B_2 : \langle B_1 \sqsubseteq \neg B_2, \beta \rangle \in \pi$$
-*neg*($\mathcal{T}) \text{ and } \beta \geq \alpha\}$

Lemma 3.5 is in the spirit of Lemma 3.4. It states that the negative closure of a subbase of \mathcal{K} can be recovered from π -*neg*(\mathcal{T}).

Proposition 3.4. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ and let $\mathcal{K}' = \langle \pi\text{-neg}(\mathcal{T}), \mathcal{A} \rangle$. Then:

$$\text{Inc}(\mathcal{K}) = \text{Inc}(\mathcal{K}')$$

Proof of Proposition 3.4. Recall that in standard *DL-Lite*, a *DL-Lite_{core}* knowledge base $\mathcal{K}_s = \langle \mathcal{T}_s, \mathcal{A}_s \rangle$ is inconsistent if and only if the knowledge base $\langle \text{cln}(\mathcal{T}_s), \mathcal{A}_s \rangle$ is inconsistent, where $\text{cln}(\mathcal{T}_s)$ is the negative closure of \mathcal{K}_s defined in [Calvanese et al., 2007a] (see Section 1.4).

Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a π -*DL-Lite_{core}* knowledge base. Now assume that $\text{Inc}(\mathcal{K}) = \alpha$. This means that $\mathcal{K}_{>\alpha}$ is consistent and $\mathcal{K}_{\geq\alpha}$ is inconsistent. This also means that (using the above lemmas):

$$\langle \text{cln}(\mathcal{T}), \mathcal{A} \rangle_{>\alpha} \text{ is consistent,}$$

and

$$\langle \text{cln}(\mathcal{T}), \mathcal{A} \rangle_{\geq\alpha} \text{ is inconsistent.}$$

Now, using Lemma 3.5, this also means that:

$$\langle \pi - \text{neg}(\mathcal{T}), \mathcal{A} \rangle_{>\alpha} \text{ is consistent}$$

while

$$\langle \pi - \text{neg}(\mathcal{T}), \mathcal{A} \rangle_{\geq\alpha} \text{ is inconsistent,}$$

which means that the inconsistency degree of $\mathcal{K}' = \langle \pi - \text{neg}(\mathcal{T}), \mathcal{A} \rangle$ is equal to α . The converse follows in a similar way. \square

Proposition 3.4 is important since it provides a way to compute the inconsistency degree of a π -*DL-Lite_{core}* knowledge base. Indeed, computing the inconsistency degree of $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ is reduced to computing the inconsistency degree of $\mathcal{K}' = \langle \pi - \text{neg}(\mathcal{T}), \mathcal{A} \rangle$.

3.5.2 Computing inconsistency degrees in π -*DL-Lite_{core}*

We now provide a characterization of the inconsistency degree of a π -*DL-Lite_{core}* knowledge base by only focusing on $\langle \pi\text{-neg}(\mathcal{T}), \mathcal{A} \rangle$. First recall that the ABox only contains positive membership assertions (facts). Hence, the ABox alone is always consistent. Similarly, the TBox $\pi\text{-neg}(\mathcal{T})$ alone (namely, when ABox= \emptyset) is also consistent. Indeed, it is easy to define an interpretation \mathcal{I} which is a model of $\pi\text{-neg}(\mathcal{T})$. For each $\langle B_i \sqsubseteq \neg B_j, \alpha \rangle \in \pi\text{-neg}(\mathcal{T})$, we let $(B_i)^{\mathcal{I}} = \emptyset$ if B_i is a concept and $(R)^{\mathcal{I}} = \emptyset$ if B_i is of the form $\exists R$ or $\exists R^-$ and R is a role. \mathcal{I} is then trivially a model of $\pi\text{-neg}(\mathcal{T})$. Hence, pieces responsible of inconsistency should involve both elements from $\pi\text{-neg}(\mathcal{T})$ and \mathcal{A} .

We now introduce the concept of a conflict and to what extent its elements are conflicting.

Definition 3.6. A conflict \mathcal{C} of $\langle \pi\text{-neg}(\mathcal{T}), \mathcal{A} \rangle$ is a subbase of $\langle \pi\text{-neg}(\mathcal{T}), \mathcal{A} \rangle$ of the form

$$\{\langle B_1 \sqsubseteq \neg B_2, \alpha_1 \rangle, \langle X, \alpha_2 \rangle, \langle Y, \alpha_3 \rangle\}$$

such that:

- $\langle B_1 \sqsubseteq \neg B_2, \alpha_1 \rangle \in \pi\text{-neg}(\mathcal{T})$.
- $\langle X, \alpha_2 \rangle \in \mathcal{A}$ and $\langle Y, \alpha_3 \rangle \in \mathcal{A}$ with X and Y are such that there exist two individuals a and b where:

$$X = \begin{cases} A(a) & \text{if } B_1 \text{ is a basic concept } A, \\ P(a, b) & \text{if } B_1 \text{ is of the form } \exists P \text{ and } P \text{ is a role,} \\ P(b, a) & \text{if } B_1 \text{ is of the form } \exists P^- \text{ and } P \text{ is a role.} \end{cases}$$

$$Y = \begin{cases} A(a) & \text{if } B_2 \text{ is a basic concept } A, \\ P(a, b) & \text{if } B_2 \text{ is of the form } \exists P \text{ and } P \text{ is a role,} \\ P(b, a) & \text{if } B_2 \text{ is of the form } \exists P^- \text{ and } P \text{ is a role.} \end{cases}$$

Besides, an inconsistency problem is always defined with respect to some ABox assertions and a TBox axiom, since a TBox may be incoherent but never inconsistent. Before introducing the property of conflict in π -DL-Lite_{core}, let us first remind the Calvanese *et al.* result [Calvanese *et al.*, 2010].

Lemma 3.6. *Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a DL-Lite knowledge base. If \mathcal{K} is inconsistent, then there exists a subset $\mathcal{A}_0 \subseteq \mathcal{A}$ with at most two elements, such that $\mathcal{T} \cup \mathcal{A}_0$ is inconsistent.*

The following lemma relates the concept of conflict given in Definition 3.6 with a standard definition of conflicts.

Lemma 3.7. *\mathcal{C} is a conflict of $\langle \pi\text{-neg}(\mathcal{T}), \mathcal{A} \rangle$ (using Definition 3.6) if and only if it is a minimal inconsistent subset of $\langle \pi\text{-neg}(\mathcal{T}), \mathcal{A} \rangle$.*

Proof of Lemma 3.7. Assume that \mathcal{C} is a conflict in the sense of Definition 3.6. By definition, \mathcal{C} is inconsistent.

- Let us show that \mathcal{C} is minimal. Indeed, let $\mathcal{C} = \{\langle B_1 \sqsubseteq \neg B_2, \alpha_1 \rangle, \langle X, \alpha_2 \rangle, \langle Y, \alpha_3 \rangle\}$. First note that if $B_1 = B_2$ (hence $X = Y$ and $\alpha_2 = \alpha_3$) then clearly $\{\langle B_1 \sqsubseteq \neg B_1, \alpha_1 \rangle\}$ and $\langle X, \alpha_2 \rangle$ are individually consistent. More generally, when $B_1 \neq B_2$, we have $\mathcal{C} \setminus \{\langle B_1 \sqsubseteq \neg B_1, \alpha_1 \rangle\}$ is consistent since it is only composed of assertional facts. Similarly, $\mathcal{C} \setminus \{\langle X, \alpha_2 \rangle\}$ (resp. $\mathcal{C} \setminus \{\langle Y, \alpha_3 \rangle\}$) is also consistent. Indeed if $X = B_1(a)$ (the other cases where $X = P(a, b)$ or $X = P(b, a)$ follow similarly), then it is enough to define a model \mathcal{I} in which $(B_1)^{\mathcal{I}} = \{a\}$ and $(B_2)^{\mathcal{I}} = \emptyset$.
- Now let us show the converse. Namely, assume that \mathcal{C} is a minimal inconsistent subset of $\langle \pi\text{-neg}(\mathcal{T}), \mathcal{A} \rangle$. Since $\pi\text{-neg}(\mathcal{T})$ and \mathcal{A} taken alone are consistent, then \mathcal{C} necessarily contains at least an element of $\pi\text{-neg}(\mathcal{T})$ and at least an element from \mathcal{A} . Besides, from Lemma 3.6, there exists at most two elements from \mathcal{A} in \mathcal{C} . If there exists exactly one element $\langle X, \alpha_2 \rangle$ from $\mathcal{A} \in \mathcal{C}$, then there exists necessarily one negative axiom $\langle B_1 \sqsubseteq \neg B_1, \alpha_1 \rangle$ in $\pi\text{-neg}(\mathcal{T})$ which is inconsistent with $\langle X, \alpha_2 \rangle$. This is a particular case of Definition 3.6 where $B_1 = B_2$, $X = Y$ and $\alpha_2 = \alpha_3$. Now, if there are two elements $\langle X, \alpha_2 \rangle$ and $\langle Y, \alpha_3 \rangle$ from \mathcal{A} in \mathcal{C} then necessarily there exists again one axiom $\langle B_1 \sqsubseteq \neg B_2, \alpha_1 \rangle$ from \mathcal{T} (otherwise \mathcal{C} will not be minimal). We get again the characterization of conflicts given in Definition 3.6. □

A conflict is clearly an inconsistent subset of information. It is minimal (up to a particular case where $B_1 = B_2$). Indeed, removing any element of a conflict restores consistency. A particular case is when $B_1 \sqsubseteq \neg B_1$ belongs to $\pi\text{-neg}(\mathcal{T})$. This corresponds to the situation of an unsatisfiable concept. Namely, there is no way to find an individual that belongs to B_1 . In this case, a conflict is only composed of two elements. A conflict hence involves one negative axiom from $\pi\text{-neg}(\mathcal{T})$ and one or two membership assertions.

Example 3.8. [Example continued] Using Definition 3.6, from $\pi\text{-neg}(\mathcal{T})$ of Example 3.7 and the ABox of Example 3.1, we have the following conflicts:

- $\mathcal{C}_1 = \{\langle \text{NeedBased} \sqsubseteq \neg \text{NeedBased}, \gamma_2 \rangle, \langle \text{NeedBased}(\text{John}), \gamma_9 \rangle\}$ (conflict composed only of two elements).

- $\mathcal{C}_2 = \{ \langle Wandering \sqsubseteq \neg NeedBased, \gamma_5 \rangle, \langle NeedBased(John), \gamma_9 \rangle, \langle Wandering(John), \gamma_4 \rangle \}$
- $\mathcal{C}_3 = \{ \langle \exists satisfiedBy \sqsubseteq \neg NeedBased, \gamma_2 \rangle, \langle satisfiedBy(John, Gifts), \gamma_{11} \rangle, \langle NeedBased(John), \gamma_9 \rangle \}$. ■

The following definition introduces the concepts of the degree of a conflict.

Definition 3.7. Let \mathcal{C} be a conflict. We define the degree of conflict, denoted by $Deg(\mathcal{C})$, as:

$$Deg(\mathcal{C}) = \min(\alpha_1, \alpha_2, \alpha_3),$$

where $(B_1 \sqsubseteq \neg B_2, \alpha_1) \in \mathcal{C}$, $(X, \alpha_2) \in \mathcal{C}$ and $(Y, \alpha_3) \in \mathcal{C}$, and X, Y are defined in Definition 3.6.

Example 3.9. [Example continued] From Example 3.8, the degree of the conflict \mathcal{C}_1 is $Deg(\mathcal{C}_1) = (\gamma_2, \gamma_9) = \gamma_2$, the degree of the conflict \mathcal{C}_2 is $Deg(\mathcal{C}_2) = \min(\gamma_5, \gamma_4, \gamma_9) = \gamma_4$ and the degree of the conflict \mathcal{C}_3 is $Deg(\mathcal{C}_3) = \min(\gamma_2, \gamma_{11}, \gamma_9) = \gamma_2$. ■

We are now ready to give a characterization of $Inc(\mathcal{K})$ using conflicts and their degrees:

Proposition 3.5. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a π -DL-Lite_{core} knowledge base and π -neg(\mathcal{T}) be its negated closure. Then:

$$\begin{aligned} Inc(\mathcal{K}) &= Inc(\langle \pi - neg(\mathcal{T}), \mathcal{A} \rangle) \\ &= \max\{Deg(\mathcal{C}) : \mathcal{C} \text{ is a conflict of } \langle \pi - neg(\mathcal{T}), \mathcal{A} \rangle\} \end{aligned}$$

Proof of Proposition 3.5. Assume that $Inc(\langle \pi - neg(\mathcal{T}), \mathcal{A} \rangle) = \alpha$. This means that:

$$\langle \pi - neg(\mathcal{T}), \mathcal{A} \rangle_{>\alpha} \text{ is consistent}$$

but

$$\langle \pi - neg(\mathcal{T}), \mathcal{A} \rangle_{\geq\alpha} \text{ is inconsistent}$$

This also means that there exists a conflict $\mathcal{C} \in \langle \pi - neg(\mathcal{T}), \mathcal{A} \rangle_{\geq\alpha}$. Indeed, to build a conflict \mathcal{C} from the inconsistent knowledge base $\langle \pi - neg(\mathcal{T}), \mathcal{A} \rangle_{\geq\alpha}$, it is enough to proceed iteratively by removing one element at once until reaching minimal inconsistency. More precisely,

- i) we first let $\mathcal{C} = \langle \pi - neg(\mathcal{T}), \mathcal{A} \rangle_{\geq\alpha}$,
- ii) if there exists $x \in \mathcal{C}$ such that $\mathcal{C} \setminus \{x\}$ is inconsistent, then $\mathcal{C} = \mathcal{C} \setminus \{x\}$,
- iii) repeat step (ii) until there is no x such that $\mathcal{C} \setminus \{x\}$ is inconsistent. \mathcal{C} is then a conflict.

Besides, $Deg(\mathcal{C}) = \alpha$. Otherwise $\langle \pi - neg(\mathcal{T}), \mathcal{A} \rangle_{>\alpha}$ would be inconsistent. Now, the fact that $\langle \pi - neg(\mathcal{T}), \mathcal{A} \rangle_{>\alpha}$ is consistent means that there is no conflict \mathcal{C} that belongs to $\langle \pi - neg(\mathcal{T}), \mathcal{A} \rangle_{>\alpha}$. Hence, $\max\{Deg(\mathcal{C}) : \mathcal{C} \text{ is a conflict of } \langle \pi - neg(\mathcal{T}), \mathcal{A} \rangle\} = \alpha$.

The converse is also straightforward. Indeed, let $\max\{Deg(\mathcal{C}) : \mathcal{C} \text{ is a conflict of } \langle \pi - neg(\mathcal{T}), \mathcal{A} \rangle\} = \alpha$. This means that $\langle \pi - neg(\mathcal{T}), \mathcal{A} \rangle_{>\alpha}$ is consistent. Otherwise there exists a conflict $\mathcal{C} \in \langle \pi - neg(\mathcal{T}), \mathcal{A} \rangle_{>\alpha}$ with $Deg(\mathcal{C}) > \alpha$ (which is impossible).

This also means that $\langle \pi - neg(\mathcal{T}), \mathcal{A} \rangle_{\geq\alpha}$ is inconsistent. Therefore, by definition, $Inc(\langle \pi - neg(\mathcal{T}), \mathcal{A} \rangle) = \alpha$. □

Example 3.10. [Example continued] From Example 3.9, one can easily check that $Inc(\langle \pi - neg(\mathcal{T}), \mathcal{A} \rangle) = \gamma_4$. ■

Proposition 3.5 is important since it provides a natural way to compute $Inc(\langle \pi - neg(\mathcal{T}), \mathcal{A} \rangle)$. A contradiction is present when the same individual belongs to two concepts that compose a negated axiom (i.e. an axiom of $\pi - neg(\mathcal{T})$). The idea in computing the inconsistency degree is to evaluate over \mathcal{A} suitable weighted queries expressed from $\pi - neg(\mathcal{T})$ to exhibit whether the ABox \mathcal{A} contains or not contradictions and to compute the inconsistency degree.

The first idea in the algorithm is to first remove the redundancies from both the TBox and ABox. Then we compute for each π -negated axiom $\langle B_1 \sqsubseteq \neg B_2, \alpha \rangle$ in $\pi - neg(\mathcal{T})$ all pairs of instances from the ABox \mathcal{A} that contradict this π -negated axiom. To this end, we will use FOL-reducibility property of standard *DL-Lite* (see Section 1.4). It consists in evaluating FOL queries over \mathcal{A} stored in a database using for instance an SQL engine in order to detect contradictions. Note that for an efficient evaluation of queries, we remove the redundancies from that ABox. This does not affect the results according to Lemma 3.1.

We will first need to use some standard notations. ψ denotes a translation function that takes as argument a possibilistic negative axiom $\langle B_1 \sqsubseteq \neg B_2, \alpha \rangle$ and produces a weighted FOL query of the form $\langle q, \alpha \rangle$. Note that the semantics of q is similar to the one used in standard *DL-Lite*. Here we simply use the notation $\langle q, \alpha \rangle$ in order to take into account the weight α when computing inconsistency degree. The notation σ_i used below simply represents the degree attached to an assertion $A(x_i)$ in the ABox.

Definition 3.8. ψ is a function that transforms all axioms in $\pi - neg(\mathcal{T})$ to a weighted query $\langle q, \alpha \rangle$:

$$\psi(\langle B_1 \sqsubseteq \neg B_2, \alpha \rangle) = \langle (x, \sigma_1, \sigma_2). \lambda_1(x, \sigma_1) \wedge \lambda_2(x, \sigma_2), \alpha \rangle$$

with λ_i is a translation function from axioms in $\pi - neg(\mathcal{T})$ to FOL formulas, defined as follows:

- $\lambda_i(x, \sigma_i) = A_i(x, \sigma_i)$ if $B_i = A_i$,
- $\lambda_i(x, \sigma_i) = \exists y. P_i(x, y, \sigma_i)$ if $B_i = \exists P_i$,
- $\lambda_i(x, \sigma_i) = \exists y. P_i(y, x, \sigma_i)$ if $B_i = \exists P_i^-$,

where σ_i is the degree to which $A_i(x)$ (resp. $P_i(x, y)$, $P_i(y, x)$) holds in the ABox.

Intuitively, if $\langle B_1 \sqsubseteq \neg B_2, \alpha \rangle$ belongs to $\pi - neg(\mathcal{T})$, then a query associated with $B_1 \sqsubseteq \neg B_2$ is simply denoted by $\psi(\langle B_1 \sqsubseteq \neg B_2, \alpha \rangle)$ and it means return all pairs of assertions $\{\langle X, \sigma_1 \rangle, \langle Y, \sigma_2 \rangle\}$ that are present in the ABox and conflict with $B_1 \sqsubseteq \neg B_2$ where X and Y are given in Definition 3.6. In fact, ψ can be viewed as a simple rewriting of the concept of conflict presented in Definition 3.6.

Example 3.11. [Example continued] From Example 3.7, recall that we have $\pi - neg(\mathcal{T})$. Using ψ defined in Definition 3.8 on $\langle NeedBased \sqsubseteq \neg Loyal, \gamma_2 \rangle$, we obtain $\langle (x, \sigma_1, \sigma_2). NeedBased(x, \sigma_1) \wedge Loyal(x, \sigma_2), \gamma_2 \rangle$. Applying ψ on all axioms in $\pi - neg(\mathcal{T})$ gives the following queries:

- $(q_1) \langle (x, \sigma_1, \sigma_2). NeedBased(x, \sigma_1) \wedge Loyal(x, \sigma_2), \gamma_2 \rangle$
- $(q_2) \langle (x, \sigma_1, \sigma_2). Impulse(x, \sigma_1) \wedge NeedBased(x, \sigma_2), \gamma_{10} \rangle$
- $(q_3) \langle (x, \sigma_1, \sigma_2). Discount(x, \sigma_1) \wedge NeedBased(x, \sigma_2), \gamma_2 \rangle$
- $(q_4) \langle (x, \sigma_1, \sigma_2). NeedBased(x, \sigma_1) \wedge NeedBased(x, \sigma_2), \gamma_2 \rangle$
- $(q_5) \langle (x, \sigma_1, \sigma_2). Wandering(x, \sigma_1) \wedge NeedBased(x, \sigma_2), \gamma_5 \rangle$
- $(q_6) \langle (x, \sigma_1, \sigma_2). \exists y. satisfiedBy(x, y, \sigma_1) \wedge NeedBased(x, \sigma_2), \gamma_2 \rangle$.

One query is associated with each negated query of $\pi\text{-neg}(\mathcal{T})$. For instance $\langle(x, \sigma_1, \sigma_2). \text{NeedBased}(x, \sigma_1) \wedge \text{Loyal}(x, \sigma_2), \gamma_2\rangle$, means compute all pair of membership assertions $\text{NeedBased}(x, \sigma_1)$ and $\text{Loyal}(x, \sigma_2)$ that belong to \mathcal{A} . If \mathcal{A} is implemented using a relational database, this can be easily computed using an SQL query. ■

3.5.3 An algorithm for computing inconsistency degrees

Now, we provide below an algorithm called *Inconsistency*, which takes as input a $\mathcal{K}' = \langle\pi\text{-neg}(\mathcal{T}), \mathcal{A}\rangle$ and computes $\text{Inc}(\mathcal{K}')$, the inconsistency degree of \mathcal{K}' (recall that it is equal to $\text{Inc}(\mathcal{K})$, the inconsistency degree of \mathcal{K}). Algorithm 1 implements main definitions and properties presented in this chapter.

Input: $\mathcal{K}' = \langle\pi\text{-neg}(\mathcal{T}), \mathcal{A}\rangle$

Output: $\text{Inc}(\mathcal{K}')$

```

1: remove redundancies from  $\pi\text{-neg}(\mathcal{T})$  and  $\mathcal{A}$ 
2:  $cont = 0$ 
3: for all  $(\phi_i, \alpha_i) \in \pi\text{-neg}(\mathcal{T}); i = 1..|\pi\text{-neg}(\mathcal{T})|$  do
4:   if  $\alpha_i > cont$  then
5:      $(q, \alpha_q) \leftarrow (\psi(\phi_i, \alpha_i))$ 
6:     if  $\text{Eval}(q, \mathcal{A}) \neq \emptyset$  then
7:        $\beta \leftarrow \min(\alpha_q, \max(\text{Eval}(q, \mathcal{A})))$ 
8:       if  $\beta > cont$  then
9:          $cont \leftarrow \beta$ 
return  $cont$ 

```

Algorithm 1: *Inconsistency*

Algorithm 1 has as input the π -negated closure of \mathcal{T} plus the ABox \mathcal{A} . It has as output the inconsistency degree of the whole $\pi\text{-DL-Lite}$ knowledge base. The variable $cont$ stores the highest inconsistency degree found during the execution of the algorithm. At the beginning, we assume that \mathcal{K}' is consistent. This is the meaning of the initialization Step 2: $cont=0$. Then for each weighted negated axiom (of $\pi\text{-neg}(\mathcal{T})$) we look whether the current inconsistency degree can be increased or not. In line 4, if $\alpha_i \leq cont$ then the inconsistency degree cannot increase. Hence, there is no need to consider conflicts induced by the negated axiom (ϕ_i, α_i) . $\text{Eval}(q, \mathcal{A})$ denotes the evaluation of a weighted query q over \mathcal{A} obtained by transforming an axiom of $\pi\text{-neg}(\mathcal{T})$ with the function given in Definition 3.8. $\text{Eval}(q, \mathcal{A})$ (uses an SQL engine for instance) returns all possibilistic assertions that contradict the query and their corresponding certainty degrees. Note that if $\exists P$ (resp. $\exists P^-$) in one of the atoms of the query q , the function $\text{Eval}(q, \mathcal{A})$ returns all first (resp. second) components of the role P that may be grounded for the query. Next, the function $\max(\text{Eval}(q, \mathcal{A}))$ is used to return the maximal weight, stored in the variable β , of all pairs of assertions that contradict a query q .

This degree represents the inconsistency level of the ABox \mathcal{A} and the asked query q and it is calculated as follows: for each pair of assertions $\langle B_1(a), \alpha_i \rangle$ and $\langle B_2(a), \alpha_j \rangle$ presented in a query result, we only consider one constant having the lowest certainty degree, i.e. $(a, \min(\alpha_j, \alpha_i))$. Note that the use of the \min operator for propagating and aggregating the certainty degrees comes from the property of the conjunction of necessity valued formulas (see Section 3.2). Recall that dropping only one assertion leads to eliminate a conflict (Definition 3.6). The degree β corresponds the highest one among these degrees. In case of consistency, the “if part” of the algorithm (lines 6-9) is never used, and the algorithm returns the value 0 (namely, $\text{Inc}(\mathcal{K})=0$). This explains why $cont$ is initialized to 0 (line 2).

Example 3.12. [Example continued] From queries of Example 3.11 and the ABox of Example 3.1, we have: $\text{Eval}(q_1, \mathcal{A})=\emptyset$, $\text{Eval}(q_2, \mathcal{A})=\emptyset$ and $\text{Eval}(q_3, \mathcal{A})=\emptyset$. Next, we have $\text{Eval}(q_4, \mathcal{A}) \neq \emptyset$ with

$(q_4): \langle (x, \sigma_1, \sigma_2). \text{NeedBased}(x, \sigma_1) \wedge \text{NeedBased}(x, \sigma_2), \gamma_2 \rangle$ and $\langle (a, .\gamma_9, \gamma_9) \rangle$ is the query result. Thus $\beta = \gamma_9$ and $\text{cont} = \max(0, \min(\gamma_9, \gamma_2)) = \gamma_2$. Continuing with q_5 , we have $\text{Eval}(q_5, \mathcal{A}) \neq \emptyset$ where $\langle (a, .\gamma_4, \gamma_9) \rangle$ is the query result. So, $\beta = \gamma_4$ and $\text{cont} = \max(\gamma_2, \min(\gamma_4, \gamma_5)) = \gamma_4$. Lastly, we have $\text{Eval}(q_6, \mathcal{A}) \neq \emptyset$ where $\langle (a, .\gamma_9, \gamma_{11}) \rangle$ is the query result. So, $\beta = \gamma_9$ and $\text{cont} = \max(\gamma_4, \min(\gamma_2, \gamma_9)) = \gamma_4$.

Therefore, the inconsistency degree of the knowledge base is $\text{Inc}(\mathcal{K}) = \gamma_4$. ■

We now provide two propositions that show on one hand that π -DL-Lite extends standard DL-Lite and on the other hand that the computational complexity of Algorithm 1 is the same as the one in standard DL-Lite.

Proposition 3.6. *Let $\mathcal{K}_s = \langle \mathcal{T}_s, \mathcal{A}_s \rangle$ be a standard DL-Lite knowledge base. Let $\mathcal{K}_\pi = \langle \mathcal{T}_\pi, \mathcal{A}_\pi \rangle$ a π -DL-Lite knowledge base where \mathcal{T}_π (resp. \mathcal{A}_π) is defined from \mathcal{T}_s (resp. \mathcal{A}_s) by assigning a degree 1 to each axiom of \mathcal{T}_s (resp. \mathcal{A}_s), namely: $\mathcal{T}_\pi = \{ \langle \phi_i, 1 \rangle : \phi_i \in \mathcal{T}_s \}$ and $\mathcal{A}_\pi = \{ \langle \phi_i, 1 \rangle : \phi_i \in \mathcal{A}_s \}$. Then \mathcal{K}_s is consistent (in the sense of standard DL-Lite) if and only if $\text{Inc}(\mathcal{K}_\pi) = 0$ and \mathcal{K}_s is inconsistent if and only if $\text{Inc}(\mathcal{K}_\pi) = 1$.*

Proposition 3.7. *The complexity of Algorithm 1 is the same as the one used in standard DL-Lite ([Calvanese et al., 2007a], section 3.3, Theorem 26).*

The complexity of reasoning in DL-Lite is recalled in Table 1.7.

Proof of Proposition 3.7. To see why proposition 3.7 holds it is enough to see the differences between Algorithm 1 and the one used in ([Calvanese et al., 2007a], section 3.1.3) for standard DL-Lite. The first remark, concerns the returned result. In our algorithm, results of queries are weighted while in standard DL-Lite, they are not. This does not change the complexity. The difference concerns lines 6-9, where in standard DL-Lite algorithm they are replaced by:

```

1: if  $\text{Eval}(q, \mathcal{A}) \neq \emptyset$  then
2:   return True
3: else
4:   return False

```

At first, in case of consistency both algorithms perform the same steps, because the “if part of the algorithm” is never considered. Now in case of inconsistency, the worst case appears when the whole “loop” is used, namely inconsistency appears with the last element of $\pi\text{-neg}(\mathcal{T})$. In both cases, let \mathcal{A} be the result of the evaluation of $\text{Eval}(q_c, \mathcal{A})$. This needs at least $\mathcal{O}(|\mathcal{A}|)$ steps. Algorithm 1 (contrary to the algorithm in standard DL-Lite [Calvanese et al., 2007a]) computes also $\max\{\alpha_i : \langle \phi_i, \alpha_i \rangle \in \mathcal{A}\}$ which needs again $\mathcal{O}(|\mathcal{A}|)$. Since $\mathcal{O}(2|\mathcal{A}|) = \mathcal{O}(|\mathcal{A}|)$, our algorithm has the same complexity as in standard DL-Lite. Hence we increase the expressive power of DL-Lite while keeping the complexity as low as the one of standard DL-Lite. □

3.6 Possibilistic DL-Lite_F and possibilistic DL-Lite_R

In this section, we first briefly show how to extend the possibilistic DL-Lite_{core} approach to DL-Lite_R and DL-Lite_F, two other important fragments of DL-Lite family. These extensions, denoted by π -DL-Lite_R and π -DL-Lite_F, follow the same steps as π -DL-Lite_{core}.

We first give rules to obtain the negated closure of π -DL-Lite_R and π -DL-Lite_F knowledge bases. These rules extend the ones proposed in Section 3.4 to obtain the negated closure of π -DL-Lite_{core}. We then generalize inconsistency degree checking process for π -DL-Lite_R and π -DL-Lite_F knowledge bases.

Considering \mathcal{L}_R (resp. \mathcal{L}_F) a $DL-Lite_R$ (resp. $DL-Lite_F$) description language, a π - $DL-Lite_R$ (resp. π - $DL-Lite_F$) knowledge base $\mathcal{K} = \{\langle \phi_i, \alpha_i \rangle : 1, \dots, n\}$ is a finite set of possibilistic axioms of the form $\langle \phi, \alpha \rangle$ where ϕ is an axiom expressed in \mathcal{L}_R (resp. \mathcal{L}_F) and $\alpha \in]0, 1]$ is the certainty degree of ϕ . As in π - $DL-Lite_{core}$, the semantics of π - $DL-Lite_R$ (resp. π - $DL-Lite_F$) is given by a possibility distribution, denoted by $\pi_{\mathcal{K}}$, defined over the set of all interpretations $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ of a $DL-Lite_R$ (resp. $DL-Lite_F$) language \mathcal{L}_R (resp. \mathcal{L}_F). This possibility distribution is computed using again Definition 3.2.

3.6.1 π - $DL-Lite_F$ negated closure

Recall that the $DL-Lite_F$ extends $DL-Lite_{core}$ with the ability of specifying functionality on roles or on their inverses of the form:

$$(\text{funct } R)$$

Let us start by defining the negated closure of a π - $DL-Lite_F$ knowledge base. The following rules are added to rules 3.1-3.3 (Section 5.1) to show how to obtain this negated closure of π - $DL-Lite_F$ knowledge base.

Rule 3.4. If $\langle (\text{funct } R), \alpha \rangle \in \mathcal{T}$ then add $\langle (\text{funct } R), \alpha \rangle$ to π -neg(\mathcal{T}).

Rule 3.5. If $\langle \exists R \sqsubseteq \neg \exists R, \alpha \rangle \in \pi$ -neg(\mathcal{T}) then add $\langle \exists R^- \sqsubseteq \neg \exists R^-, \alpha \rangle$ to π -neg(\mathcal{T}).

Rule 3.6. If $\langle \exists R^- \sqsubseteq \neg \exists R^-, \alpha \rangle \in \pi$ -neg(\mathcal{T}) then add $\langle \exists R \sqsubseteq \neg \exists R, \alpha \rangle$ to π -neg(\mathcal{T}).

Once the π - $DL-Lite_F$ negated closure computed, calculating the inconsistency degree of the knowledge base comes down to compute the maximal degree of potential conflicts. A conflict \mathcal{C} in \mathcal{K} is an inconsistent subset of \mathcal{K} , such that no one of subsets of \mathcal{C} is consistent. By adding functionality on roles, new forms of conflict are present in addition to conflicts defined for a π - $DL-Lite_{core}$ knowledge base (Definition 3.6). These new conflicts are of the form:

$$\begin{aligned} & \{ \langle (\text{funct } P), \alpha_1 \rangle, \langle P(a, b), \alpha_2 \rangle, \langle P(a, c), \alpha_3 \rangle \}, \text{ with } b \text{ different from } c. \\ & \{ \langle (\text{funct } P^-), \alpha_1 \rangle, \langle P(b, a), \alpha_2 \rangle, \langle P(c, a), \alpha_3 \rangle \}, \text{ with } b \text{ different from } c. \end{aligned}$$

As said in Section 3.5, the main idea of computing the inconsistency degree is to evaluate over the ABox queries obtained from the π -negated closure in order to determine the maximal degree of conflicting elements. To obtain queries from a π - $DL-Lite_F$ knowledge base, we extend Definition 3.8 as follows:

$$\psi(\langle (\text{funct } P), \alpha \rangle) = \langle (x, \sigma_1, \sigma_2). \exists y. \exists z. P(x, y, \sigma_1) \wedge P(x, z, \sigma_2) \wedge y \neq z, \alpha \rangle$$

$$\psi(\langle (\text{funct } P^-), \alpha \rangle) = \langle (x, \sigma_1, \sigma_2). \exists y. \exists z. P(y, x, \sigma_1) \wedge P(z, x, \sigma_2) \wedge y \neq z, \alpha \rangle$$

Where σ_i is the degree to which $R(x, y)$ and $R(x, z)$ hold in the ABox.

Example 3.13. [Example continued] Let us extend the TBox of Example 3.1 with the following axiom $\langle (\text{funct SatisfiedBy}), \gamma_{14} \rangle$ and the ABox with the following axiom $\langle \text{SatisfiedBy}(\text{john}, \text{discount}), \gamma_{15} \rangle$. One can check that $\{ \langle (\text{funct SatisfiedBy}), \gamma_{14} \rangle, \langle \text{SatisfiedBy}(\text{john}, \text{discount}), \gamma_{15} \rangle, \langle \text{SatisfiedBy}(\text{john}, \text{gifts}), \gamma_{11} \rangle \}$ is a conflict set. ■

3.6.2 π -DL-Lite_R negated closure

The DL-Lite_R extends DL-Lite_{core} with the ability of specifying inclusion axioms between roles of the form:

$$R \sqsubseteq E$$

We now show how to obtain the negated closure of a π -DL-Lite_R knowledge base. In addition to Rules 3.1-3.3 proposed in section 3.5 for π -DL-Lite_{core} knowledge bases, the following rules should be added for π -DL-Lite_R.

Rule 3.7. If $\langle R_1 \sqsubseteq R_2, \alpha_1 \rangle \in \mathcal{T}$ and $\langle \exists R_2 \sqsubseteq \neg B, \alpha_2 \rangle \in \pi\text{-neg}(\mathcal{T})$ or $\langle B \sqsubseteq \neg \exists R_2, \alpha_2 \rangle \in \pi\text{-neg}(\mathcal{T})$ then add $\langle \exists R_1 \sqsubseteq \neg B, \min(\alpha_1, \alpha_2) \rangle$ to $\pi\text{-neg}(\mathcal{T})$.

Rule 3.8. If $\langle R_1 \sqsubseteq R_2, \alpha_1 \rangle \in \mathcal{T}$ and $\langle \exists R_2^- \sqsubseteq \neg B, \alpha_2 \rangle \in \pi\text{-neg}(\mathcal{T})$ or $\langle B \sqsubseteq \neg \exists R_2^-, \alpha_2 \rangle \in \pi\text{-neg}(\mathcal{T})$ then add $\langle \exists R_1^- \sqsubseteq \neg B, \min(\alpha_1, \alpha_2) \rangle$ to $\pi\text{-neg}(\mathcal{T})$.

Rule 3.9. If $\langle R_1 \sqsubseteq R_2, \alpha_1 \rangle \in \mathcal{T}$ and $\langle R_2 \sqsubseteq \neg R_3, \alpha_2 \rangle \in \pi\text{-neg}(\mathcal{T})$ or $\langle R_3 \sqsubseteq \neg R_2, \alpha_2 \rangle \in \pi\text{-neg}(\mathcal{T})$ then add $\langle R_1 \sqsubseteq \neg R_3, \min(\alpha_1, \alpha_2) \rangle$ to $\pi\text{-neg}(\mathcal{T})$.

Rule 3.10. if $\langle R \sqsubseteq \neg R, \alpha \rangle \in \pi\text{-neg}(\mathcal{T})$ or $\langle \exists R \sqsubseteq \neg \exists R, \alpha \rangle \in \pi\text{-neg}(\mathcal{T})$ or $\langle \exists R^- \sqsubseteq \neg \exists R^-, \alpha \rangle \in \pi\text{-neg}(\mathcal{T})$ then add $\langle R \sqsubseteq \neg R, \alpha \rangle$ and $\langle \exists R \sqsubseteq \neg \exists R, \alpha \rangle$ and $\langle \exists R^- \sqsubseteq \neg \exists R^-, \alpha \rangle$ to $\pi\text{-neg}(\mathcal{T})$.

Given this set of rules and syntax of a π -DL-Lite_R knowledge base, a new form of conflicts may be generated. Namely,

$$\{\langle R_1 \sqsubseteq \neg R_2, \alpha_1 \rangle, \langle R_1(a, b), \alpha_2 \rangle, \langle R_2(a, b), \alpha_3 \rangle\}$$

To this end, Definition 3.8 needs to be extended in order to obtain weighted queries used to exhibit contradictions in the ABox and compute the the inconsistency degree of a π -DL-Lite_R knowledge base as follows:

$$\psi(\langle R_1 \sqsubseteq \neg R_2, \alpha \rangle) = \langle (x, y, \sigma_1, \sigma_2). \nu_1(x, y, \sigma_1) \wedge \nu_2(x, y, \sigma_2), \alpha \rangle$$

with

- $\nu_i(x, y, \sigma_i) = P_i(x, y, \sigma_i)$ if $R_i = P_i$
- $\nu_i(x, y, \sigma_i) = P_i(y, x, \sigma_i)$ if $R_i = P_i^-$

Where σ_i is the degree to which $R(x, y)$ holds in the ABox.

Example 3.14. [Examples 3.1 and 3.7 continued] Let us extend the TBox with the following axiom $\langle \text{AppropriateFor}^- \sqsubseteq \text{SatisfiedBy}, \gamma_{14} \rangle$ and the ABox with the following axiom $\langle \text{AppropriateFor}(\text{Gifts}, \text{John}), \gamma_{15} \rangle$. Using rule 3.7, one can generate the following negated axiom: $\langle \text{AppropriateFor}^- \sqsubseteq \neg \text{NeedBased}, \gamma_{14} \rangle$. One can check that $\{\langle \text{AppropriateFor}^- \sqsubseteq \neg \text{NeedBased}, \gamma_{14} \rangle, \langle \text{AppropriateFor}(\text{Gifts}, \text{John}), \gamma_{15} \rangle, \langle \text{Needbased}(\text{John}, \gamma_9) \rangle\}$ is a conflict. ■

3.7 Basic inferences in π -DL-Lite_{core}

In this section, we present standard DL-Lite_{core} inference services (*i.e.* subsumption and instance checking) within a possibility theory setting. We show how to compute possibilistic inferences given in Definition 3.4 and Definition 3.5 when ϕ is either a membership assertion (*i.e.* a fact) or a subsumption relation (*i.e.* a TBox axiom).

In π -DL-Lite_{core}, we define two different kinds of inference services, namely flat subsumption (*resp.* instance checking) and weighted subsumption (*resp.* instance checking). The main difference between flat subsumption (*resp.* instance checking) and weighted subsumption (*resp.* instance checking) is that in the first case, we only check whether the subsumption (*resp.* instance checking) holds whatever is the degree, while in the second case, the subsumption (*resp.* instance checking) should be satisfied to a maximal degree.

In what follows, we detail these two types of inferences and we start by studying flat inference.

Proposition 3.8 (Flat subsumption). *Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a π -DL-Lite_{core} knowledge base (such that $\text{Inc}(\mathcal{K}) < 1$), C_1 and C_2 be two general concepts, X be an atomic concept not appearing in \mathcal{T} and x be a constant not appearing in \mathcal{A} . Then, $\mathcal{K} \models_{C_1} \sqsubseteq C_2$ if and only if the knowledge base $\mathcal{K}_1 = \langle \mathcal{T}_1, \mathcal{A}_1 \rangle$ is inconsistent to some degree ($\exists \alpha > 0$ such that $\text{Inc}(\mathcal{K}_1) = \alpha$) where*

$$\mathcal{T}_1 = \mathcal{T}_{>\text{Inc}(\mathcal{K})} \cup \{ \langle X \sqsubseteq C_1, 1 \rangle, \langle X \sqsubseteq \neg C_2, 1 \rangle \} \text{ and } \mathcal{A}_1 = \{ \langle X(x), 1 \rangle \}$$

Proof of Proposition 3.8. Let us assume that $\mathcal{K} \models_{C_1} \sqsubseteq C_2$. By Definition 3.4, this means that:

$$N(C_1 \sqsubseteq C_2) > \text{Inc}(\mathcal{K}).$$

By definition of necessity measures, this leads to:

$$\begin{aligned} N(C_1 \sqsubseteq C_2) &= 1 - \max_{\mathcal{I} \in \Omega} \{ \pi(\mathcal{I}) : \mathcal{I} \not\models_{C_1} \sqsubseteq C_2 \} > \text{Inc}(\mathcal{K}) \\ &= \max_{\mathcal{I} \in \Omega} \{ \pi(\mathcal{I}) : \mathcal{I} \not\models_{C_1} \sqsubseteq C_2 \} < 1 - \text{Inc}(\mathcal{K}). \end{aligned}$$

which means that:

$$\forall \mathcal{I} \in \Omega \text{ such that } \mathcal{I} \not\models_{C_1} \sqsubseteq C_2, \text{ we have } \pi(\mathcal{I}) < 1 - \text{Inc}(\mathcal{K}). \quad (1)$$

Besides, it is easy to show that:

$$\mathcal{I} \not\models_{\mathcal{T}_{>\text{Inc}(\mathcal{K})}} \text{ if and only if } \pi(\mathcal{I}) = 1 - \text{Inc}(\mathcal{K}). \quad (2)$$

From (1) and (2), we conclude that:

$$\forall \mathcal{I} \in \Omega \text{ such that } \mathcal{I} \models_{\mathcal{T}_{>\text{Inc}(\mathcal{K})}}, \text{ we have } \mathcal{I} \models_{C_1} \sqsubseteq C_2. \quad (3)$$

From (3), it is impossible to satisfy $\mathcal{T}_{>\text{Inc}(\mathcal{K})}$ and formulas $\{X \sqsubseteq C_1, X \sqsubseteq \neg C_2, X(x)\}$ (since models of $\mathcal{T}_{>\text{Inc}(\mathcal{K})}$ satisfy $C_1 \sqsubseteq C_2$) which means that \mathcal{K}_1 is inconsistent.

Conversely, let us assume that $\mathcal{K}_1 = \langle \mathcal{T}_1, \mathcal{A}_1 \rangle$ is inconsistent. Then, there exists an $\alpha > 0$ such that $\text{Inc}(\mathcal{K}_1) = \alpha$. This means that:

$$\mathcal{K}_{1 \geq \alpha} \text{ is inconsistent,}$$

and

$\mathcal{K}_{1>\alpha}$ is consistent.

Besides, it is easy to see that:

$$\mathcal{K}_{1>\alpha} \models_{\pi} \{ \langle X \sqsubseteq C_1, 1 \rangle, \langle X \sqsubseteq \neg C_2, 1 \rangle, \langle X(x), 1 \rangle \}. \quad (4)$$

This means that:

$$\mathcal{K}_{1>\alpha} \not\models C_1 \sqsubseteq C_2 \quad (5)$$

From (4) and (5), one can easily show that there exists an interpretation \mathcal{I} such that:

$$\pi(\mathcal{I}) < 1 - \alpha \text{ and } \mathcal{I} \not\models C_1 \sqsubseteq C_2.$$

Since all axioms of \mathcal{K}_1 have weights greater than $Inc(\mathcal{K})$ then:

$$\pi(\mathcal{I}) < 1 - \alpha \leq 1 - Inc(\mathcal{K}) \quad (6)$$

So, from (6), we have $N(C_1 \sqsubseteq C_2) > Inc(\mathcal{K})$. Therefore by Definition 3.4, $\mathcal{K} \models_{\pi} C_1 \sqsubseteq C_2$. \square

Example 3.15. [Example continued] Let us check if $\mathcal{K} \models_{\pi} NeedBased \sqsubseteq Loyal$. From Example 3.12, we have $Inc(\mathcal{K}) = \gamma_4$ then $\mathcal{T}_{>\gamma_4}$ is as follows:

$$\begin{array}{ll} \langle Discount \sqsubseteq Loyal, \gamma_6 \rangle & \langle NeedBased \sqsubseteq Discount, \gamma_8 \rangle \\ \langle Impulse \sqsubseteq \neg NeedBased, \gamma_{10} \rangle & \langle Wandering \sqsubseteq Impulse, \gamma_5 \rangle \\ \langle \exists satisfiedBy^- \sqsubseteq FidelityMethod, 1 \rangle & \langle \exists satisfiedBy \sqsubseteq Loyal, 1 \rangle \end{array}$$

By adding the intermediary concept X , the knowledge base $\mathcal{K}_1 = \langle \mathcal{T}_1, \mathcal{A}_1 \rangle$ is as follow:

$$\mathcal{T}_1 = \mathcal{T}_{>\gamma_4} \cup \{ \langle X \sqsubseteq NeedBased, 1 \rangle, \langle X \sqsubseteq \neg Loyal, 1 \rangle \} \text{ and } \mathcal{A}_1 = \{ \langle X(x), 1 \rangle \}$$

Computing now $\pi\text{-neg}(\mathcal{T}_1)$, we obtain the following negative inclusion axioms:

$$\begin{array}{ll} \langle Impulse \sqsubseteq \neg NeedBased, \gamma_{10} \rangle & \langle NeedBased \sqsubseteq \neg X, \gamma_6 \rangle \\ \langle Wandering \sqsubseteq \neg NeedBased, \gamma_5 \rangle & \langle X \sqsubseteq \neg X, \gamma_6 \rangle \\ \langle X \sqsubseteq \neg Loyal, 1 \rangle & \langle \exists satisfiedBy \sqsubseteq \neg X, 1 \rangle \\ \langle Discount \sqsubseteq \neg X, \gamma_6 \rangle & \end{array}$$

One can easily check that $X \sqsubseteq \neg X$ conflicts with $X(x)$. Then, we conclude that \mathcal{K}_1 is inconsistent. Therefore $\mathcal{K} \not\models_{\pi} NeedBased \sqsubseteq Loyal$. \blacksquare

Proposition 3.9 (Flat instance checking). *Let \mathcal{K} be a π -DL-Lite_{core} knowledge base (such that $Inc(\mathcal{K}) < 1$), C be a concept, X be an atomic concept not appearing in \mathcal{T} and x be a constant appearing in \mathcal{A} . Then, $\mathcal{K} \models C(x)$ if and only if the knowledge base $\mathcal{K}_1 = \langle \mathcal{T}_1, \mathcal{A}_1 \rangle$ is inconsistent to some degree where*

$$\mathcal{T}_1 = \mathcal{T}_{>Inc(\mathcal{K})} \cup \{ \langle X \sqsubseteq \neg C, 1 \rangle \} \text{ and } \mathcal{A}_1 = \mathcal{A}_{>Inc(\mathcal{K})} \cup \{ \langle X(x), 1 \rangle \}$$

Proof of Proposition 3.9. The proof is basically the same as the one of Proposition 3.8. \square

Example 3.16. [Example continued] Let us check if $\mathcal{K} \models \text{Loyal}(\text{John})$. Consider the TBox $\mathcal{T}_{>\gamma_4}$ of Example 3.15 and the following ABox $\mathcal{A}_{>\gamma_4}$ where $\text{Inc}(\mathcal{K}) = \gamma_4$:

$$\langle \text{NeedBased}(\text{John}), \gamma_9 \rangle \qquad \langle \text{satisfiedBy}(\text{John}, \text{Gifts}), \gamma_{11} \rangle$$

By adding the intermediary concept X , $\mathcal{K}_1 = \langle \mathcal{T}_1, \mathcal{A}_1 \rangle$ is as follow:

$$\mathcal{T}_1 = \mathcal{T}_{>\gamma_4} \cup \{ \langle X \sqsubseteq \neg \text{Loyal}, 1 \rangle \} \quad \text{and} \quad \mathcal{A}_1 = \mathcal{A}_{>\gamma_4} \cup \{ \langle X(\text{John}), 1 \rangle \}$$

Computing now $\pi\text{-neg}(\mathcal{T}_1)$, we obtain the following negative inclusion axioms:

$$\begin{array}{ll} \langle \text{Impulse} \sqsubseteq \neg \text{NeedBased}, \gamma_{10} \rangle & \langle \text{Discount} \sqsubseteq \neg X, \gamma_6 \rangle \\ \langle \text{Wandering} \sqsubseteq \neg \text{NeedBased}, \gamma_5 \rangle & \langle \text{NeedBased} \sqsubseteq \neg X, \gamma_6 \rangle \\ \langle X \sqsubseteq \neg \text{Loyal}, 1 \rangle & \langle \exists \text{satisfiedBy} \sqsubseteq \neg X, 1 \rangle \end{array}$$

One can easily check that $\langle \text{NeedBased} \sqsubseteq \neg X, \gamma_6 \rangle$ conflicts with $\langle X(\text{John}), 1 \rangle$ and $\langle \text{NeedBased}(\text{John}), \gamma_9 \rangle$ and $\langle \exists \text{satisfiedBy} \sqsubseteq \neg X, 1 \rangle$ conflict with $\langle X(\text{John}), 1 \rangle$ and $\langle \text{satisfiedBy}(\text{John}, \text{Gifts}), \gamma_{11} \rangle$. Then, we conclude that \mathcal{K}_1 is inconsistent. Therefore $\mathcal{K} \models \text{Loyal}(\text{John})$. ■

As we can see, flat inference is done in a similar way than inference in standard $DL\text{-Lite}_{core}$ and it permits to cope with inconsistency. However, the second type of inference (i.e. weighted inference) is stronger than flat inference and it deals with uncertainty by determining to what extent an inference task can be done from a π -DL-Lite_{core} knowledge base.

Proposition 3.10 (Weighted subsumption). *Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a π -DL-Lite_{core} knowledge base, C_1 and C_2 be two general concepts, X be an atomic concept not appearing in \mathcal{T} , and x be a constant not appearing in \mathcal{A} . Then, $\mathcal{K} \models_{\pi} \langle C_1 \sqsubseteq C_2, \alpha \rangle$ if and only if the $\text{Inc}(\mathcal{K}_1) = \alpha > \text{Inc}(\mathcal{K})$ where $\mathcal{K}_1 = \langle \mathcal{T}_1, \mathcal{A}_1 \rangle$ with*

$$\mathcal{T}_1 = \mathcal{T}_{\geq \alpha} \cup \{ \langle X \sqsubseteq C_1, 1 \rangle, \langle X \sqsubseteq \neg C_2, 1 \rangle \} \quad \text{and} \quad \mathcal{A}_1 = \{ \langle X(x), 1 \rangle \}$$

Proof of Proposition 3.10. The proof of Proposition 3.10 is similar to the one of Proposition 3.8. Let us assume that $\mathcal{K} \models_{\pi} \langle C_1 \sqsubseteq C_2, \alpha \rangle$. From Definition 3.5, we have

$$N(C_1 \sqsubseteq C_2) = \alpha > \text{Inc}(\mathcal{K})$$

By definition of necessity measure, this leads to:

$$\begin{aligned} N(C_1 \sqsubseteq C_2) &= 1 - \max_{\mathcal{I} \in \Omega} \{ \pi(\mathcal{I}) : \mathcal{I} \not\models C_1 \sqsubseteq C_2 \} = \alpha > \text{Inc}(\mathcal{K}), \\ &= \max_{\mathcal{I} \in \Omega} \{ \pi(\mathcal{I}) : \mathcal{I} \not\models C_1 \sqsubseteq C_2 \} = 1 - \alpha < 1 - \text{Inc}(\mathcal{K}). \end{aligned}$$

which means that:

$$\forall \mathcal{I} \in \Omega \text{ such that } \mathcal{I} \not\models C_1 \sqsubseteq C_2, \text{ we have } \pi_{\mathcal{K}}(\mathcal{I}) \leq 1 - \alpha < 1 - \text{Inc}(\mathcal{K}) \quad (1)$$

Note that $\alpha > \text{Inc}(\mathcal{K})$. This means that $\mathcal{T}_{\geq \alpha}$ is consistent, and

$$\forall \mathcal{I}' \in \Omega \text{ such that } \mathcal{I}' \models \mathcal{T}_{\geq \alpha}, \text{ we have } \pi_{\mathcal{K}}(\mathcal{I}') \leq 1 - \alpha \quad (2)$$

From (1) and (2), we conclude that:

$$\forall \mathcal{I}' \in \Omega \text{ such that } \mathcal{I}' \models \mathcal{T}_{\geq \alpha}, \text{ it implies } \mathcal{I}' \models_{\pi} \langle C_1 \sqsubseteq C_2, \alpha \rangle \text{ and } \\ \mathcal{I}' \not\models_{\pi} \{ \langle X \sqsubseteq C_1, 1 \rangle, \langle X \sqsubseteq \neg C_2, 1 \rangle, \langle X(x), 1 \rangle \}.$$

This also means that \mathcal{K}_1 is inconsistent. Hence $Inc(\mathcal{K}_1) \geq \alpha > Inc(\mathcal{K})$. Since $\mathcal{T}_{\geq \alpha} \cup \{ \langle X \sqsubseteq C_1, 1 \rangle, \langle X \sqsubseteq \neg C_2, 1 \rangle, \langle X(x), 1 \rangle \}$ is inconsistent. Then $Inc(\mathcal{K}_1) = \max_{\mathcal{I} \in \Omega} \{ \pi_{\mathcal{K}}(\mathcal{I}) \} \leq 1 - \alpha$ (since all formulas of \mathcal{K}_1 have a weight greater than α).

Let us now show that $Inc(\mathcal{K}_1) = \alpha$. It is enough to show that:

$$\mathcal{T}_{> \alpha} \cup \{ \langle X \sqsubseteq C_1, 1 \rangle, \langle X \sqsubseteq \neg C_2, 1 \rangle, \langle X(x), 1 \rangle \} \text{ is consistent}$$

Assume that it is inconsistent. This means that $\forall \mathcal{I} \in \Omega$, if $\mathcal{I} \models \mathcal{T}_{> \alpha}$ then $\mathcal{I} \not\models \{ X \sqsubseteq C_1, X \sqsubseteq \neg C_2, X(x) \}$ and $\mathcal{I} \models C_1 \sqsubseteq C_2$. Recall that $\mathcal{T}_{> \alpha}$ is consistent. This means that $\max \{ \mathcal{I}' : \mathcal{I}' \not\models C_1 \sqsubseteq C_2 \} < 1 - \alpha$ which contradicts $N(C_1 \sqsubseteq C_2) = \alpha$. Hence $\mathcal{K}_{1>\alpha}$ is consistent. Therefore $Inc(\mathcal{K}_1) = \alpha > Inc(\mathcal{K})$.

Conversely, let us assume that \mathcal{K}_1 is inconsistent and $Inc(\mathcal{K}_1) = \alpha > Inc(\mathcal{K})$. This means that:

$$\mathcal{K}_{1>\alpha} \text{ is consistent, (3)}$$

and

$$\mathcal{K}_{1 \geq \alpha} \text{ is inconsistent. (4)}$$

From (3) we have:

$$\mathcal{K}_{1>\alpha} \models_{\pi} \{ \langle X \sqsubseteq C_1, 1 \rangle, \langle X \sqsubseteq \neg C_2, 1 \rangle, \langle X(x), 1 \rangle \}$$

Since trivially $\{ \langle X \sqsubseteq C_1, 1 \rangle, \langle X \sqsubseteq \neg C_2, 1 \rangle, \langle X(x), 1 \rangle \} \subseteq \mathcal{K}_{1>\alpha}$

This means that:

$$\mathcal{K}_{1>\alpha} \not\models_{\pi} \langle C_1 \sqsubseteq C_2, \alpha \rangle \text{ (5)}$$

From (4), we have

$$\mathcal{T}_{\geq \alpha} \cup \{ \langle X \sqsubseteq C_1, 1 \rangle, \langle X \sqsubseteq \neg C_2, 1 \rangle, \langle X(x), 1 \rangle \} \text{ is inconsistent.}$$

Since $Inc(\mathcal{K}_1) \geq \alpha > Inc(\mathcal{K})$, this means that $\mathcal{T}_{\geq \alpha}$ is consistent and

$$\forall \mathcal{I}' \text{ such that } \mathcal{I}' \models \mathcal{T}_{\geq \alpha}, \text{ we have } \pi_{\mathcal{K}}(\mathcal{I}') \leq 1 - \alpha \text{ (6)}$$

$$\forall \mathcal{I}' \text{ such that } \mathcal{I}' \models \mathcal{T}_{\geq \alpha} \text{ implies } \mathcal{I}' \models_{\pi} \langle C_1 \sqsubseteq C_2, \alpha \rangle \text{ but } \\ \mathcal{I}' \not\models_{\pi} \{ \langle X \sqsubseteq C_1, 1 \rangle, \langle X \sqsubseteq \neg C_2, 1 \rangle, \langle X(x), 1 \rangle \} \text{ (7)}$$

From (5), (6) and (7), one can show that there exists an interpretation \mathcal{I} such that:

$$\pi(\mathcal{I}) \leq 1 - \alpha \text{ and } \mathcal{I} \not\models C_1 \sqsubseteq C_2. \text{ (8)}$$

From (8), we conclude that $N(C_1 \sqsubseteq C_2) = \alpha$. Therefore by Definition 3.5, $\mathcal{K} \models_{\pi} \langle C_1 \sqsubseteq C_2, \alpha \rangle$. \square

Example 3.17. [Example continued] From Example 3.15, one can easily check that $Inc(\mathcal{K}_1) = \gamma_6$ then $\mathcal{K} \models_{\pi} \langle NeedBased \sqsubseteq Loyal, \gamma_6 \rangle$. \blacksquare

Proposition 3.11 (Weighted instance checking). *Let \mathcal{K} be a π -DL-Lite_{core} knowledge base, C be a concept, X be an atomic concept not appearing in \mathcal{T} and x be a constant not appearing in \mathcal{A} . Then, $\mathcal{K} \models_{\pi} \langle C(x), \alpha \rangle$ if and only if $Inc(\mathcal{K}_1) = \alpha > Inc(\mathcal{K})$ where $\mathcal{K}_1 = \langle \mathcal{T}_1, \mathcal{A}_1 \rangle$ with*

$$\mathcal{T}_1 = \mathcal{T}_{\geq \alpha} \cup \{ \langle X \sqsubseteq \neg C, 1 \rangle \} \quad \text{and} \quad \mathcal{A}_1 = \mathcal{A}_{\geq \alpha} \cup \{ \langle X(x), 1 \rangle \}$$

Proof of 3.11. The same proof as the proof of Proposition 3.10. □

Example 3.18. [Example continued] From Example 3.16, one can easily check that $Inc(\mathcal{K}_1) = \gamma_6$ then $\mathcal{K} \models_{\pi} \langle Loyal(John), \gamma_6 \rangle$. ■

Hence from Proposition 3.8-3.11, we deduce that flat subsumption (*resp.* instance checking) and weighted subsumption (*resp.* instance checking) can be obtained using Algorithm 1 presented above. The above results show that the complexity of flat subsumption (*resp.* instance checking) and weighted subsumption (*resp.* instance checking) in π -DL-Lite is the same as the one used in standard DL-Lite.

3.8 Query answering in possibilistic DL-Lite

In this section, we briefly address the problem of query answering within π -DL-Lite framework. The problem of standard query answering is closely related to the ontology-based data access problem which takes a set of assertions (*i.e.* an ABox), an ontology (*i.e.* a TBox) and a conjunctive query q and aims to find if there exists an answer or find all the answers to q over the set of data. In such a setting, an ontology acts as a schema used to enrich the query. The problem of query answering within DL-Lite setting has been mainly studied in [Calvanese *et al.*, 2007a]. Query answering process comes down first to the reformulation of the query q over the TBox in order to enrich it while eliminating all the redundancies and then evaluate the new obtained queries over the ABox.

We now briefly present query answering process over the π -DL-Lite setting. This procedure follows similar steps as in [Calvanese *et al.*, 2007a; Straccia, 2006a; Straccia, 2012]. It consists in:

1. Query reformulation: given a query q over $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ a π -DL-Lite knowledge base, we first use the positive axioms of $\mathcal{T}_{>Inc(\mathcal{K})}$ to enrich the query q . This leads at each possible application of a positive axiom to obtain a new query q' . The resulting set of queries is then used under the ABox (stored as a database). Let us denote by $Q = ref(q, \mathcal{T}_{>Inc(\mathcal{K})})$ the set of queries obtained by reformulating q over $\mathcal{T}_{>Inc(\mathcal{K})}$.
2. Query evaluation: given $ref(q, \mathcal{T}_{>Inc(\mathcal{K})})$, we first evaluate over $\mathcal{A}_{>Inc(\mathcal{K})}$ each q' in $ref(q, \mathcal{T}_{>Inc(\mathcal{K})})$ while taking the most certain answers. Let us denote by $ans_{\pi}(\mathcal{K}, q)$ the certain answers of the query q over $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ a π -DL-Lite knowledge base.

In DL-Lite, the evaluation of a Conjunctive Query (CQ) uses the notion of FOL-reducibility and it is based on a method that separates the TBox and the ABox. Namely, we use positive axioms of the TBox for reformulating a CQ on the Union of Conjunctive Query (UCQ) to be evaluated over the ABox (may be represented by a relational database). Given a CQ q , we recall the query reformulation algorithm PerfectRef proposed in [Calvanese *et al.*, 2007a] that reformulates q taking into account only positive axioms of a π -DL-Lite TBox \mathcal{T} having weights strictly greater than the inconsistency degree.

Input: $\mathcal{T}_{>\alpha}$ where $\alpha = Inc(\mathcal{K})$ and q a CQ

Output: Q query reformulation

```

1:  $Q_1 \leftarrow \{q\}$  // set of queries
2: repeat
3:    $Q_2 \leftarrow Q_1$ 
4:   for all  $q \in Q_2$  do
5:     for all  $g$  in  $q$  do
6:       for all  $PI_i \in \mathcal{T}_{\geq\alpha}$  do
7:         if  $PI$  is applicable to  $g$  then
8:            $Q_1 \leftarrow Q_1 \cup \{q[g/gr(g, PI)]\}$ 
9:   for all  $g_1, g_2$  in  $q$  do
10:    if  $g_1$  and  $g_2$  unify then
11:       $Q_1 \leftarrow Q_1 \cup \{\tau[reduce(q, g_1, g_2)]\}$ 
12: until  $Q_2 \leftarrow Q_1$  return ( $Q_2$ )

```

Algorithm 2: $PerfectRef(q, \mathcal{T}_{>\alpha})$

In this algorithm, $gr(g, PI)$ denotes the result of applying a positive axiom to an atom of the query. Let us use the symbol "_" to denote non-distinguished non-shared variables (i.e. appeared only once in the query), the function $gr(g, PI)$ is described as follows: A positive axiom is applicable to a factual concept $A(x)$, if the positive inclusion has the concept A in its right-hand side. Similarly a positive inclusion is applicable to a factual relation $P(x_1, x_2)$, if either (i) $x_1 = _$ and $\exists P$ is in its right-hand side, or (ii) $x_2 = _$ and $\exists P^-$ is in its right-hand side, (iii) positive inclusion is a role inclusion and on its right-hand side is P or P^- .

Furthermore $q[g/gr(g, PI)]$ denotes the CQ obtained from q by replacing the atom g with a new atom $gr(g, PI)$. In addition $reduce$ is a function that takes as input a CQ q and two atoms g_1 and g_2 and returns a CQ q' obtained by applying to q the most general unifier between g_1 and g_2 . Finally τ is a function that takes as input a CQ q and returns a new CQ obtained by replacing each occurrence of an unbound variable in q with the symbol "_". Note that $gr(g, PI)$, $q[g/gr(g, PI)]$, τ and $reduce$ are the same used in [Calvanese et al., 2007a]. For more details see [Calvanese et al., 2007a].

Example 3.19. [Examples 3.1 continued] Let us consider the following TBox with $\mathcal{T}_{>Inc(\mathcal{K})}$.

$\langle Discount \sqsubseteq Loyal, \gamma_6 \rangle$	$\langle NeedBased \sqsubseteq Discount, \gamma_8 \rangle$
$\langle Impulse \sqsubseteq \neg NeedBased, \gamma_{10} \rangle$	$\langle Wandering \sqsubseteq Impulse, \gamma_5 \rangle$
$\langle \exists satisfiedBy^- \sqsubseteq FidelityService, 1 \rangle$	$\langle \exists satisfiedBy \sqsubseteq Loyal, 1 \rangle$

Let us consider the following conjunctive query q :

$$q(x) \leftarrow \exists y. satisfiedBy(x, y) \wedge FidelityService(y)$$

At the first execution of algorithm $PerfectRef(q, \mathcal{T}_{>\alpha})$ the following query is added in Q_1 :

$$q_1(x) \leftarrow \exists y. satisfiedBy(x, y) \wedge satisfiedBy(_, y)$$

since $\exists satisfiedBy^- \sqsubseteq FidelityService$ is applicable to the atom $FidelityService(y)$. Now, one can see that the two atoms of the added query unify, then the algorithm inserts the following query:

$$q_2(x) \leftarrow satisfiedBy(x, _)$$

The above two queries and the original one are returned by the algorithm as the set of queries obtained by reformulating q over $\mathcal{T}_{>Inc(\mathcal{K})}$. ■

We now explain the main ideas behind query answering in π -DL-Lite framework. Let $q(\vec{x}) \leftarrow \exists \vec{y}.conj(\vec{x}, \vec{y})$ be a conjunctive query with answer variable $\vec{x} = x_1, \dots, x_n$ and arity n . Given an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, recall that in standard DL-Lite, $q^{\mathcal{I}}$ is interpreted as a set of tuples $\vec{a} \in (\Delta)^n$ that belong to the domain of interpretation such that if we substitute \vec{x} by the constants \vec{a} the query q will be evaluated as true under \mathcal{I} .

In the spirit of instance checking (presented in Section 3.7), when certainty degrees are available over the set of constants in \mathcal{A} , query answering process comes down to search the most certain answers for the query q and to compute to what extent the answer of this query holds. As mentioned at the beginning of this section, the first step of query answering is the query reformulation. This step leads to obtain a set of queries where the union of the answer sets of these queries will be the answer to the original query. Before given the way to deal with $Q = ref(q, \mathcal{T}_{>Inc(\mathcal{K})})$ the set of queries obtained by reformulating q over $\mathcal{T}_{>Inc(\mathcal{K})}$, let us first show how to deal with CQ over π -DL-Lite setting in order to take into account the weights attached to assertions in the ABox.

Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be a DL-Lite interpretation and $q(\vec{x}) \leftarrow \exists \vec{y}.conj(\vec{x}, \vec{y})$ be a CQ. Let us first consider the case where the query does not involve existential variables, namely q is of the form $q(\vec{x}) \leftarrow \bigwedge A_i(\vec{x})$ where A_i are atoms. Let $\vec{a} \in (\Delta^{\mathcal{I}})^n$ be a tuple of constants considered as a possible substitution of the conjunction between atoms under \mathcal{I} . Within π -DL-Lite setting, the certainty degree of \vec{a} is the minimum weight (i.e. *min*) of certainty degrees of constants $a_i \in \vec{a}$. Recall that within standard possibilistic logic, given ϕ and φ two formulas, we have $N(\phi \wedge \varphi) = \min(N(\phi), N(\varphi))$.

Example 3.20. Consider the following ABox: $\langle A(a), \alpha_1 \rangle, \langle B(a), \alpha_2 \rangle, \langle A(b), \alpha_3 \rangle$ and $\langle B(b), \alpha_4 \rangle$ with $\alpha_i \in]0, 1]$ and the query $q(x) \leftarrow A(x) \wedge B(x)$. The answer set to the query q consists of $\langle a, \min(\alpha_1, \alpha_2) \rangle$ and $\langle b, \min(\alpha_3, \alpha_4) \rangle$. ■

Let us consider now the case where the query contains existential quantifications. Suppose that we have a query that contains only one atom with an existential variable y (i.e. $q(x) \leftarrow \exists y.P(x, y)$). Then for each fixed constant a that may substitute x a free variable, there may exist several possible constants c that may substitute y where each one is attached with a certainty degree. In that case, the certainty degree of the answer is the maximal degree (i.e. *max*) for each valid join on y . Recall that within standard possibilistic logic, given ϕ and φ two formulas, we have $N(\phi \vee \varphi) \geq \max(N(\phi), N(\varphi))$.

Example 3.21. Let us consider the following ABox: $\langle P(a, b_1), \alpha_1 \rangle, \langle P(a, b_2), \alpha_2 \rangle, \langle P(b_1, c_1), \alpha_3 \rangle, \langle P(b_1, c_2), \alpha_4 \rangle, \langle P(b_2, c_1), \alpha_5 \rangle, \langle P(b_2, c_2), \alpha_6 \rangle$ with $\alpha_i \in]0, 1]$ and the query $q(x) \leftarrow \exists y.P(x, y)$. The answer set of this query consists of $\langle a, \max(\alpha_1, \alpha_2) \rangle, \langle b_1, \max(\alpha_3, \alpha_4) \rangle$ and $\langle b_2, \max(\alpha_5, \alpha_6) \rangle$. ■

Now the certainty degree of a general conjunctive query $q(\vec{x}) \leftarrow \exists \vec{y}.conj(\vec{x}, \vec{y})$ is simply computed by first considering for each valid join on y the certainty degree of the conjunction of possible substitution using the *min* to aggregate their values, and then considering the maximal degree for each valid join having the same constants as answer.

Example 3.22. Let us consider the ABox of Example 3.21 and the query $q(x_1, x_2) \leftarrow \exists y. P(x_1, y) \wedge P(y, x_2)$. The possible joins and their certainty degrees are:

$\langle a, b_1, c_1, \min(\alpha_1, \alpha_3) \rangle,$
 $\langle a, b_1, c_2, \min(\alpha_1, \alpha_4) \rangle,$
 $\langle a, b_2, c_1, \min(\alpha_2, \alpha_5) \rangle$ and
 $\langle a, b_2, c_2, \min(\alpha_2, \alpha_6) \rangle.$

Then the answer set consists of

$\langle a, c_1, \max(\min(\alpha_1, \alpha_3), \min(\alpha_2, \alpha_5)) \rangle$ and
 $\langle a, c_2, \max(\min(\alpha_1, \alpha_4), \min(\alpha_2, \alpha_6)) \rangle.$ ■

Let us denote by $\langle \vec{a}, \alpha \rangle$ an answer of a query q with \vec{a} is a tuple of constants occurring in $\mathcal{A}_{>inc(\mathcal{K})}$ and α is the least certainty degree in which all atoms occurring in the query q grounded by \vec{a} having

weights at least greater or equal to α are necessarily true. Namely, we say that $\mathcal{K}_{>inc(\mathcal{K})}$ entails a tuple \vec{a} for a query q to a degree α , denoted $\mathcal{K}_{>inc(\mathcal{K})} \models \langle q(\vec{a}), \alpha \rangle$ if and only if $\forall \mathcal{I}, q^I(\vec{a}) \geq \alpha$.

For a given conjunctive query, it may correspond several tuples $\langle \vec{a}_i, \alpha_i \rangle$ as answer sets. So, a more interesting thing is to find the maximal entailment degree of a query. Given q a conjunctive query, this maximal entailment degree of q is as follows:

$$\alpha = \max \{ \alpha_i \mid \mathcal{K}_{>inc(\mathcal{K})} \models \langle q(\vec{a}_i), \alpha_i \rangle \}.$$

Example 3.23. [Example 3.1 continued] Let us consider now the following ABox obtained from $\mathcal{A}_{>Inc(\mathcal{K})}$ and some new assertions:

$\langle \text{NeedBased}(\text{Mary}), \gamma_8 \rangle$	$\langle \text{satisfiedBy}(\text{John}, \text{Gifts1}), \gamma_{10} \rangle$
$\langle \text{NeedBased}(\text{John}), \gamma_9 \rangle$	$\langle \text{satisfiedBy}(\text{John}, \text{Gifts2}), \gamma_6 \rangle$
$\langle \text{NeedBased}(\text{Paul}), 1 \rangle$	$\langle \text{satisfiedBy}(\text{Mary}, \text{Gifts3}), \gamma_{11} \rangle$
$\langle \text{NeedBased}(\text{Bob}), \gamma_{12} \rangle$	$\langle \text{satisfiedBy}(\text{Mary}, \text{Gifts1}), \gamma_5 \rangle$

Let us consider the following query q :

$$q(x) \leftarrow \exists y. \text{NeedBased}(x) \wedge \text{satisfiedBy}(x, y)$$

According to the above ABox, we have two tuples as the answer set of the query q : $\langle \text{John}, \max(\min(\gamma_9, \gamma_{10}), \min(\gamma_9, \gamma_6)) \rangle$ and $\langle \text{Mary}, \max(\min(\gamma_8, \gamma_{11}), \min(\gamma_8, \gamma_5)) \rangle$ where $\mathcal{K}_{>inc(\mathcal{K})} \models \langle q(\text{John}), \gamma_9 \rangle$ and $\mathcal{K}_{>inc(\mathcal{K})} \models \langle q(\text{Mary}), \gamma_8 \rangle$. One can easily check that the maximal entailment degree of q is γ_9 . ■

Now given $Q = \text{ref}(q, \mathcal{T}_{>Inc(\mathcal{K})})$ the set of queries obtained by reformulating q over $\mathcal{T}_{>Inc(\mathcal{K})}$, we view Q as a disjunction of queries. Let us denote by $\text{ans}_\pi(\mathcal{K}, q)$ the answer set of the query q formed by considering the union of all answer sets of each $q_i \in Q$. As said above, for each query $q_i \in Q$, there may correspond a set of tuples $\langle \vec{a}_i, \alpha_i \rangle$ as answer set where the maximal degree of the α_i 's corresponds the maximal entailment degree of q_i . Indeed, the entailment degree of q is the highest degree among the entailment degrees of all q_i 's found by query reformulation since within possibilistic logic, given ϕ and φ two formulas, we have $N(\phi \vee \varphi) \geq \max(N(\phi), N(\varphi))$.

Example 3.24. [Example 3.23 continued] Let us consider the following query q :

$$q(x) \leftarrow \text{Loyal}(x)$$

By applying $\text{PerfectRef}(q, \mathcal{T}_{>\alpha})$ algorithm, the set $\text{ref}(q, \mathcal{T}_{>Inc(\mathcal{K})})$ will contain in addition of the original query the following ones:

$$q_1(x) \leftarrow \text{Discount}(x) \text{ and } q_2(x) \leftarrow \text{NeedBased}(x)$$

Let us consider now the following ABox:

$\langle \text{Loyal}(\text{Mary}), \gamma_7 \rangle$	$\langle \text{Discount}(\text{Bob}), \gamma_6 \rangle$
$\langle \text{Loyal}(\text{John}), \gamma_9 \rangle$	$\langle \text{NeedBased}(\text{Eric}), \gamma_5 \rangle$
$\langle \text{NeedBased}(\text{Paul}), \gamma_{13} \rangle$	$\langle \text{Discount}(\text{Elise}), \gamma_{10} \rangle$

According to the above ABox and the set of queries $\text{ref}(q, \mathcal{T}_{>Inc(\mathcal{K})})$, the answer set $\text{ans}(\mathcal{K}, q)$ of the query q is $\langle \text{John}, \gamma_9 \rangle, \langle \text{Mary}, \gamma_7 \rangle, \langle \text{Paul}, \gamma_{13} \rangle, \langle \text{Bob}, \gamma_6 \rangle, \langle \text{Eric}, \gamma_5 \rangle, \langle \text{Elise}, \gamma_{10} \rangle$. The entailment degree of q is equal to γ_{13} . ■

When the assertional facts in the ABox are attached with weights, an answer to a query is a set of tuples that satisfy with a certain degree the asked query. In that case, one may consider a new reasoning task.

It consists first in ranking answers according to their certainty degrees and then selecting the relevant ones. This task is closely related to the top-k ranking query answering approach introduced in [Straccia, 2006b; Straccia, 2006a; Straccia, 2012] within fuzzy *DL-Lite* logics. The top-k query answering permits to order the results according to their attached degrees and then chose the top-k relevant results. More formally, given $\mathcal{K}=\langle T, A \rangle$ a π -*DL-Lite* knowledge base and $Q=\text{ref}(q, \mathcal{T}_{>Inc(\mathcal{K})})$ the set of queries obtained by reformulating q over $\mathcal{T}_{>Inc(\mathcal{K})}$, we search for the top-k ranked tuples of the answer set of q , denoted $ans_k(\mathcal{K}, q)=\text{Top}_k\{\langle \bar{a}, \alpha \rangle | q_i \in Q \text{ and } \mathcal{A}_{>Inc(\mathcal{K})} \models (q_i(c), \alpha)\}$. To compute the set $ans_k(\mathcal{K}, q)$, one can calculate first the answer set $ans_\pi(\mathcal{K}, q)$ by considering the union of all answer sets of query reformulation, and then order $ans_\pi(\mathcal{K}, q)$ in descending order with respect to degrees of tuples and then selecting the top-k relevant tuples.

Example 3.25. [Example continued] From the answer set $ans(\mathcal{K}, q)$ of Example 3.24, the top-4 relevant results for $q(x)\leftarrow Loyal(x)$ are: $\langle Paul, \gamma_{13} \rangle, \langle Elise, \gamma_{10} \rangle, \langle John, \gamma_9 \rangle, \langle Mary, \gamma_7 \rangle$. ■

3.9 Discussions and related works

There are few works devoted to the possibilistic extension of description logics (DLs). The original work on this topic has been introduced in [Hollunder, 1995] where a possibilistic DL knowledge base is only syntactically defined as a set of terminological (*resp.* assertion) axioms attached with possibility or necessity degrees. In [Hollunder, 1995], the author is only interested in basic inference tasks such as instance checking and subsumption problem and extends them with respect to possibilistic entailments. These inference services allow to check whether an axiom is inferred or not from a possibilistic DL knowledge base and the degree of possibility or the necessity of its entailment. From an algorithmic point of view, Hollunder's [Hollunder, 1995] method uses classical inference algorithm for DLs to check inferences. In our definition of a possibilistic *DL-Lite* knowledge base, we only represent certainty degrees using necessity values. This is in agreement with standard propositional logic, and has a meaningful counterpart when generating possibility distributions.

In [Qi *et al.*, 2011; Qi *et al.*, 2007b; Qi *et al.*, 2007a], the authors first provided the syntax and semantics of possibilistic DLs knowledge bases and then they define inference process. In this chapter, the syntax and semantic of π -*DL-Lite* knowledge bases is given in a similar way as in [Qi *et al.*, 2011].

Given \mathcal{K} a possibilistic DL knowledge base, in [Qi *et al.*, 2011; Qi *et al.*, 2007b; Qi *et al.*, 2007a] an α -cut approach is used to compute the inconsistency degree of \mathcal{K} . Namely, the inconsistency degree of \mathcal{K} is the maximum weight of axioms making the α -cut of \mathcal{K} by this weight inconsistent and the strict α -cut of \mathcal{K} by this same weight consistent. Using such approach, the algorithm proposed in [Qi *et al.*, 2011; Qi *et al.*, 2007b; Qi *et al.*, 2007a] operates a binary search to find this weight and this comes down to achieve a number of calls, at most $\log_2(n) + 1$, where n is the size of the different uncertainty scale appearing in the knowledge base, to a standard DLs (without uncertainty) inconsistency checking algorithm. This definition is in the spirit of standard possibilistic logic [Dubois *et al.*, 1994]. In [Calvanese *et al.*, 2007a] checking inconsistency of a standard *DL-Lite* knowledge base can be done in a polynomial time with respect to combined complexity k (the size of the whole knowledge base). Knowing that the size n of certainty scales is at least equal to the size of the knowledge base k , it is important to note that although the α -cut algorithm [Qi *et al.*, 2011; Qi *et al.*, 2007b; Qi *et al.*, 2007a] requires an extra cost (at least $\log_2(n)$ calls to a standard DL inconsistency check algorithm in order to compute inconsistency degree). However, this still can be achieved in polynomial time.

Other lines of approaches have been proposed to compute the inconsistency degrees of possibilistic DLs knowledge bases, such as the works of [Couchariere *et al.*, 2008a; Couchariere *et al.*, 2008b; Qi *et al.*, 2008b; Zhu *et al.*, 2013]. These approaches are based on the extension of a Tableau algorithm within possibilistic logic setting. As shown in [Couchariere *et al.*, 2008a; Qi *et al.*, 2008b; Zhu *et al.*, 2013],

such extension does not require any extra computational cost. However, these approaches are not defined within *DL-Lite* languages.

In [Zhou *et al.*, 2009], the authors give a method to measure the inconsistency of a *DL-Lite* knowledge base based on the three-valued semantics. The proposed algorithm calculates the number of axioms that fall into the third truth value (denoting contradictory information) and can be achieved in polynomial time. This clearly departs from our approach.

In our work, computing inconsistency degree is done by a direct extension of the main standard consistency check algorithm presented in [Calvanese *et al.*, 2007a] of a standard *DL-Lite* knowledge base. This represents a new way to compute the inconsistency degree of a possibilistic *DL-Lite* knowledge base that departs from the existing works. There is no existing work that extends rules for the defined negated closure in a possibility theory framework. Our approach comes down to first defining the notion of negated closure when each axiom in the knowledge base is equipped with a certainty degree. This negated closure is then transformed to weighted queries performed over the set of individuals in order to compute the inconsistency degree. The inconsistency degree associated with a query and a given tuple of individuals (provided as an answer for the query) is the maximum weight among all the certainty degrees of the query and this tuple. The computational complexity associated to this procedure does not require any extra cost comparing with the one of consistency checking algorithm of classical *DL-Lite* knowledge bases.

3.10 Conclusion

In this chapter, we proposed a possibilistic extension of *DL-Lite*. We first introduced the syntax and the semantics of such extension. We provided properties of π -*DL-Lite* and showed how to compute the inconsistency degree of π -*DL-Lite* knowledge base having a complexity identical to the one used in standard *DL-Lite*. This is done by defining π -*DL-Lite* negative closure that extends the one of standard *DL-Lite*. Then, we gave a method to check the consistency for π -*DL-Lite*. Finally, we discussed inference problems. In particular, we distinguished different inference tasks depending whether we use flat inferences or weighted inferences. Results of this chapter are important since they extend *DL-Lite* languages to deal with priorities (between TBox axioms or ABox axioms) or uncertainty degrees without increasing the computational complexity.

In several situations, pieces of information are provided with uncertainty which can simply represent reliability of the distinct sources. Next chapter addresses the problem of fusion of multiple data sources linked to the same terminology in the case where the sources have different levels of priorities.

MIN-BASED CONDITIONING AND MERGING APPROACH OF *DL-Lite* KNOWLEDGE BASES

4.1 Introduction

In several situations, pieces of information are provided with uncertainty which can represent reliability of the distinct sources. Possibility theory is a very natural framework to deal with such pieces of information. In the framework of possibility theory, several approaches for merging possibilistic logic bases have been proposed (e.g. [Benferhat *et al.*, 1999; Benferhat *et al.*, 2000; Benferhat *et al.*, 1993a; Qi *et al.*, 2010b]). However there is no work that deals with the problem of merging possibilistic DLs knowledge bases by introducing convenient fusion operators although the impact of possibilistic DLs on ontology merging has been introduced in [Qi *et al.*, 2011].

In the first part of this chapter, we study merging within possibilistic *DL-Lite* framework. We first focus on the use of minimum-based (min-based) operator, well known as idempotent conjunctive operator presented in Section 2.3.2, for merging possibilistic *DL-Lite* knowledge bases. We then place ourselves in the context of Ontology-based Data Access (OBDA) setting, in which a TBox is used to reformulate posed queries to offer a better access to the set of data encoded in the ABox [Poggi *et al.*, 2008]. We go one step further in the definition of merging operators for π -*DL-Lite* knowledge bases by investigating the aggregation of assertional bases (ABox) which are linked to the same TBox. Two important results of this study are:

- Our merging approach based on conflict resolution can be extended to define other merging operators, and
- The computational complexity of *min*-based assertional fusion outcome is polynomial.

Another important reasoning task in possibilistic setting is the one of conditioning possibilistic knowledge bases [Benferhat *et al.*, 2002c]. Unfortunately, there is to the best of our knowledge no approach for conditioning possibilistic DLs or *DL-Lite* when a new uncertain information is available. This chapter fills this gap and gives a first result of possibilistic *DL-Lite* conditioning.

The rest of this chapter is organized as follows: Section 4.2 first introduces merging of π -*DL-Lite* possibility distributions using a min-based operator and then discusses the syntactic counterpart when merging π -*DL-Lite* knowledge bases. In Section 4.3, we first introduce a syntactic merging operator, namely a *min*-based assertional operator based on conflict resolution. We show that such a merging operator gives a more satisfactory result compared with the one proposed in Section 4.2. We then study merging at a semantic level, and we show that our operator has a natural counterpart when combining several possibility distributions. Lastly, we rephrase the set of postulates proposed in [Koniuczny and Pino Pérez, 2002] to characterize the logical behavior of belief bases merging operators and we provide a postulates-based logical analysis of the *min*-based assertional operator in the light of this new set of

postulates dedicated to the uncertain *DL-Lite* framework. Section 4.4 presents first result on possibilistic *DL-Lite* conditioning and Section 4.5 concludes the chapter.

4.2 Min-based merging of π -*DL-Lite* knowledge bases

This section first introduces merging of π -*DL-Lite* possibility distributions using a min-based operator and then discusses the syntactic counterpart of the proposed operator when merging π -*DL-Lite* knowledge bases.

Example 4.1. Let $\mathcal{K}=\langle \mathcal{T}, \mathcal{A} \rangle$ be a π -*DL-Lite* knowledge base where $\mathcal{T}=\{(T \sqsubseteq \neg S, .8), (\exists F \sqsubseteq T, .6), (\exists F^- \sqsubseteq S, .5)\}$ and $\mathcal{A}=\{(F(b, c), 1)\}$. The possibility distribution $\pi_{\mathcal{K}}$ associated to \mathcal{K} is computed using Definition 3.2 as follows where $\Delta^{\mathcal{I}}=\{b, c\}$:

\mathcal{I}	$\Delta^{\mathcal{I}}$	$\pi_{\mathcal{K}}$
\mathcal{I}_1	$S = \{b, c\}, T = \{b\}, F = \{(b, c)\}$.2
\mathcal{I}_2	$S = \{b, c\}, T = \{\}, F = \{(b, c)\}$.4
\mathcal{I}_3	$S = \{c\}, T = \{b\}, F = \{(b, c)\}$	1

Table 4.1: Example of a possibility distribution induced from a π -*DL-Lite* knowledge base

One can observe that $\pi_{\mathcal{K}}(\mathcal{I}_3)=1$ meaning that $\pi_{\mathcal{K}}$ is normalized, and thus, \mathcal{K} is consistent. ■

4.2.1 Merging of π -*DL-Lite* possibility distributions

In a possibility theory framework, several fusion operators (e.g. [Dubois *et al.*, 1992; Benferhat *et al.*, 1997b; Benferhat and Kaci, 2003]) have been proposed for merging pieces of information issued from different and potentially conflicting or inconsistent sources. These fusion operators lead to combine multiple possibility distributions that encode sources of information to obtain a unique possibility distribution that represent the global point of view of available information. Moreover, a syntactic counterpart for each fusion operator used to combine possibility distributions has been introduced to merge possibilistic knowledge bases.

When the distinct sources that provide possibility distributions are dependent, then the recommended fusion operator is the min-based operator well-known as idempotent conjunctive operator. The aim of this section is to study semantic merging of π -*DL-Lite* possibility distributions using min-based operator. The syntactic counterpart of this combination will be presented in Section 4.2.2. Let us assume that π_1, \dots, π_n are possibility distributions provided by n sources of information that share the same domain of interpretations (namely $\Delta_1^{\mathcal{I}} = \dots = \Delta_n^{\mathcal{I}}$), and that all possibility distributions use the same scale to represent uncertainty.

Definition 4.1. A min-based operator or idempotent conjunctive operator, denoted by \oplus , is a mapping from $\forall \mathcal{I} \in \Omega, \nu(\mathcal{I}) = (\pi_1(\mathcal{I}), \dots, \pi_n(\mathcal{I}))$ a vectors of possibility values to an interval $[0, 1]$ defined as follows:

$$\pi_{\oplus}(\mathcal{I}) = \min_{\mathcal{I} \in \Omega}(\nu(\mathcal{I}))$$

According to Definition 4.1, the min-based operator ignores redundancy. Since, if all the sources provide the same possibility distribution then the result of fusion using \oplus is the same possibility distribution.

Proposition 4.1. *The min-based operator satisfies the characteristic properties:*

- If $\forall \mathcal{I} \in \Omega : \pi_i(\mathcal{I}) = 1$ then $\pi_{\oplus}(\mathcal{I}) = 1$.
- If $\forall \mathcal{I}', \forall \mathcal{I} \in \Omega : \pi_i(\mathcal{I}) \geq \pi_i(\mathcal{I}')$ then $\pi_{\oplus}(\mathcal{I}) \geq \pi_{\oplus}(\mathcal{I}')$.

Example 4.2. Let us continue example 4.1. Let π_2 be the possibility distribution that encodes pieces of information of \mathcal{K}_2 a π -DL-Lite knowledge base. Assume that $\Delta^{\mathcal{I}} = \{b, c\}$:

\mathcal{I}	\mathcal{I}	π_2
\mathcal{I}_1	$S = \{b, c\}, T = \{b\}, P = \{b\}$	1
\mathcal{I}_2	$S = \{b, c\}, T = \{\}, P = \{\}$	1
\mathcal{I}_3	$S = \{c\}, T = \{b\}, P = \{\}$.1

Using Definition 4.1, we present the following possibility distribution π_{\oplus} as the result of merging π_1 (Example 4.1) and π_2 .

\mathcal{I}	\mathcal{I}	π_{\oplus}
\mathcal{I}_1	$S = \{b, c\}, T = \{b\}, P = \{b\}, F = \{(b, c)\}$.2
\mathcal{I}_2	$S = \{b, c\}, T = \{\}, P = \{\}, F = \{(b, c)\}$.4
\mathcal{I}_3	$S = \{c\}, T = \{b\}, P = \{\}, F = \{(b, c)\}$.1

■

One can easily check that merging two π -DL-Lite normalized possibility distributions may lead to a sub-normalized possibility distribution. This is the case with our example (Example 4.2).

Now, we focus on the normalization problem when the use of min-based operator \oplus provides a subnormalized possibility distribution. Let us consider:

$$h(\pi_{\oplus}) = \max_{\mathcal{I} \in \Omega} \{\pi_{\oplus}(\mathcal{I})\}$$

a function that computes to what extent there exists at least one interpretation which is confirmed by all sources. Considering $\pi_{N\oplus}$ the normalized possibility distribution of π_{\oplus} , $\pi_{N\oplus}$ must respect the following conditions.

Proposition 4.2. $\forall \mathcal{I}', \forall \mathcal{I} \in \Omega$, the minimal requirements for $\pi_{N\oplus}$ are:

- $\exists \mathcal{I}, \pi_{N\oplus}(\mathcal{I}) = 1$.
- If $\pi_{\oplus}(\mathcal{I}) > \pi_{\oplus}(\mathcal{I}')$ then $\pi_{N\oplus}(\mathcal{I}) > \pi_{N\oplus}(\mathcal{I}')$.
- If $\pi_{\oplus}(\mathcal{I})$ is normalized then $\pi_{\oplus}(\mathcal{I}) = \pi_{N\oplus}(\mathcal{I})$.

The first condition states that $\pi_{N\oplus}$ must be normalized (there exists at least one interpretation such that $\pi_{N\oplus}(\mathcal{I}) = 1$). The second condition entails that only interpretations having possibility degrees equal to $h(\pi_{\oplus})$ can receive value 1 in the normalization process. Assuming these following requirements for $\pi_{N\oplus}$ (Proposition 4.2), we consider the following definition of normalization.

Definition 4.2. For every $\mathcal{I} \in \Omega$ and $h(\pi_{\oplus}) > 0$

$$\pi_{N\oplus}(\mathcal{I}) = \begin{cases} 1 & \text{if } \pi_{\oplus}(\mathcal{I}) = h(\pi_{\oplus}) \\ \pi_{\oplus}(\mathcal{I}) & \text{otherwise} \end{cases}$$

Example 4.3. [Example 4.2 continued] Using Definition 4.2, we present the following normalized possibility distribution $\pi_{N\oplus}$ from π_{\oplus} .

\mathcal{I}	\mathcal{I}	$\pi_{N_{\oplus}}$
\mathcal{I}_1	$S = \{b, c\}, T = \{b\}, P = \{b\}, F = \{(b, c)\}$.2
\mathcal{I}_2	$S = \{b, c\}, T = \{\}, P = \{\}, F = \{(b, c)\}$	1
\mathcal{I}_3	$S = \{c\}, T = \{b\}, P = \{\}, F = \{(b, c)\}$.1

■

4.2.2 Syntactical merging of π -DL-Lite knowledge bases

Let us consider $\mathcal{K}_1, \dots, \mathcal{K}_n$ a set of π -DL-Lite knowledge bases where each \mathcal{K}_i represents the knowledge of a single source of information or agent and π_1, \dots, π_n is a set of n possibility distributions associated with $\mathcal{K}_1, \dots, \mathcal{K}_n$. Namely each π -DL-Lite knowledge base \mathcal{K}_i is associated with a possibility distribution π_i which is its semantical counterpart. In the previous section, we have presented merging of possibility distributions using min-based operator. In this section, we give a syntactical counterpart of merging n π -DL-Lite knowledge bases $\mathcal{K}_1, \dots, \mathcal{K}_n$ provided by n different sources.

Let us consider S_1, \dots, S_n be the signatures of $\mathcal{K}_1, \dots, \mathcal{K}_n$. A signature S is the set of concept names, role names and individual names used in \mathcal{K} . We assume that all \mathcal{K}_i 's share the same signature. Namely if a concept name (*resp.* role name, individual name) A appears in S_1 and S_2 then A is assumed to be the same.

Now, we look to identify syntactically the min-based operator \oplus on the \mathcal{K}_i 's which correspond to the min-based operator \oplus on the π_i 's reviewed in the Section 4.2.1. More formally, given (π_1, \dots, π_n) possibility distributions associated with $(\mathcal{K}_1, \dots, \mathcal{K}_n)$ π -DL-Lite knowledge bases, then for the min-based operator \oplus applied to (π_1, \dots, π_n) , we look for a π -DL-Lite knowledge base \mathcal{K}_{\oplus} constructed from $(\mathcal{K}_1, \dots, \mathcal{K}_n)$ such that $\pi_{\oplus} = \pi_{\mathcal{K}_{\oplus}}$.

Definition 4.3. The syntactic counterpart of the min-based operator \oplus for π -DL-Lite knowledge bases is defined as follow where $(\phi_i, \alpha_i) \in \mathcal{K}_1$ and $(\varphi_i, \beta_i) \in \mathcal{K}_2$:

$$\mathcal{K}_{\oplus} = \mathcal{K}_1 \cup \mathcal{K}_2$$

Example 4.4. Let $\mathcal{K}_2 = \langle \mathcal{T}, \mathcal{A} \rangle$ be a π -DL-Lite knowledge base where $\mathcal{T} = \{(P \sqsubseteq S, .7), (P \sqsubseteq T, .9)\}$ and $\mathcal{A} = \{(S(b), .9)\}$. The possibility distribution π_2 associated to \mathcal{K}_2 is presented in Example 4.2.

Now, using Definition 4.3, we present the following π -DL-Lite knowledge base \mathcal{K}_{\oplus} as result of merging \mathcal{K}_1 (Example 4.1) and \mathcal{K}_2 : $\mathcal{T}_{\oplus} = \{(T \sqsubseteq \neg S, .8), (P \sqsubseteq S, .7), (P \sqsubseteq T, .9), (\exists F \sqsubseteq T, .6), (\exists F^- \sqsubseteq S, .5)\}$ $\mathcal{A}_{\oplus} = \{(S(b), .9), (F(b, c), 1)\}$. The semantic counterpart of \mathcal{K}_{\oplus} is reported in Example 4.2 through the possibility distribution π_{\oplus} . ■

Remark 4.1. In DL-Lite literature, it is often assumed that TBox are consistent (admit a model). This makes sense when one has one source of information. However, when we deal with multiple sources such assumption is questionable. In particular, it may happen that, one can check that there is no ABox \mathcal{A} such that $\mathcal{T}_{\oplus} \cup \mathcal{A}$ is consistent.

In general, merging two π -DL-Lite consistent knowledge bases may lead to an inconsistent knowledge base. This is the case with our example (Example 4.4) where \mathcal{K}_1 (Example 4.1) and \mathcal{K}_2 (Example 4.4) are consistent but their merging is inconsistent. Hence, we study the normalization problem at the syntactical level when the use of min-based operator \oplus provides an inconsistent π -DL-Lite knowledge base.

Let \mathcal{K}_{\oplus} be a π -DL-Lite knowledge base associated with π_{\oplus} , a sub-normalized possibility distribution where $h(\pi_{\oplus}) = \max_{\mathcal{I} \in \Omega} \{\pi_{\oplus}(\mathcal{I})\}$. Then the normalization rule can be defined as follows where $\mathcal{K}_{N_{\oplus}}$ denotes the normalized π -DL-Lite knowledge base:

Proposition 4.3. *The possibility distribution, where*

$$\mathcal{I} \in \Omega: \pi_{N\oplus}(\mathcal{I}) = \begin{cases} 1 & \text{if } \pi_{\oplus}(\mathcal{I}) = h(\pi_{\oplus}) \\ \pi_{\oplus}(\mathcal{I}) & \text{otherwise} \end{cases}$$

is associated with:

$$\mathcal{K}_{N\oplus} = \{(\phi_i, \alpha_i) \mid (\phi_i, \alpha_i) \in \mathcal{K}_{\oplus} \text{ and } \alpha_i > 1 - h(\pi_{\oplus})\}$$

According to Definition 4.3, the normalization does not modify the certainty degrees of π -DL-Lite knowledge base encoded by π_{\oplus} . It just permits to ignore the presence of contradictions (or conflicts) and maintains all the axioms of \mathcal{K}_{\oplus} whose certainty degrees are higher than the inconsistency degree of \mathcal{K}_{\oplus} .

Example 4.5. Using Definition 4.3, we give the normalized π -DL-Lite knowledge base $\mathcal{K}_{N\oplus}$ associated to the normalized possibility distribution $\pi_{N\oplus}$: $\mathcal{T}_{\oplus} = \{(T \sqsubseteq \neg S, .8), (P \sqsubseteq S, .7), (P \sqsubseteq T, .9)\}$ $\mathcal{A}_{\oplus} = \{(S(b), .95), F(b, c), 1)\}$.

In this example, it is easy to see that $\mathcal{K}_{N\oplus}$ is consistent and this is confirmed through $\pi_{N\oplus}$ where the interpretation \mathcal{I}_1 is a model of $\mathcal{K}_{N\oplus}$. ■

It is important to note that the normalization process allows to find an inconsistency degree identical to the one computed using an algorithmic approach proposed in Section 3.5 to compute the inconsistency degree of a π -DL-Lite knowledge base.

4.3 Min-based assertional merging approach for π -DL-Lite knowledge bases

This section introduces a syntactic merging operator, namely a *min*-based assertional operator based on conflict resolution. We show that such a merging operator gives a more satisfactory result compared with the one proposed in the previous Section 4.2 and has a natural semantic counterpart.

4.3.1 Syntactic merging of π -DL-Lite assertional bases

Let us consider $\mathcal{A}_1, \dots, \mathcal{A}_n$ a set of assertional bases (ABox) where each \mathcal{A}_i represents assertional facts provided by a single source of information. We assume that we have a well-formed and coherent terminological base (TBox) \mathcal{T} where each \mathcal{A}_i is consistent with \mathcal{T} .

In this section, we study syntactic merging of n assertional bases $\mathcal{A}_1, \dots, \mathcal{A}_n$ that are linked to the same TBox \mathcal{T} . We cast available information within the π -DL-Lite framework. For the sake of simplicity, we omit the weights notation attached to the TBox axioms considered as the ones having the highest certainty level, namely, an axiom in \mathcal{T} is of the form $(\varphi, 1)$. We only represent explicitly weights attached to \mathcal{A}_i assertions. An assertion f in \mathcal{A}_i is of the form $f = (\varphi, \alpha)$ where $\alpha \in [0, 1]$. Note that copies of the same assertions φ are allowed in several \mathcal{A}_i and they are considered as different in the sense of priorities or certainty and not in terms of interpretations since we use the *Unique Name Assumption*.

Let us consider S_1, \dots, S_n be the signatures of $\mathcal{A}_1, \dots, \mathcal{A}_n$ and \mathcal{T} . Recall that a signature S of a knowledge base \mathcal{K} is the set of concept names, role names and individual names used in \mathcal{K} . We assume that all \mathcal{A}_i 's and \mathcal{T} share the same signature. We look to identify a syntactical merging operator on the \mathcal{A}_i 's with respect to a TBox \mathcal{T} which will be semantically meaningful. Merging at semantic level will be presented in Section 4.3.2.

Example 4.6. Let $\mathcal{K}=\langle \mathcal{T}, \mathcal{A} \rangle$ be a π -DL-Lite knowledge base where $\mathcal{T}=\{(A \sqsubseteq B, 1), (B \sqsubseteq \neg C, .9)\}$ and $\mathcal{A}=\{(A(a), .6), (C(b), .5)\}$. The possibility distribution $\pi_{\mathcal{K}}$ associated to \mathcal{K} is computed using Definition 3.2 as follows where $\Delta^{\mathcal{T}}=\{a, b\}$:

I	I	$\pi_{\mathcal{K}}$
I_1	$A=\{a\}, B=\{ \}, C=\{b\}$	0
I_2	$A=\{a\}, B=\{a\}, C=\{b\}$	1
I_3	$A=\{ \}, B=\{ \}, C=\{a,b\}$.4
I_4	$A=\{a,b\}, B=\{a,b\}, C=\{ \}$.5

Table 4.2: Example of a possibility distribution induced from a π -DL-Lite KB

One can observe that $\pi_{\mathcal{K}}(I_2)=1$, meaning that $\pi_{\mathcal{K}}$ is normalized, and thus, \mathcal{K} is consistent. ■

Merging using the classical *min*-based operator

To show properties of our assertional-based merging operator, we first perform merging of $\mathcal{A}_1, \dots, \mathcal{A}_n$ with respect to \mathcal{T} using the classical *min*-based merging operator proposed in Section 4.2 to aggregate π -DL-Lite knowledge bases. The *min*-based merging operator, denoted by \oplus considers the union of all ABox. Namely: $\mathcal{A}_{\oplus}=\mathcal{A}_1 \cup \mathcal{A}_2 \cup \dots \cup \mathcal{A}_n$. The following definition gives the formal logical representation of the normalized knowledge base.

Definition 4.4. Let \mathcal{T} be a TBox and \mathcal{A}_{\oplus} be the aggregation of $\mathcal{A}_1, \dots, \mathcal{A}_n$, n ABox using classical *min*-based operator. Let $x=Inc(\langle \mathcal{T}, \mathcal{A}_{\oplus} \rangle)$. Then, the normalized knowledge base, denoted, $\mathcal{K}_{N\oplus}$ is such that:

$$\mathcal{K}_{N\oplus}=\langle \mathcal{T}, \{(\varphi, \alpha) : (\varphi, \alpha) \in \mathcal{A}_{\oplus} \text{ and } \alpha > x\} \rangle$$

Example 4.7. [Example 4.6 continued] Let us continue with the TBox $\mathcal{T}=\{A \sqsubseteq B, B \sqsubseteq \neg C\}$ presented in Example 4.6 while assuming that the certainty degree of each axioms is set to 1. Let us consider the following set of ABox to be linked to \mathcal{T} :

$$\mathcal{A}_1=\{(A(a), .6), (C(b), .5)\},$$

$$\mathcal{A}_2=\{(C(a), .4), (B(b), .8), (A(b), .7)\} \text{ and}$$

$$\mathcal{A}_3=\{(A(b), .2), (A(c), .5), (B(c), .4)\}.$$

We have $\mathcal{A}_{\oplus}=\{(A(a), .6), (C(b), .5), (C(a), .4), (B(b), .8), (A(b), .7), (A(b), .2), (A(c), .5), (B(c), .4)\}$ where $Inc(\langle \mathcal{T}, \mathcal{A}_{\oplus} \rangle) = .5$. Then $\mathcal{K}_{N\oplus}=\mathcal{T} \cup \{(A(a), .6), (B(b), .8), (A(b), .7)\}$. ■

According to Definition 4.4, merging operation does not modify the certainty degrees of the π -DL-Lite knowledge base. It just permits to ignore the presence of contradictions (or conflicts) and maintain all the assertions of \mathcal{A}_{\oplus} whose certainty degrees are higher than the inconsistency degree of $\langle \mathcal{T}, \mathcal{A}_{\oplus} \rangle$. It is clear that the formal expression of the normalized π -DL-Lite knowledge base \mathcal{K}_{\oplus} given in Definition 4.4 provides a consistent knowledge base. However, this result is not very satisfactory, since many assertions in $\mathcal{A}_1, \dots, \mathcal{A}_n$, which are not involved in any conflict are thrown out.

Example 4.8. [Example 4.7 continued] One can see that the assertions $(A(c), .5)$ and $(B(c), .4)$ are not involved in any conflict, but they are not integrated in the merging result. ■

To this end, we investigate a new approach to merge assertional bases based on conflict detection.

Min-based assertional merging using conflict resolution

Let $\mathcal{K}=\langle\mathcal{T}, \mathcal{A}\rangle$ be a π -DL-Lite knowledge base. In Section 3.5, it was shown that computing the inconsistency degree of \mathcal{K} comes down to compute the one of $\langle\pi - neg(\mathcal{T}), \mathcal{A}\rangle$ where $\pi-neg(\mathcal{T})$ is the negated closure of \mathcal{T} . Indeed, computing inconsistency degree of \mathcal{K} consists in calculating the maximal weight attached to minimal inconsistent subsets involved in inconsistency (i.e. Definition 3.6). Within a DL-Lite setting, the inconsistency problem is always defined with respect to some ABox, since a TBox may be incoherent but never inconsistent. Recall that in this chapter, we assume that \mathcal{T} is coherent. So, from the definition of minimal inconsistent subsets, we define the notion of ABox conflict as a minimal inconsistent subset of assertions. More formally:

Definition 4.5. Let $\mathcal{K}=\langle\mathcal{T}, \mathcal{A}\rangle$ be an inconsistent π -DL-Lite knowledge base where axioms in \mathcal{T} are set to 1. A subbase $\mathcal{C} \subseteq \mathcal{A}$ is said to be an assertional conflict set of \mathcal{K} if and only if

- $Inc(\langle\mathcal{T}, \mathcal{C}\rangle) > 0$ and
- $\forall f \in \mathcal{C}, Inc(\langle\mathcal{T}, \mathcal{C} - \{f\}\rangle)=0$

It is clear that in Definition 4.5, removing any assertion φ from \mathcal{C} restores the consistency of $\langle\mathcal{T}, \mathcal{C}\rangle$. Recall that when the TBox is coherent, a conflict involves exactly two assertions.

Example 4.9. [Example 4.7 continued] Let us consider \mathcal{T} and \mathcal{A}_{\oplus} from the above example. The $\pi-neg(\mathcal{T})=\{A \sqsubseteq \neg C, B \sqsubseteq \neg C\}$. One can compute the following conflict sets:

$$\begin{aligned} \mathcal{C}_1 &= \{(A(a), .6), (C(a), .4)\}, \\ \mathcal{C}_2 &= \{(C(b), .5), (B(b), .8)\}, \\ \mathcal{C}_3 &= \{(C(b), .5), (A(b), .7)\} \text{ and} \\ \mathcal{C}_4 &= \{(C(b), .5), (A(b), .2)\}. \end{aligned}$$

Let us assume that $\mathcal{A}_1, \dots, \mathcal{A}_n$ are assertional bases provided by n sources of information to be linked to the same TBox \mathcal{T} and they use the same scale to represent uncertainty. Let denote by $f=(\varphi, \alpha)$ an assertion or a fact in \mathcal{A}_i , we define the notion of conflict vector as follows:

Definition 4.6. Let \mathcal{T} be a TBox and $\mathcal{A}_1, \dots, \mathcal{A}_n$ be a set of ABox provided by n distinct sources of information to be linked to \mathcal{T} . Then $\forall f \in \mathcal{A}_i$ we define a conflict vector associated with,

$$\forall i \in \{1, \dots, n\}, \forall f = (\varphi, \alpha) \in \mathcal{A}_i, \mathcal{V}(f) = \langle \nu_1, \nu_2, \dots, \nu_n \rangle$$

such that:

$$\forall j = 1..n : \nu_j(f) = \begin{cases} 1 & \text{if } \langle\mathcal{T}, \{(\varphi, 1) \cup \mathcal{A}_i\}\rangle \text{ is consistent} \\ Inc(\langle\mathcal{T}, \{(\varphi, 1) \cup \mathcal{A}_i\}\rangle) & \text{otherwise} \end{cases}$$

Where ν_i represents the i^{th} component of the vector \mathcal{V} .

Intuitively, for each assertion provided by an information source we built upon a vector that represents to what extend this latter contradicts the other ones provided by other source. To this end, we add first the assertion with a highest prescribed level in each source and then we compute the inconsistency degree of this one. It is obvious that the conflict vector of a non conflicting assertion is equal to $\mathcal{V}(f)=\langle 1, 1, \dots, 1\rangle$. However assertions that are involved in conflict will have at least a ν_i strictly less than 1.

Example 4.10 (Example continued). One can obtain the following conflict vectors:

$$\begin{aligned} \mathcal{V}((A(a), .6)) &= \langle 1, .6, 1 \rangle, \\ \mathcal{V}((A(b), .7)) &= \langle .5, 1, 1 \rangle, \end{aligned}$$

$\mathcal{V}((A(b), .2)) = \langle .5, 1, 1 \rangle$,
 $\mathcal{V}((A(c), .5)) = \langle 1, 1, 1 \rangle$,
 $\mathcal{V}((B(b), .8)) = \langle .5, 1, 1 \rangle$,
 $\mathcal{V}((B(c), .4)) = \langle 1, 1, 1 \rangle$,
 $\mathcal{V}((C(a), .4)) = \langle .4, 1, 1 \rangle$ and
 $\nu((C(b), .5)) = \langle 1, .2, .8 \rangle$

■

From now on, we give the way to aggregate assertional bases using conflict vectors attached to each assertion. Let denote by Σ the set of conflict vectors, we define the *min*-based assertional merging operators, denoted by Λ as follows:

Definition 4.7. Let \mathcal{T} be a TBox and $\mathcal{A}_1, \mathcal{A}_2, \dots, \mathcal{A}_n$ be a set of ABox provided by n sources to be linked to \mathcal{T} . Let Σ be the collection of conflict vectors associated to each assertion on \mathcal{A}_i . Then the *min*-based assertional merging operator, denoted by Λ , is defined on Σ as follows:

$$\forall \mathcal{V}(f) \in \Sigma: \Lambda(f) = \min\{\nu_i(f)\}$$

Let us denote by Σ_Λ , the vector resulting by *min* aggregation of conflict vectors.

Example 4.11 (Example continued). Σ_Λ contains the following elements:

$\Lambda((A(a), .6)) = .6$,
 $\Lambda((A(b), .7)) = .5$,
 $\Lambda((A(b), .2)) = .5$,
 $\Lambda((A(c), .5)) = 1$,
 $\Lambda((B(b), .8)) = .5$,
 $\Lambda((B(c), .4)) = 1$,
 $\Lambda((C(a), .4)) = .4$ and
 $\Lambda((C(b), .5)) = .2$.

■

According to conflict vectors, one can associate to the set of assertions a new pre-order by attaching to each of them a new weight (*i.e.* $\forall (\varphi, \alpha) \in \mathcal{A}_i: (\varphi, \alpha) = (\varphi, \Lambda(f))$). According to this new pre-order, we define the knowledge base resulting from the fusion operation as follows.

Definition 4.8. Let \mathcal{T} be a TBox and $\mathcal{A}_1, \dots, \mathcal{A}_n$ be a set of n ABox to be linked to \mathcal{T} . Let $\mathcal{A}_\Lambda = \{(\varphi, \Lambda(f)) : f = (\varphi, \alpha) \in \mathcal{A}_i \text{ and } \Lambda(f) \in \Sigma_\Lambda\}$. Let $x = \text{Inc}(\langle \mathcal{T}, \mathcal{A}_\Lambda \rangle)$. Then the resulting knowledge base \mathcal{K}_Λ is such that:

$$\mathcal{K}_\Lambda = \langle \mathcal{T}, \{(\varphi, \alpha) : (\varphi, \alpha) \in \mathcal{A}_\Lambda \text{ and } \alpha > x\} \rangle$$

Example 4.12 (Example continued). One can obtain:

$\mathcal{A}_\Lambda = \{(A(a), .6), (A(b), .5), (A(b), .5), (A(c), 1), (B(b), .5), (B(c), 1), (C(a), .4), (C(b), .2)\}$

where $\text{Inc}(\langle \mathcal{T}, \mathcal{A}_\Lambda \rangle) = .4$.

Then $\mathcal{K}_\Lambda = \mathcal{T} \cup \{(A(a), .6), (A(b), .5), (A(b), .5), (A(c), 1), (B(b), .5), (B(c), 1)\}$.

■

According to Definition 4.8, it is clear that method based on conflict vectors is more productive than the classical definition of the *min*-based merging operator proposed in Definition 4.4. Note that this approach can easily propose others aggregation modes such as product-based merging or sum-based merging.

The definition of this merging operator is based on a notion of conflict measure between sources of information. However, one can observe that original weights attached to assertions are lost. Regarding for instance assertion $B(c)$, it is provided by only one source where its initial weight was .4. This means that

$B(c)$ is not a totally reliable information. In the new knowledge base its weight is raised to 1. This can be justified by the fact that such assertion is not involved in any conflict. However when we proceed to an iteration process this approach may be not very useful. To overcome such limitation while preserving the same productivity of the fusion result, we propose the following definition.

Definition 4.9. Let \mathcal{T} be a TBox and $\mathcal{A}_1, \dots, \mathcal{A}_n$ be a set of n ABox to be linked to \mathcal{T} . Let $\mathcal{A}_\Lambda = \{(\varphi, \Lambda(f)) : (\varphi, \alpha) \in \mathcal{A}_i\}$. Let $x = Inc(\langle \mathcal{T}, \mathcal{A}_\Lambda \rangle)$. Then the resulting knowledge base \mathcal{K}'_Λ is such that:

$$\mathcal{K}'_\Lambda = \langle \mathcal{T}, \{f = (\varphi, \alpha) \in \mathcal{A}_i : i \in \{1, \dots, n\}, (\varphi, \Lambda(f)) \in \mathcal{A}_\Lambda \text{ and } \Lambda(f) > x\} \rangle$$

4.3.2 Semantic counterpart

Let us consider $\mathcal{A}_1, \dots, \mathcal{A}_n$ a set of ABox's where each \mathcal{A}_i represents data of a single source of information. We assume that we have a well-formed and coherent TBox \mathcal{T} where each \mathcal{A}_i is consistent with the \mathcal{T} . Let π_1, \dots, π_n be the set of possibility distributions associated with $\mathcal{K}_1, \dots, \mathcal{K}_n$ where each $\mathcal{K}_i = \langle \mathcal{T}, \mathcal{A}_i \rangle$. Namely each π -DL-Lite knowledge base \mathcal{K}_i is associated with a possibility distribution π_i which is its semantic counterpart.

In this section, we investigate fusion of weighted π -DL-Lite assertional bases at semantic level. We show that such merging operation is the natural semantic counterpart of the Λ merging operators (presented in 4.3.1) used to merge π -DL-Lite ABox $\mathcal{A}_1, \dots, \mathcal{A}_n$ with respect to a \mathcal{T} .

More formally, given (π_1, \dots, π_n) possibility distributions associated with $(\mathcal{K}_1, \dots, \mathcal{K}_n)$ π -DL-Lite knowledge bases, then for the proposed operator Λ applied to aggregate $\mathcal{A}_1, \dots, \mathcal{A}_n$ with respect to \mathcal{T} , we look for a π -DL-Lite possibility distribution π_Λ constructed from the aggregation of (π_1, \dots, π_n) with the semantic counterpart of Λ that corresponds to the possibility distribution $\pi_{\mathcal{K}_\Lambda}$ induced from \mathcal{K}_Λ . Namely $\pi_\Lambda = \pi_{\mathcal{K}_\Lambda}$.

As usual, assume that π_1, \dots, π_n share the same domain of interpretations (namely $\Delta_1^{\mathcal{T}} = \dots = \Delta_n^{\mathcal{T}}$), and that all possibility distributions use the same scale to represents uncertainty. The following definition introduces the semantic definition of conflict vectors.

Definition 4.10. Let $\mathcal{A}_1, \dots, \mathcal{A}_n$ be a set of ABox and π_1, \dots, π_n be a the set of possibility distributions induced from $\mathcal{K}_1, \dots, \mathcal{K}_n$ where each $\mathcal{K}_i = \langle \mathcal{T}, \mathcal{A}_i \rangle$. Then $\forall f \in \mathcal{A}_i$ with $f = (\varphi, \alpha)$, we define semantically a conflict vector, denoted by $\mathcal{V}(f)$, as follows:

$$\mathcal{V}(f) = \langle \Pi_{\pi_1}(\varphi), \Pi_{\pi_2}(\varphi), \dots, \Pi_{\pi_n}(\varphi) \rangle$$

where $\forall i = 1..n: \Pi_{\pi_i}(f)$ denotes the possibility measure of φ induced from the possibility distribution π_i

Intuitively, a conflict vector associated to any ABox assertion represents to what extent this latter is compatible with available knowledge provided by each source.

Example 4.13. [Examples continued] Assuming that $\Delta^{\mathcal{T}} = \{a, b, c\}$, let us consider the following possibility distributions π_1, π_2 and π_3 to be merged. Note that we only have considered interpretations that are models of \mathcal{T} .

\mathcal{I}	\mathcal{I}	π_1	π_2	π_3
\mathcal{I}_1	$A = \{a\}, B = \{a\}, C = \{b, c\}$	1	.2	.5
\mathcal{I}_2	$A = \{b\}, B = \{b\}, C = \{a, c\}$.4	1	.5
\mathcal{I}_3	$A = \{c\}, B = \{c\}, C = \{a, b\}$.4	.2	.8
\mathcal{I}_4	$A = \{a, b\}, B = \{a, b\}, C = \{c\}$.5	.6	.5
\mathcal{I}_5	$A = \{a, c\}, B = \{a, c\}, C = \{b\}$	1	.2	.8
\mathcal{I}_6	$A = \{b, c\}, B = \{b, c\}, C = \{a\}$.4	1	1
\mathcal{I}_7	$A = \{a, b, c\}, B = \{a, b, c\}, C = \{\}$.5	.6	1
\mathcal{I}_8	$A = \{\}, B = \{\}, C = \{a, b, c\}$.4	.2	.5

Table 4.3: Possibility distributions induced from three knowledge bases

One can compute the following conflict vectors for each assertion:

$$\mathcal{V}(A(a)) = \langle \max(1, .5, 1, 1), \max(.2, .6, .2, .6), \max(.5, .6, .8, 1) \rangle = \langle 1, .6, 1 \rangle,$$

$$\mathcal{V}(A(b)) = \langle \max(.4, .5, .5, 5), \max(1, .6, .1, .6), \max(.5, .5, 1, 1) \rangle = \langle .5, 1, 1 \rangle,$$

$$\mathcal{V}(A(c)) = \langle \max(.4, 1, .4, .5), \max(.2, .2, 1, .6), \max(.8, .8, 1, 1) \rangle = \langle 1, 1, 1 \rangle,$$

$\mathcal{V}(B(b)) = \langle .5, 1, 1 \rangle$, $\mathcal{V}(B(c)) = \langle 1, 1, 1 \rangle$, $\mathcal{V}(C(a)) = \langle .4, 1, 1 \rangle$ and $\mathcal{V}(C(b)) = \langle 1, .2, .8 \rangle$ which are equal to the ones computed syntactically in Example 4.10. ■

Let us denote by Σ the collection of conflict vectors associated to each assertion of \mathcal{A}_i . Next definition introduces *min*-based assertional merging operator, denoted Λ , on the conflict vectors of Σ .

Definition 4.11. Let $\mathcal{A}_1, \dots, \mathcal{A}_n$ be a set of ABox and π_1, \dots, π_n be a the set of possibility distributions induced from $\mathcal{K}_1, \dots, \mathcal{K}_n$ where each $\mathcal{K}_i = \langle \mathcal{T}, \mathcal{A}_i \rangle$. Let Σ be the collection of conflict vectors associated to each assertion on \mathcal{A}_i computed using Definition 4.10. Then the *min*-based assertional merging operator, denoted by Λ , is defined on Σ as follows: $\forall \mathcal{V}(f) \in \Sigma: \mathcal{V}(f) = \langle \Pi_{\pi_1}(\varphi), \Pi_{\pi_2}(\varphi), \dots, \Pi_{\pi_n}(\varphi) \rangle$,

$$\Lambda(f) = \min\{\nu_i(f) \in \mathcal{V}(f)\}$$

Let us denote by Σ_Λ , the vector resulting by *min*-based aggregation of conflict vectors.

Example 4.14. [Example continued] One can compute the set Σ_Λ as follow: $\Lambda((A(a), .6)) = .6$, $\Lambda((A(b), .7)) = .5$, $\Lambda((A(b), .2)) = .5$, $\Lambda((A(c), .5)) = 1$, $\Lambda((B(b), .8)) = .5$, $\Lambda((B(c), .4)) = 1$, $\Lambda((C(a), .4)) = .4$ and $\Lambda((C(b), .5)) = .2$. ■

From Definition 4.11, one can associate to each assertion a new weight that represents its compatibility with other assertions provided by the other sources.

Definition 4.12. Let $\mathcal{A}_1, \dots, \mathcal{A}_n$ be a set of ABox and π_1, \dots, π_n be a the set of possibility distributions induced from $\mathcal{K}_1, \dots, \mathcal{K}_n$ where each $\mathcal{K}_i = \langle \mathcal{T}, \mathcal{A}_i \rangle$. Then the possibility distribution π_Λ as follows:

$$\forall I \in \Omega : \pi_\Lambda(I) = \begin{cases} 1 & \text{if } \forall (\varphi, \alpha) \in \mathcal{A}_i, I \models \varphi \\ 1 - \max\{\Lambda((\varphi, \alpha)) : (\varphi, \alpha) \in \mathcal{A}_i, \text{ and } I \not\models \varphi\} & \text{otherwise} \end{cases}$$

where $\Lambda(\varphi_i)$ is the compatibility measure of φ_i computed using definition 4.11

Example 4.15 (Example continued). We have $(A(c), .1)$, $(B(c), 1)$, $(A(a), .6)$, $(A(b), .5)$, $(B(b), .5)$, $(C(a), .4)$, $(C(b), .2)$. Then:

\mathcal{I}	\mathcal{I}_1	\mathcal{I}_2	\mathcal{I}_3	\mathcal{I}_4	\mathcal{I}_5	\mathcal{I}_6	\mathcal{I}_7	\mathcal{I}_8
π_Λ	0	0	.4	0	.5	.4	.6	0

 Table 4.4: Possibility distribution resulting from assertional *min*-based merging.

■

One can check that merging normalized possibility distributions may lead to a sub-normalized possibility distribution. This is the case with our example. Indeed, we focus on the normalization problem when the use of *min*-based assertional operators *min* provides a subnormal possibility distribution.

Definition 4.13. Let us consider: $h(\pi_\Lambda) = \max_{I \in \Omega} \{\pi_\Lambda(I)\}$. Then for every $I \in \Omega$ and $h(\pi_\Lambda) > 0$,

$$\pi_{N\Lambda}(I) = \begin{cases} 1 & \text{if } \pi_\Lambda(I) = h(\pi_\Lambda) \\ \pi_\Lambda(I) & \text{otherwise} \end{cases}$$

Example 4.16 (continued). From previous Example, we have:

\mathcal{I}	\mathcal{I}_1	\mathcal{I}_2	\mathcal{I}_3	\mathcal{I}_4	\mathcal{I}_5	\mathcal{I}_6	\mathcal{I}_7	\mathcal{I}_8
π_Λ	0	0	.4	0	.5	.4	.6	0
$\pi_{N\Lambda}$	0	0	.4	0	.5	.4	1	0

Table 4.5: Normalized possibility distribution resulting from assertional *min*-based merging

■

The following proposition states the equivalence between the semantic and syntactic approaches.

Proposition 4.4. Let $\mathcal{A}_1, \dots, \mathcal{A}_n$ be a set of ABox and π_1, \dots, π_n be a the set of possibility distributions induced from $\mathcal{K}_1, \dots, \mathcal{K}_n$ where each $\mathcal{K}_i = \langle \mathcal{T}, \mathcal{A}_i \rangle$. Then the possibility distribution

$$\pi_{N\Lambda}(I) = \begin{cases} 1 & \text{if } \pi_\Lambda(I) = h(\pi_\Lambda) \\ \pi_\Lambda(I) & \text{otherwise} \end{cases}$$

is associated with

$$\mathcal{K}_\Lambda = \langle \mathcal{T}, \{(\varphi, \Lambda(f)) : (\varphi, \Lambda(f)) \in \mathcal{A}_\Lambda \text{ and } \Lambda(f) > x\} \rangle$$

4.3.3 Logical properties

Let us use $E = \{\mathcal{K}_1, \dots, \mathcal{K}_n\}$ to denote a multi-set, called belief profile, that represents the knowledge bases to be merged (where each \mathcal{K}_i is associated with a possibility distribution π_i). Let us use Δ to denote a merging operator. This merging operator can be parametrized by an integrity constraint, being a knowledge base \mathcal{K} , and $\Delta_{\mathcal{K}}(E)$ denotes the result of the merging operator under this constraint \mathcal{K} . A logical characterization of integrity constraint merging operators has been proposed in [Konieczny and Pino Pérez, 2002] through a set of rational postulates extended from the ones proposed for belief revision [Katsuno and Mendelzon, 1991]. The following postulates rephrase the ones proposed in [Konieczny and Pino Pérez, 2002] within *DL-Lite* framework.

$$(M_0^\pi) \quad \Delta_{\mathcal{K}}(E) \models \mathcal{K}$$

$$(M_1^\pi) \quad \text{if } \mathcal{K} \text{ is consistent, then } \Delta_{\mathcal{K}}(E) \text{ is consistent}$$

$$(M_2^\pi) \quad \text{if } \mathcal{K} \cup \bigcup_{\mathcal{K}_i \in E} \mathcal{K}_i \text{ is consistent, then } \Delta_{\mathcal{K}}(E) = \mathcal{K} \cup \bigcup_{\mathcal{K}_i \in E} \mathcal{K}_i$$

$$(M_3^\pi) \quad \text{if } E_1 \approx E_2 \text{ and } \mathcal{K}_1 \equiv \mathcal{K}_2, \text{ then } \Delta_{\mathcal{K}_1}(E_1) \equiv \Delta_{\mathcal{K}_2}(E_2).$$

(\mathbf{M}_4^π) if $\mathcal{K}_1 \models \mathcal{K}$ and $\mathcal{K}_2 \models \mathcal{K}$, then $\Delta_{\mathcal{K}}(\mathcal{K}_1 \cup \mathcal{K}_2)$ is consistent implies that $\Delta_{\mathcal{K}}(\mathcal{K}_1 \cup \mathcal{K}_2) \cup \mathcal{K}_2$ is consistent

(\mathbf{M}_5^π) $\Delta_{\mathcal{K}}(E_1) \cup \Delta_{\mathcal{K}}(E_2) \models \Delta_{\mathcal{K}}(E_1 \uplus E_2)$

(\mathbf{M}_6^π) if $\Delta_{\mathcal{K}}(E_1) \cup \Delta_{\mathcal{K}}(E_2)$ is consistent, then $\Delta_{\mathcal{K}}(E_1 \uplus E_2) \models \Delta_{\mathcal{K}}(E_1) \cup \Delta_{\mathcal{K}}(E_2)$

(\mathbf{M}_7^π) $\Delta_{\mathcal{K}}(E) \cup \mathcal{K}' \models \Delta_{\mathcal{K} \cup \mathcal{K}'}(E)$

(\mathbf{M}_8^π) if $\Delta_{\mathcal{K}}(E) \cup \mathcal{K}'$ is consistent, then $\Delta_{\mathcal{K} \cup \mathcal{K}'}(E) \models \Delta_{\mathcal{K}}(E) \cup \mathcal{K}'$

($\mathbf{M}_{\text{maj}}^\pi$) $\exists n \Delta_{\mathcal{K}}(E_1 \uplus E_2^n) \models \Delta_{\mathcal{K}}(E_2)$

(\mathbf{M}_I^π) $\forall n \Delta_{\mathcal{K}}(E_1 \uplus E_2^n) \equiv \Delta_{\mathcal{K}}(E_1 \uplus E_2)$

With:

1. $\mathcal{K}_1 \models \mathcal{K}_2$ if and only if $\arg \max_I \pi_{\mathcal{K}_1}(I) \subseteq \arg \max_I \pi_{\mathcal{K}_2}(I)$
2. $\mathcal{K}_1 \equiv \mathcal{K}_2$ if and only if $\mathcal{K}_1 \models \mathcal{K}_2$ and $\mathcal{K}_2 \models \mathcal{K}_1$
3. $E_1 \approx E_2$ if and only if there exists a bijection g from E_1 to E_2 such that $\forall \mathcal{K} \in E_1 : \pi_{\mathcal{K}} = \pi_{g(\mathcal{K})}$
4. \uplus is the union of multisets [Knuth, 1998]
5. $E^n = \underbrace{E \uplus \dots \uplus E}_{n \text{ times}}$

Note that in the special case where we only consider only one TBox \mathcal{T}_1 for E , these postulates are equivalent with the ones proposed in [Qi *et al.*, 2006b], by considering the revision of \mathcal{T}_1 by the shared TBox \mathcal{T} . Hence, our postulates extend (with very few adaptations) the notion of Revision of [Qi *et al.*, 2006b].

For the assertional-based merging operator considered in the present chapter, the integrity constraint is $\mathcal{K} = \langle \mathcal{T}, \emptyset \rangle$ where \mathcal{T} is the set of TBox axioms of each $\mathcal{K}_i \in E$ and $\mathcal{K}_i = \langle \mathcal{T}, \mathcal{A}_i \rangle$.

Finally, one can check that the min-based assertional merging operators satisfies (\mathbf{M}_0^π), (\mathbf{M}_1^π), (\mathbf{M}_2^π), (\mathbf{M}_3^π), (\mathbf{M}_5^π), (\mathbf{M}_6^π), (\mathbf{M}_7^π), (\mathbf{M}_8^π), (\mathbf{M}_I^π) and falsifies (\mathbf{M}_4^π), ($\mathbf{M}_{\text{maj}}^\pi$).

4.4 Conditioning of possibilistic *DL-Lite* knowledge bases: Preliminary results

In this section, we first study conditioning of π -*DL-Lite* knowledge bases semantically by conditioning the possibility distribution associated to *DL-Lite* interpretations by the new information. We start by adapting the standard conditioning proposed in the possibilistic setting to the π -*DL-Lite* setting. We show in particular that conditioning the possibility distribution within *DL-Lite* differs from the one proposed by [Benferhat *et al.*, 2002c] within the standard possibilistic setting in the sense that a direct adaptation of conditioning to π -*DL-Lite* framework is not satisfactory. Roughly speaking, according to the interaction between the new information and the knowledge base, we identify situations where conditioning in *DL-Lite* differs from the one of the standard possibilistic setting. To this end, we study revision at syntactic level of π -*DL-Lite* knowledge bases. We propose two other definitions that generalize and refine the classical one.

4.4.1 Conditioning of DL-Lite possibility distributions

Let $\mathcal{K}=\langle \mathcal{T}, \mathcal{A} \rangle$ be a π -DL-Lite knowledge base where $\pi_{\mathcal{K}}$ is its joint possibility distribution computed according to Definition 3.2. For the sake of simplicity, we assume that \mathcal{K} is consistent (namely $\pi_{\mathcal{K}}$ is normalized).

Example 4.17. Let $\mathcal{K}=\langle \mathcal{T}, \mathcal{A} \rangle$ be a π -DL-Lite knowledge base where $\mathcal{T}=\{(A \sqsubseteq B, .4)\}$ and $\mathcal{A}=\{(A(a), .5), (C(a), .7)\}$. One can compute $\pi_{\mathcal{K}}$ the possibility distribution induced from \mathcal{K} using Definition 3.2.

\mathcal{I}	\mathcal{I}	$\pi_{\mathcal{K}}$
\mathcal{I}_1	$A = \{\}, B = \{\}, C = \{\}$.3
\mathcal{I}_2	$A = \{a\}, B = \{\}, C = \{\}$.3
\mathcal{I}_3	$A = \{\}, B = \{a\}, C = \{\}$.3
\mathcal{I}_4	$A = \{\}, B = \{\}, C = \{a\}$.5
\mathcal{I}_5	$A = \{a\}, B = \{a\}, C = \{\}$.3
\mathcal{I}_6	$A = \{a\}, B = \{\}, C = \{a\}$.6
\mathcal{I}_7	$A = \{\}, B = \{a\}, C = \{a\}$.5
\mathcal{I}_8	$A = \{a\}, B = \{a\}, C = \{a\}$	1

Table 4.6: Example of a possibility distribution $\pi_{\mathcal{K}}$ computed using Definition 3.2.

One can observe that $\pi_{\mathcal{K}}(\mathcal{I}_8)=1$ meaning that the knowledge base is consistent. Note that we have chosen a simple example in order to enumerate all interpretations. This will be helpful to illustrate the conditioning of a π -DL-Lite possibility distribution. ■

Let us denote by (φ, μ) the new information to be accepted. Within the π -DL-Lite setting, φ may be an assertion of the form $A(a)$ or $P(a, b)$, a positive inclusion axiom of the form $B_1 \sqsubseteq B_2$ or a negative inclusion axiom of the form $B_1 \sqsubseteq \neg B_2$ and $\mu \in]0, 1]$. The new input can be a totally reliable information (i.e. $\mu=1$) or uncertain (i.e. $0 < \mu < 1$). In π -DL-Lite, conditioning comes down to add the new information with its prescribed level of certainty while ensuring the consistency of the results.

In the following, we investigate conditioning at the semantic level. It consists in conditioning the original possibility distribution $\pi_{\mathcal{K}}$ by the new information (φ, μ) . This operation takes as input a possibility distribution $\pi_{\mathcal{K}}$ and the new information (φ, μ) and transforms $\pi_{\mathcal{K}}$ to a revised possibility distribution $\pi'=\pi_{\mathcal{K}}(\cdot | (\varphi, \mu))$. Here, the input (φ, μ) is considered as a constraint that must be satisfied in π' . More precisely, the revised distribution is such that $\Pi'(\varphi)=1$ (recall that in the possibilistic setting, in order for an event φ to have a certainty degree greater than zero, it must be totally possible, hence $\Pi'(\varphi)=1$, see Section 2.2.2) and $N'(\varphi) \geq \mu$ meaning that the axiom φ is certain at least to the degree μ . Here Π' (*resp.* N') is the possibility (*resp.* necessity) measure induced by the revised possibility distribution π' .

Logical properties

In [Benferhat *et al.*, 2002c], conditioning in the possibilistic logic setting is characterized with the following properties rephrased in our framework. A revised possibility distribution π' is considered eligible for revising the initial distribution $\pi_{\mathcal{K}}$ with the new input (φ, μ) if it satisfies the following properties.

$$(A1) \max_{\mathcal{I} \in \Omega} (\pi'(\mathcal{I})) = 1.$$

$$(A2) \Pi'(\varphi) = 1 \text{ and } N'(\varphi) \geq \mu.$$

$$(A3) \forall \mathcal{I}_1 \neq \varphi, \mathcal{I}_2 \neq \varphi, \text{ if } \pi_{\mathcal{K}}(\mathcal{I}_1) \leq \pi_{\mathcal{K}}(\mathcal{I}_2) \text{ then } \pi'(\mathcal{I}_1) \leq \pi'(\mathcal{I}_2).$$

(A4) $\forall \mathcal{I}_1 \models \varphi, \mathcal{I}_2 \models \varphi$, if $\pi_{\mathcal{K}}(\mathcal{I}_1) \leq \pi_{\mathcal{K}}(\mathcal{I}_2)$ then $\pi'(\mathcal{I}_1) \leq \pi'(\mathcal{I}_2)$.

(A5) If $N_{\mathcal{K}}(\varphi) > 0$ then $\forall \mathcal{I} \models \varphi$: $\pi_{\mathcal{K}}(\mathcal{I}) = \pi'(\mathcal{I})$

(A6) If $\pi_{\mathcal{K}}(\mathcal{I}) = 0$ then $\pi'(\mathcal{I}) = 0$.

Property **A1** ensures the consistency of the revised possibility distribution by guaranteeing a normalized distribution π' . **A2** guarantees that the added information should be inferred from the revised distribution π' with a weight at least equal to its prescribed priority level. **A3** ensures that the relative order between the interpretations that falsify φ is preserved. **A4** states that the new possibility distribution π' should preserve the previous pre-order between interpretations which are models of φ . **A5** means that the conditioning process does not affect models of φ when φ is a priori fully accepted. **A6** states that every impossible interpretation remains impossible after conditioning. In order to satisfy properties **A3** and **A4**, it is clear that the conditioning operation should condition both the interpretations satisfying φ and those falsifying φ .

According to properties **A1-A6**, two different types of possibility distribution conditioning when $\Pi(\varphi) > 0$ are proposed in [Dubois and Prade, 1988a], namely in an ordinal setting and in a quantitative setting. These conditionings are extended to the case where the new input is uncertain in [Dubois and Prade, 1997] and studied in [Benferhat et al., 2011]. In this chapter, we only focus on conditioning in the ordinal setting, well-known as min-based conditioning [Benferhat et al., 2002c].

Belief conditioning (or revision) with uncertain information was studied in many works and its close relation to Jeffrey's rule [Jeffrey, 1965] (generalizing probability theory's conditioning) is pointed out. In [Benferhat et al., 2002c] the possibilistic counterpart was given for belief revision with uncertain inputs when dealing with belief bases encoded in possibilistic logics (e.g. [Dubois et al., 1994]) The authors show that the conditioning process comes down syntactically to adding the new information with a prescribed level of certainty while maintaining the consistency of the resulting base and semantically to conditioning the possibility distribution representing the current epistemic state in order to add the new input.

Min-based π -DL-Lite possibility distribution conditioning

In order to define conditioning of possibility distribution $\pi_{\mathcal{K}}$, let us first recall that in standard propositional possibilistic logic, the necessity measure is the dual of the possibility measure and it is defined by $N(\phi) = 1 - \Pi(\neg\phi)$ where ϕ is a propositional formula. In possibilistic DL-Lite, a necessity measure cannot be defined as the dual of the possibility measure because the negation of an axiom in *DL-Lite* is not allowed (see Section 3.2).

The following definition rephrases conditioning within the possibilistic *DL-Lite* setting.

Definition 4.14. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a π -DL-Lite knowledge base and $\pi_{\mathcal{K}}$ be its joint possibility distribution. Let (φ, μ) be a new information. The min-based conditioning is extended to the π -DL-Lite setting as follows:

$$\begin{aligned} \bullet \forall \mathcal{I} \models \varphi, \pi_{\mathcal{K}}(\cdot |_{\mathcal{m}}(\varphi, \mu)) &= \begin{cases} 1 & \text{if } \pi_{\mathcal{K}}(\mathcal{I}) = \Pi(\varphi) \\ \pi(\mathcal{I}) & \text{otherwise} \end{cases} \\ \bullet \forall \mathcal{I} \not\models \varphi, \pi_{\mathcal{K}}(\cdot |_{\mathcal{m}}(\varphi, \mu)) &= \begin{cases} 1 - \mu & \text{if } \pi(\mathcal{I}) = \max\{\pi(\mathcal{I}) : \mathcal{I} \not\models \varphi\} \\ 1 - \mu & \text{if } \pi_{\mathcal{K}}(\mathcal{I}) > 1 - \mu \\ \pi(\mathcal{I}) & \text{otherwise} \end{cases} \end{aligned}$$

According to Definition 4.14, accepting the input consists in raising the degree of the most plausible model of φ to 1. This allows to deal only with axioms that are consistent with the input. For the counter-models, it is clear that the most plausible is set to $1-\mu$ and all the interpretations that are more compatible than $1-\mu$ should be shifted down to $1-\mu$.

Proposition 4.5. *Let $\mathcal{K}=\langle \mathcal{T}, \mathcal{A} \rangle$ be a π -DL-Lite knowledge base and $\pi_{\mathcal{K}}$ be its joint possibility distribution. Let (φ, μ) be a new information. Then $\pi'=\pi_{\mathcal{K}}(\cdot|(\varphi, \mu))$ computed using Definition 4.14 satisfies postulates (A1)-(A6).*

Example 4.18. Let us consider $\pi_{\mathcal{K}}$ presented in Example 4.17. Assume that we have in this example separately two cases of new information pieces to be accepted. The first one is $(B \sqsubseteq \neg C, .9)$ and the second one is $(B \sqsubseteq \neg C, .2)$. Using Definition 4.14, the min-based revised possibility distribution $\pi' = \pi_{\mathcal{K}}(\mathcal{I}|_m(B \sqsubseteq \neg C, .9))$ (resp. $\pi' = \pi_{\mathcal{K}}(\mathcal{I}|_m(B \sqsubseteq \neg C, .2))$) is as follows:

\mathcal{I}	\mathcal{I}	$\pi_{\mathcal{K}}$	$\pi_{\mathcal{K}}(\mathcal{I} _m(B \sqsubseteq \neg C, .9))$	$\pi_{\mathcal{K}}(\mathcal{I} _m(B \sqsubseteq \neg C, .2))$
\mathcal{I}_1	$A = \{\}, B = \{\}, C = \{\}$.3	.3	.3
\mathcal{I}_2	$A = \{a\}, B = \{\}, C = \{\}$.3	.3	.3
\mathcal{I}_3	$A = \{\}, B = \{a\}, C = \{\}$.3	.3	.3
\mathcal{I}_4	$A = \{\}, B = \{\}, C = \{a\}$.5	.5	.5
\mathcal{I}_5	$A = \{a\}, B = \{a\}, C = \{\}$.3	.3	.3
\mathcal{I}_6	$A = \{a\}, B = \{\}, C = \{a\}$.6	1	1
\mathcal{I}_7	$A = \{\}, B = \{a\}, C = \{a\}$.5	.1	.5
\mathcal{I}_8	$A = \{a\}, B = \{a\}, C = \{a\}$	1	.1	.8

Table 4.7: Example of possibility distribution revised by two information pieces.

In this example, the first scenario is revising $\pi_{\mathcal{K}}$ associated to \mathcal{K} with the input $(B \sqsubseteq \neg C, .9)$. Given that in $\pi_{\mathcal{K}}$, we have a priori $\Pi(B \sqsubseteq \neg C)=.6$ (hence it's necessity is 0) then the new input requires to be satisfied to increase the necessity of the axiom $B \sqsubseteq \neg C$ until .9.

In the second scenario, the necessity of the axiom $B \sqsubseteq \neg C$ has to be shifted down to .2. One can observe in $\pi_{\mathcal{K}}$ that the interpretations $\{\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4, \mathcal{I}_5, \mathcal{I}_6\} \models B \sqsubseteq \neg C$ where $\Pi(B \sqsubseteq \neg C) = .6$ while $\{\mathcal{I}_7, \mathcal{I}_8\} \not\models B \sqsubseteq \neg C$ where $\max\{\pi(\mathcal{I}) : \mathcal{I} \not\models B \sqsubseteq \neg C\} = 1$. ■

Definition 4.14 is a direct adaptation of conditioning in possibilistic logic [Dubois and Prade, 1988a] to π -DL-Lite framework. As it will be shown in the following example, conditioning of Definition 4.14 is not satisfactory as it provides somehow counterintuitive results. More precisely, conditioning of Definition 4.14 works when the new information is inconsistent with the knowledge base or it is a priori inferred with a weight less than its prescribed level μ . Hence conditioning here consists in simply adding the new information to the old knowledge (it is a kind of knowledge expansion). However, conditioning of Definition 4.14 does not work properly when the input is a priori inferred with a weight greater than its prescribed level μ . The following example illustrates this situation.

Example 4.19. Assume that we have a π -DL-Lite knowledge base \mathcal{K} where the TBox $\mathcal{T}=\{(A \sqsubseteq B, .4), (B \sqsubseteq C, .7)\}$ and the ABox $\mathcal{A}=\{(A(a), .3)\}$. One can easily check that we have a priori $\mathcal{K} \models_{\pi} (A \sqsubseteq C, .4)$ (indeed, as it is shown in Table 4.8, the axiom $A \sqsubseteq C$ has a necessity degree of .4 in the possibility distribution $\pi_{\mathcal{K}}$ associated to \mathcal{K}). Now assume the two following situations: In the first one, the information piece to be accepted by \mathcal{K} is $(A \sqsubseteq C, .9)$ while in the second situation \mathcal{K} is revised with $(A \sqsubseteq C, .2)$. Let $\pi_{\mathcal{K}}$ be the possibility distribution associated with \mathcal{K} and Let $\pi'=\pi_{\mathcal{K}}(\mathcal{I}|_m(A \sqsubseteq C, .9))$ (resp. $\pi''=\pi_{\mathcal{K}}(\mathcal{I}|_m(A \sqsubseteq C, .2))$) the conditioned min-based possibility distribution using Definition 4.14.

\mathcal{I}	\mathcal{I}	$\pi_{\mathcal{K}}$	$\pi' = \pi_{\mathcal{K}}(\mathcal{I} _m(A \sqsubseteq C, .9))$	$\pi'' = \pi_{\mathcal{K}}(\mathcal{I} _m(A \sqsubseteq C, .2))$
\mathcal{I}_1	$A=\{\}, B=\{a\}, C=\{\}$.3	.3	.3
\mathcal{I}_2	$A=\{a\}, B=\{\}, C=\{a\}$.6	.6	.6
\mathcal{I}_3	$A=\{\}, B=\{\}, C=\{\}$.7	.7	.7
\mathcal{I}_4	$A=\{\}, B=\{\}, C=\{a\}$.7	.7	.7
\mathcal{I}_5	$A=\{\}, B=\{a\}, C=\{a\}$.7	.7	.7
\mathcal{I}_6	$A=\{a\}, B=\{a\}, C=\{a\}$	1	1	1
\mathcal{I}_7	$A=\{a\}, B=\{\}, C=\{\}$.6	.1	.8
\mathcal{I}_8	$A=\{a\}, B=\{a\}, C=\{\}$.3	.1	.3

Table 4.8: Second example of possibility distribution conditioning by two information pieces.

The interpretations $\{\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4, \mathcal{I}_5, \mathcal{I}_6\}$ satisfy the input axiom $A \sqsubseteq C$ and we have a priori $\Pi(A \sqsubseteq C) = 1$ and $\Pi_n(A \sqsubseteq C) = .6$. The possibility degrees of the interpretations $\{\mathcal{I}_7, \mathcal{I}_8\}$ are set to $(1-.9) = .1$ in order to ensure that $N'(A \sqsubseteq C) = .9$. It is easy to check that properties **(A1)**-**(A6)** are satisfied by the distribution π' computed according to Definition 4.14. However when the input is $(A \sqsubseteq C, .2)$, there is a problem regarding the possibility degree associated to \mathcal{I}_2 in π'' . Indeed, we have $A \sqsubseteq C$ is implied by the fact $A \sqsubseteq B$ and $B \sqsubseteq C$. Hence, in order to have a necessity degree of $A \sqsubseteq C$ of .2 then one has to shift down at least the necessity degree of the axiom $A \sqsubseteq B$ down to .2 as it has a lower priority than $B \sqsubseteq C$. However, if the necessity of $A \sqsubseteq B$ is shifted down to .2 then the corresponding $\pi_{\mathcal{K}}$ after this modification will not be equivalent to the one given in Table 4.8. For instance, the interpretation \mathcal{I}_2 will be associated with a degree of .8 instead of .6 currently. Clearly revision with conditioning of Definition cannot fully capture syntactic revision detailed in the following section. ■

It is important to note that in the *DL-Lite* framework, it is not guaranteed that any set of interpretations represents a *DL-Lite* axiom (see Section 3.2).

In the next section, we analyze revision at syntactic level. We then provide a definition of conditioning possibility distributions that refines Definition 4.14.

4.4.2 Syntactic revision

In this section, we study revision with the new information (φ, μ) at the syntactic level. Revision here consists in obtaining from a π -*DL-Lite* knowledge base $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ associated to a possibility distribution $\pi_{\mathcal{K}}$ and an uncertain input information (φ, μ) , a new π -*DL-Lite* knowledge base $\mathcal{K}' = \langle \mathcal{T}', \mathcal{A}' \rangle$. As in possibilistic logic, in π -*DL-Lite*, revision comes down to add the new information with its prescribed level of certainty while ensuring the consistency of the revision results.

As mentioned in the previous section a simple adaptation of conditioning in possibilistic logic to π -*DL-Lite* framework is not very satisfactory. So, we identify cases where $\mathcal{K}' = \langle \mathcal{T}', \mathcal{A}' \rangle$ is associated to possibility $\pi_{\mathcal{K}'}$ such that:

$$\forall \mathcal{I} \in \Omega, \pi_{\mathcal{K}'}(\mathcal{I}) = \pi_{\mathcal{K}}(\mathcal{I}|_m(\varphi, \mu))$$

where $\pi_{\mathcal{K}'}$ is the possibility distribution obtained from conditioning $\pi_{\mathcal{K}}$ by (φ, μ) using minimum-based conditioning of Definition 4.14.

As illustrated in Example 4.19, Definition 4.14 doesn't provide good results and in particular when the (φ, μ) is inferred from the knowledge base with a weight greater than its prescribed one. Indeed, according the logical form of \mathcal{K}' , we propose an new definition of minimum-based conditioning that refine Definition 4.14.

When adding the new information to the knowledge base, several situations may be encountered, namely when the input is consistent or inconsistent with the the original knowledge. The following two subsections analyse these situations where they give formal representation of the revised knowledge base and re-define minimum-based conditioning in order to $\pi_{\mathcal{K}'}$ from $\pi_{\mathcal{K}}$.

The input (φ, μ) is inconsistent with the knowledge base

We address here the situation where the new information (φ, μ) is inconsistent with the knowledge base \mathcal{K} , namely $\Pi_{\mathcal{K}}(\varphi) < 1$ (recall that in possibility theory, if $\Pi(\varphi) < 1$ then $N(\varphi) = 0$). There are two situations to be considered. The first one is when (φ, μ) is implicitly inhibited by higher priority TBox or ABox axioms that contradict it. The second one is when (φ, μ) is not inhibited by higher priority axioms that contradict it. For these two cases, the construction of the augmented π -DL-Lite knowledge base \mathcal{K}' is performed according to the following steps:

1. Add the input φ to the knowledge base \mathcal{K} with the highest prescribed level (i.e. $\mu=1$).
2. Compute the inconsistency degree $\beta = \text{Inc}(\mathcal{K}_1)$ with $\mathcal{K}_1 = \mathcal{K} \cup \{(\varphi, 1)\}$.
3. Drop every axiom in \mathcal{K}_1 having a priority less than or equal to the inconsistency degree β . Let \mathcal{K}_2 the obtained consistent knowledge base.
4. Add φ with its prescribed level μ to \mathcal{K}_2 . Let $\mathcal{K}' = \mathcal{K}_2 \cup \{(\varphi, \mu)\}$.

These steps ensure the consistency of the resulting knowledge base after adding the input (φ, μ) with its prescribed level. The following proposition relates the resulting knowledge base \mathcal{K}' with the possibility distribution $\pi_{\mathcal{K}'}$ associated to \mathcal{K}' with the results of conditioning at the semantic level using Definition 4.14. namely $\pi_{\mathcal{K}'}(I) = \pi_{\mathcal{K}}(\mathcal{I}|_m(\varphi, \mu))$ using min-based conditioning defined in Definition 4.14.

Proposition 4.6. *Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a π -DL-Lite knowledge base and $\pi_{\mathcal{K}}$ be its joint possibility distribution. Let (φ, μ) be the added uncertain input information and $\beta = \text{Inc}(\mathcal{K}_1)$ where $\mathcal{K}_1 = \mathcal{K} \cup \{(\varphi, 1)\}$. Let $\mathcal{K}' = \langle \mathcal{T}', \mathcal{A}' \rangle$ such that $\mathcal{K}' = \{(\varphi, \mu)\} \cup \{(\phi, \alpha) : (\phi, \alpha) \in \mathcal{K} \text{ and } \alpha > \beta\}$ and let $\pi_{\mathcal{K}'}$ be the possibility distribution associated to \mathcal{K}' . Then,*

$$\forall I \in \Omega, \pi_{\mathcal{K}'}(I) = \pi_{\mathcal{K}}(\mathcal{I}|_m(\varphi, \mu)),$$

where $\pi_{\mathcal{K}}(\mathcal{I}|_m(\varphi, \mu))$ denotes the revised possibility distribution $\pi_{\mathcal{K}}$ computed using min-based conditioning defined in Definition 4.14.

Example 4.20 (examples 4.17 and 4.18 continued). We have $\mathcal{T} = \{(A \sqsubseteq B, .4)\}$ and $\mathcal{A} = \{(A(a), .5), (C(a), .7)\}$. Let us first assume a new input $(B \sqsubseteq \neg C, .9)$ and then another input $(B \sqsubseteq \neg C, .2)$. One can easily check that $\text{Inc}(\mathcal{K} \cup \{(B \sqsubseteq \neg C, 1)\}) = .4$. So, $(B \sqsubseteq \neg C, .2)$ (resp. $(B \sqsubseteq \neg C, .9)$) is inhibited (resp. not inhibited) by higher priority axioms that contradict it. For the first case, it is easy to check that $\mathcal{K}' = \{(B \sqsubseteq \neg C, .2), (A(a), .5), (C(a), .7)\}$ is such that $\pi_{\mathcal{K}'}(I) = \pi_{\mathcal{K}}(\mathcal{I}|_m(B \sqsubseteq \neg C, .2))$ presented in Example 4.18. For the second case however, $\mathcal{K}' = \{(B \sqsubseteq \neg C, .9), (A(a), .5), (C(a), .7)\}$ such that $\pi_{\mathcal{K}'}(I) = \pi_{\mathcal{K}}(\mathcal{I}|_m(B \sqsubseteq \neg C, .9))$ presented in Example 4.18. ■

The input (φ, μ) is consistent with the knowledge base

When the input (φ, μ) is consistent with the knowledge base \mathcal{K} (namely $\Pi(\varphi) = 1$), two situations are to be considered:

1. The first one is when (φ, μ) is a priori inferred from the knowledge base \mathcal{K} , namely $\mathcal{K} \models_{\pi} \varphi$.

2. The second one is when (φ, μ) cannot be inferred from \mathcal{K} , namely $\mathcal{K} \not\models_{\pi} \phi$. Here, revision is performed with a simple expansion of \mathcal{K} with the input (φ, μ) , namely $\mathcal{K}' = \mathcal{K} \cup (\varphi, \mu)$.

Let us first discuss the situation where the input (φ, μ) is a priori inferred from the knowledge base \mathcal{K} . In this situation, two scenarios can hold depending on the a priori necessity measure of φ (denoted $N(\varphi) = \nu$), and its prescribed posterior necessity $N'(\varphi) = \mu$. Namely:

1. When $\nu \leq \mu$ meaning that the new information is inferred with a certainty degree ν less than its prescribed one μ . Note that this situation is similar to the case of revising with a certain input (namely case where $\mu = 1$).
2. When $\nu > \mu$ meaning that the new information is inferred with a certainty degree ν that is greater than its prescribed one μ .

In the π -DL-Lite framework, two different kinds of inference services are given, namely flat subsumption (resp. instance checking) and weighted subsumption (resp. instance checking) (see Section 3.7). To determine to what extent the input (φ) is inferred from the knowledge base, namely $\mathcal{K} \models_{\pi} (\varphi, \nu)$ with $\nu \geq \mu$ or $\nu < \mu$, we first add to \mathcal{K} the assumption that φ is false encoded by the following statements: $\{(Y \sqsubseteq C_1, 1), (Y \sqsubseteq \neg C_2, 1), (Y(y), 1)\}$ if $\varphi = C_1 \sqsubseteq C_2$ and $\{(Y \sqsubseteq \neg C_1, 1), (Y(a), 1)\}$ if $\varphi = C_1(a)$ where Y (resp. y) is a new concept (resp. individual) not appearing in \mathcal{K} . Then we compute the inconsistency degree of the augmented knowledge base. This inconsistency degree corresponds to ν . Namely $\mathcal{K} \models_{\pi} (\varphi, \nu)$ if and only if $Inc(\mathcal{K}_1) = \nu$ where $\mathcal{K}_1 = \langle \mathcal{T}_1, \mathcal{A}_1 \rangle$ with $\mathcal{T}_1 = \mathcal{T} \cup \{(Y \sqsubseteq C_1, 1), (Y \sqsubseteq \neg C_2, 1)\}$ and $\mathcal{A}_1 = \{(Y(y), 1)\}$ or $\mathcal{T}_1 = \mathcal{T} \cup \{(Y \sqsubseteq \neg C_1, 1)\}$ and $\mathcal{A}_1 = \mathcal{A} \cup \{(Y(a), 1)\}$.

Now, the construction of the augmented π -DL-Lite knowledge base \mathcal{K}' is performed using the following steps:

1. Add the assumption that φ is false to \mathcal{K} with the highest prescribed level (i.e. $\mu = 1$).
2. Compute the inconsistency degree of the augmented knowledge base (i.e. $Inc(\mathcal{K}_1) = \nu$).
3. If $\mu \geq \nu$, then the revision outcome is $\mathcal{K}' = \mathcal{K} \cup \{(\varphi, \mu)\}$.
4. if $(\mu < \nu)$ two solutions can be proposed.
 - (a) The first one is to shift down the weights of axioms in \mathcal{K} which are between μ and ν to μ .
 - (b) The second solution is to compute first the set $\mathcal{X} \subseteq \mathcal{K}$ of axioms in \mathcal{K} that imply φ . Then we shift down the weights of axioms in \mathcal{X} which are between μ and ν to μ .

These steps ensure inferring the new input φ from the resulting knowledge base \mathcal{K}' with its prescribed level μ . Following these steps, it is clear that the revision process does not change the initial weights attached to axioms of \mathcal{K} if $\mathcal{K} \models_{\pi} (\varphi, \nu)$ with $\nu \leq \mu$. However it changes the initial weights attached to some axioms responsible or not for inferring φ from \mathcal{K} with the weight μ when $\nu > \mu$. According to the Example 4.19 presented in the previous section, conditioning proposed by Definition 4.14 is counterintuitive when $(\mu < \nu)$. To this end, we fit Definition 4.14 before giving the formal representation of \mathcal{K}' .

Semantic counterpart

Let us start with the case where $\nu > \mu$. The following definition gives a min-based conditioning of π -DL-Lite possibility distribution generalizing Definition 4.14.

Definition 4.15. Let $\mathcal{K}=\langle \mathcal{T}, \mathcal{A} \rangle$ be a π -DL-Lite knowledge base and $\pi_{\mathcal{K}}$ be its joint possibility distribution. Let (φ, μ) be the new information. The *min*-based conditioning is extended to the π -DL-Lite setting as follows:

$$\begin{aligned} \bullet \forall \mathcal{I} \models \varphi, \pi_{\mathcal{K}}(\cdot |_{\mathcal{I}}(\varphi, \mu)) &= \begin{cases} 1 & \text{if } \pi_{\mathcal{K}}(\mathcal{I}) = \Pi(\varphi) \\ 1 - \mu & \text{if } \max\{\pi(\mathcal{I}) : \mathcal{I} \not\models \varphi\} \leq \pi_{\mathcal{K}}(\mathcal{I}) \leq 1 - \mu \\ \pi(\mathcal{I}) & \text{otherwise} \end{cases} \\ \bullet \forall \mathcal{I} \not\models \varphi, \pi_{\mathcal{K}}(\cdot |_{\mathcal{I}}(\varphi, \mu)) &= \begin{cases} 1 - \mu & \text{if } \pi(\mathcal{I}) = \max\{\pi(\mathcal{I}) : \mathcal{I} \not\models \varphi\} \text{ or } \pi_{\mathcal{K}}(\mathcal{I}) > 1 - \mu \\ \pi(\mathcal{I}) & \text{otherwise} \end{cases} \end{aligned}$$

According to Definition 4.15, accepting the input consists in raising the degree of the most plausible model of φ to 1. Moreover when $N(\varphi) \geq \mu$, some models of φ will all be set to $1 - \mu$. For the counter-models, the most plausible is set to $1 - \mu$ and all interpretations that are more compatible than $1 - \mu$ should be shifted down to $1 - \mu$. Moreover, when $N(\varphi) = \nu > \mu$ the interpretations that falsify less priority axioms inferring φ will be revised.

Proposition 4.7. Let $\mathcal{K}=\langle \mathcal{T}, \mathcal{A} \rangle$ be a π -DL-Lite knowledge base and $\pi_{\mathcal{K}}$ be its joint possibility distribution. Let (φ, μ) be the new information. If $\Pi(\varphi) < 1$, Then $\pi' = \pi_{\mathcal{K}}(\cdot |_{\mathcal{I}}(\varphi, \mu))$ computed using Definition 4.15 satisfies postulates (A1), (A2), (A3), (A4), (A6).

The following proposition relates the resulting knowledge base \mathcal{K}' with the possibility distribution $\pi_{\mathcal{K}'}$ associated to \mathcal{K}' with the results of conditioning at the semantic level using Definition 4.15

Proposition 4.8. Let $\mathcal{K}=\langle \mathcal{T}, \mathcal{A} \rangle$ be a π -DL-Lite knowledge base and $\pi_{\mathcal{K}}$ be its joint possibility distribution. Let (φ, μ) be the added uncertain input information and $\nu = \text{Inc}(\mathcal{K}_1)$ where \mathcal{K}_1 is the augmented knowledge base by the assumption that φ is false. Then the revised π -DL-Lite knowledge base $\mathcal{K}'=\langle \mathcal{T}', \mathcal{A}' \rangle$ such that:

$$\mathcal{K}' = \{(\varphi, \mu)\} \cup \{(\phi, \alpha) : (\phi, \alpha) \in \mathcal{K} \text{ and } \alpha > \nu\} \cup \{(\phi, \alpha) : (\phi, \alpha) \in \mathcal{K} \text{ and } \alpha < \mu\} \cup \{(\phi, \mu) : (\phi, \alpha) \in \mathcal{K} \text{ and } \mu \leq \alpha \leq \nu\}$$

The possibility distribution $\pi_{\mathcal{K}'}$ associated to \mathcal{K}' is such that:

$$\forall I \in \Omega, \pi_{\mathcal{K}'}(I) = \pi_{\mathcal{K}}(\mathcal{I} |_{\mathcal{I}}(\varphi, \mu)),$$

where $\pi_{\mathcal{K}}(\mathcal{I} |_{\mathcal{I}}(\varphi, \mu))$ denotes the revised possibility distribution of $\pi_{\mathcal{K}}$ using the *min*-based conditioning of Definition 4.15.

Proposition 4.8 leads to shift down the weights of axioms in \mathcal{K} which are between μ and ν to μ . However, one can improve the result with a minimal change consisting in revising only the weights of some axioms responsible of implying the new information.

Example 4.21 (Examples 4.19 continued). We have $\mathcal{T}=\{(A \sqsubseteq B, .4), (B \sqsubseteq C, .7)\}$ and $\mathcal{A}=\{(A(a), .3)\}$. Let us consider $(A \sqsubseteq C, .9)$ and $(A \sqsubseteq C, .2)$. One can easily check that $\text{Inc}(\mathcal{K}_1) = .4$ where $\mathcal{K}_1 = \langle \mathcal{T} \cup \{(Y \sqsubseteq A, 1), (Y \sqsubseteq \neg C, 1)\}, \{(Y(y), 1)\} \rangle$. So $\mathcal{K} \models_{\pi} (A \sqsubseteq C, .4)$. When the input is $(A \sqsubseteq C, .9)$, then $\mathcal{K}' = \{(A \sqsubseteq B, .4), (B \sqsubseteq C, .7), (A \sqsubseteq C, .9), (A(a), .3)\}$ such that $\pi_{\mathcal{K}'}(\mathcal{I}) = \pi_{\mathcal{K}}(\mathcal{I} |_{\mathcal{I}}(A \sqsubseteq C, .9))$ presented in Example 4.19. Now, when the input is $(A \sqsubseteq C, .2)$, then $\mathcal{K}' = \{(A \sqsubseteq B, .2), (B \sqsubseteq C, .7), (A \sqsubseteq C, .2), (A(a), .2)\}$ such that $\pi_{\mathcal{K}'}(\mathcal{I}) = \pi_{\mathcal{K}}(\mathcal{I} |_{\mathcal{I}}(A \sqsubseteq C, .2))$ presented in Example 4.19 becomes as follows:

\mathcal{I}	\mathcal{I}	$\pi_{\mathcal{K}}$	$\pi' = \pi_{\mathcal{K}}(\mathcal{I} _m(A \sqsubseteq C, .9))$	$\pi'' = \pi_{\mathcal{K}}(\mathcal{I} _m(A \sqsubseteq C, .2))$
\mathcal{I}_1	$A = \{\}, B = \{a\}, C = \{\}$.3	.3	.3
\mathcal{I}_2	$A = \{a\}, B = \{\}, C = \{a\}$.6	.6	.8
\mathcal{I}_3	$A = \{\}, B = \{\}, C = \{\}$.7	.7	.8
\mathcal{I}_4	$A = \{\}, B = \{\}, C = \{a\}$.7	.7	.8
\mathcal{I}_5	$A = \{\}, B = \{a\}, C = \{a\}$.7	.7	.8
\mathcal{I}_6	$A = \{a\}, B = \{a\}, C = \{a\}$	1	1	1
\mathcal{I}_7	$A = \{a\}, B = \{\}, C = \{\}$.6	.1	.8
\mathcal{I}_8	$A = \{a\}, B = \{a\}, C = \{\}$.3	.1	.3

Table 4.9: Example of possibility distribution revision by two information pieces using Definition 4.15

■

Given the set $\mathcal{X} \subseteq \mathcal{K}$ of axioms in \mathcal{K} that infer φ , we distinguish semantically four sets of interpretations when the new information φ is satisfied:

1. Interpretations that are models of \mathcal{X} and $\mathcal{K} - \mathcal{X}$,
2. Interpretations that are models of \mathcal{X} but are not models of $\mathcal{K} - \mathcal{X}$,
3. Interpretations that are models of $\mathcal{K} - \mathcal{X}$ but are not models of \mathcal{X} and
4. Interpretations that are neither models of $\mathcal{K} - \mathcal{X}$ nor \mathcal{X} .

The following definition provides another min-based conditioning of π -DL-Lite possibility distribution that also adapts Definition 4.14.

Definition 4.16. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a π -DL-Lite knowledge base and $\pi_{\mathcal{K}}$ be its joint possibility distribution. Let (φ, μ) be the new information. Let $\mathcal{X} \subseteq \mathcal{K}$ be the set of axioms inferring φ . Let

$$\mu' = \max\{\alpha : (\phi, \alpha) \in \mathcal{K} - \mathcal{X} \text{ and } \mathcal{I} \not\models \phi\}.$$

In an ordinal setting, we define the min-based conditioning as follows:

- $\forall \mathcal{I} \models (\varphi \cup \mathcal{X}), \pi(\cdot|_m(\varphi, \mu)) = \begin{cases} 1 & \text{if } \pi(\mathcal{I}) = \Pi(\varphi) \\ \pi(\mathcal{I}) & \text{otherwise} \end{cases}$
- $\forall \mathcal{I} \models \varphi \cup (\mathcal{K} - \mathcal{X}), \mathcal{I} \not\models \mathcal{X}, \pi(\cdot|_m(\varphi, \mu)) = \begin{cases} 1 - \mu & \text{if } \pi(\mathcal{I}) = \max\{\pi(\mathcal{I}) : \mathcal{I} \not\models \varphi\} \\ \pi(\mathcal{I}) & \text{otherwise} \end{cases}$
- $\forall \mathcal{I} \models \varphi, \mathcal{I} \not\models \mathcal{X}, \mathcal{I} \not\models \mathcal{K} - \mathcal{X}, \pi(\cdot|_m(\varphi, \mu)) = \begin{cases} 1 - \mu & \text{if } \pi(\mathcal{I}) = \max\{\pi(\mathcal{I}) : \mathcal{I} \not\models \varphi\} \text{ and } 1 - \mu' \geq 1 - \mu \\ 1 - \mu' & \text{if } \pi(\mathcal{I}) = \max\{\pi(\mathcal{I}) : \mathcal{I} \not\models \varphi\} \text{ and } 1 - \mu' \leq 1 - \mu \\ \pi(\mathcal{I}) & \text{otherwise} \end{cases}$
- $\forall \mathcal{I} \not\models \varphi, \pi(\cdot|_m(\varphi, \mu)) = \begin{cases} 1 - \mu & \text{if } \pi(\mathcal{I}) = \max\{\pi(\mathcal{I}) : \mathcal{I} \not\models \varphi\} \\ 1 - \mu & \text{if } \pi(\mathcal{I}) > 1 - \mu \\ \pi(\mathcal{I}) & \text{otherwise} \end{cases}$

Proposition 4.9. *Let $\mathcal{K}=\langle \mathcal{T}, \mathcal{A} \rangle$ be a π -DL-Lite knowledge base and $\pi_{\mathcal{K}}$ be its joint possibility distribution. Let (φ, μ) be the new information. If $\Pi(\varphi) < 1$. Then $\pi' = \pi_{\mathcal{K}}(\cdot | (\varphi, \mu))$ computed using Definition 4.16 satisfies postulates (A1), (A2), (A3) and (A6).*

The following proposition relates the resulting knowledge base \mathcal{K}' with the possibility distribution $\pi_{\mathcal{K}'}$ associated to \mathcal{K}' with the results of conditioning at the semantic level using Definition 4.16.

Proposition 4.10. *Let $\mathcal{K}=\langle \mathcal{T}, \mathcal{A} \rangle$ be a π -DL-Lite knowledge base and $\pi_{\mathcal{K}}$ be its joint possibility distribution. Let (φ, μ) be the added uncertain input information and $\nu = Inc(\mathcal{K}_1)$ where \mathcal{K}_1 is the augmented knowledge base by the assumption that φ is false. Then the revised π -DL-Lite knowledge base $\mathcal{K}' = \langle \mathcal{T}', \mathcal{A}' \rangle$ such that*

$$\mathcal{K}' = \{(\varphi, \mu)\} \cup \{\mathcal{K} - \mathcal{X}\} \cup \{(\phi, \alpha) : (\phi, \alpha) \in \mathcal{X} \text{ and } \alpha > \nu\} \cup \{(\phi, \mu) : (\phi, \nu) \in \mathcal{X} \text{ and } \nu = \alpha\}$$

The possibility distribution $\pi_{\mathcal{K}'}$ associated to \mathcal{K}' is such that:

$$\forall I \in \Omega, \pi_{\mathcal{K}'}(I) = \pi_{\mathcal{K}}(\mathcal{I}|_m(\varphi, \mu)),$$

where $\pi_{\mathcal{K}}(\mathcal{I}|_m(\varphi, \mu))$ denotes the revised possibility distribution of $\pi_{\mathcal{K}}$ using the min-based conditioning defined in Definition 4.16.

Example 4.22 (Examples 4.19 continued). When the input is $(A \sqsubseteq C, .9)$, then $\mathcal{K}' = \{(A \sqsubseteq B, .4), (B \sqsubseteq C, .7), (A \sqsubseteq C, .9), (A(a), .3)\}$ such that $\pi_{\mathcal{K}'}(I) = \pi_{\mathcal{K}}(\mathcal{I}|_m(A \sqsubseteq C, .9))$ presented in Example 4.19. Now, when the input is $(A \sqsubseteq C, .2)$, then $\mathcal{K}' = \{(A \sqsubseteq B, .2), (B \sqsubseteq C, .7), (A \sqsubseteq C, .9), (A(a), .3)\}$ such that $\pi_{\mathcal{K}'}(I) = \pi_{\mathcal{K}}(\mathcal{I}|_m(A \sqsubseteq C, .2))$ becomes as follows:

\mathcal{I}	\mathcal{I}	$\pi_{\mathcal{K}}$	$\pi' = \pi_{\mathcal{K}}(\mathcal{I} _m(A \sqsubseteq C, .9))$	$\pi'' = \pi_{\mathcal{K}}(\mathcal{I} _m(A \sqsubseteq C, .2))$
\mathcal{I}_1	$A = \{\}, B = \{a\}, C = \{\}$.3	.3	.3
\mathcal{I}_2	$A = \{a\}, B = \{\}, C = \{a\}$.6	.6	.8
\mathcal{I}_3	$A = \{\}, B = \{\}, C = \{\}$.7	.7	.7
\mathcal{I}_4	$A = \{\}, B = \{\}, C = \{a\}$.7	.7	.7
\mathcal{I}_5	$A = \{\}, B = \{a\}, C = \{a\}$.7	.7	.7
\mathcal{I}_6	$A = \{a\}, B = \{a\}, C = \{a\}$	1	1	1
\mathcal{I}_7	$A = \{a\}, B = \{\}, C = \{\}$.6	.1	.8
\mathcal{I}_8	$A = \{a\}, B = \{a\}, C = \{\}$.3	.1	.3

Table 4.10: Example of possibility distribution revision by two information pieces using Definition 4.16

Let us now discuss the case where $\mu \geq \nu$. It is similar to the revision by a totally reliable information (i.e. $\mu = 1$). In this case, it is natural that all the interpretations that are models of φ must be preserved and all the interpretations that falsify φ must be set as impossible (the necessity degree of the input equals 0). In this case the conditioning operation follows from Definitions 4.15 and 4.16. Moreover conditioning according Definitions 4.15 and 4.16 agrees with Definition 4.14. Finally when (φ, μ) cannot be inferred from \mathcal{K} , this means that the revision process is performed simply with an expansion of \mathcal{K} with the input. In such situation, conditioning follows trivially according to Definitions 4.15 and 4.16 and coincides with Definition 4.14. It is similar to the case where the input is inconsistent with \mathcal{K} . Clearly, $Inc(\mathcal{K} \cup (\varphi, \mu)) = 0$.

Discussions

According to the new definition, conditioning of π -DL-Lite possibility distribution with (φ, μ) establishes a new pre-order between counter-models and models of φ . This new ranking depends on the a priori necessity measure of φ , and the prescribed posterior necessity measure of φ . Roughly speaking, if $N(\varphi) \leq \mu$ then with a min-based conditioning every interpretation that falsifies φ and that is more compatible than $1-\mu$ is shifted down to $1-\mu$. This means that some a priori pre-order on these interpretations will be lost. Moreover, the fact that within π -DL-Lite framework, the necessity measure is not the dual of the possibility measure, some a priori pre-order on interpretations which are models of φ will also be lost. This is a consequence of shifting down to $1-\mu$ some more compatible counter-models of φ when $N(\varphi) \leq \mu$. Regarding the computational complexity of the syntactic revision, it is obvious that it is polynomial since computing the inconsistency degree of a π -DL-Lite knowledge base is polynomial using the algorithm proposed in Section 3.5.

To compute the revision outcome, we need one step further when (φ, μ) is inferred from the knowledge base. Namely, we need to compute the set of axioms responsible for deducing the input. The computational complexity of this subset is also polynomial. This step is in the spirit of computing the π -negated closure of a π -DL-Lite knowledge base. Clearly, computing this subset X comes down by adding every axiom involved in computing the $\pi - neg(\mathcal{T}_1)$ starting only from the negative inclusion axiom added from the assumption that φ is false. This is for obtained TBox axioms. Obtaining ABox assertions comes down to detect all assertions in the original ABox that contradict negative inclusion in \mathcal{X} .

4.5 Conclusion

In this chapter, we first investigated merging of uncertain DL-Lite knowledge bases by adapting the min-based idempotent conjunctive operator. We then proposed a new min-based operator for merging multiple sources ABoxe's sharing the same terminology in the context of π -DL-Lite. We propose a syntactic version of this operator and its semantic counterpart. This operator turns out to be more productive than the classical one, without increasing the complexity of the merging process. In particular, it picks any piece of information that is not in contradiction with the other bases: it is not affected by the drowning effect. We finally provide an analysis in the light of a new set of postulates dedicated to uncertain DL-Lite merging.

This chapter addressed also conditioning of π -DL-Lite knowledge bases when a new piece of information (φ, μ) , possibly conflicting or uncertain, becomes available. We first studied revision at the semantic level by adapting conditioning of possibility distributions proposed within the possibilistic setting. We have shown that such conditioning may provide some counterintuitive results. We then investigated revision at the syntactic level of π -DL-Lite knowledge bases. Finally, we proposed two other definitions of π -DL-Lite possibility distribution s conditioning that generalize the first one.

From the works on merging, it seems that our merging approach based on conflict resolution is closely related to handling inconsistency approaches. In fact, among the crucial issues when merging is how to deal with conflicting information. This has led us to orientate our works towards inconsistency management. Unfortunately, even in the flat case, (when there is no priority between sources), only few works addressed this problem. To this end, the second part concerns the problem of inconsistency handling in flat DL-Lite knowledge bases.

PART III

Inconsistency handling in flat *DL-Lite* knowledge bases

NON-MERGE INCONSISTENCY MANAGEMENT ROADMAP IN FLAT *DL-Lite* KNOWLEDGE BASES

5.1 Introduction

In this chapter, we place ourselves in the context of handling an inconsistent set of pieces of information. As a case of study, we consider the setting of flat Ontology Based Data Access (OBDA) that studies how to query a set of independent data sources using an unified ontological view. A specific research challenge within the OBDA setting addresses the case when the data sources put together are inconsistent. Existing works (e.g. [Lembo and Ruzzi, 2007; Lembo *et al.*, 2010; Bienvenu and Rosati, 2013; Rosati, 2014]) have focused on the study of different inference strategies (called semantics) based on productivity criteria (how two semantics relate to each other based on their answer sets) and computational complexity. In these studies, closely related to works on consistent query answering from inconsistent databases (e.g. [Chomicki, 2007; Bertossi, 2011]), there is a lack of studies on how existing inference strategies can be placed within the space of possible inference strategies.

This chapter produces a *roadmap of different inconsistency management techniques* from a *DL-Lite* knowledge base with a multiple ABox, called an MBox *DL-Lite* knowledge base. An MBox is a multi-set of ABox's which can be issued from multiple information sources (as per the OBDA view) but could also, for instance, be resulted from revising inconsistent *DL-Lite* knowledge bases.

We build upon the state of the art and:

1. Introduce, discuss and provide properties for three main elementary changes or modifiers that can be operated on an MBox, namely expansion modifiers, splitting modifiers and selection-based modifiers.
2. Provide and study different inference strategies for query answering from MBox *DL-Lite* knowledge bases.
3. Show how the combination of modifiers and inference strategies provides a natural general setting that extends existing consistent query answering OBDA techniques.
4. Provide a study of productivity results for modifier plus inference strategies combinations.

Based on the above notions, an additional and important contribution in this chapter is providing a generalized view for handling inconsistent standard *DL-Lite* knowledge bases. The important points of this roadmap lay in its principled nature and completeness. Within this setting, the particular problem of repairing an inconsistent *DL-Lite* knowledge base can be seen as made out of a (1) composite or a complex modifier on a given MBox followed by (2) an inference-based strategy. We show that there are exactly eight major composite modifiers that can be applied on an inconsistent *DL-Lite* knowledge base and identify those that produce a single consistent and preferred repair.

The rest is organized as follows: Section 5.2 first defines the concept of *DL-Lite* knowledge bases with multiple ABox, and then introduces three elementary modifiers that, applied on MBox, define new modifiers. Section 5.3 presents several inference-based strategies that can be applied on an MBox *DL-Lite* knowledge base. Section 5.4 investigates the problem of repairing an inconsistent *DL-Lite* knowledge base, which is considered as composed of a composite modifier applied on a given MBox followed by an inference-based strategy. Section 5.5 studies the productivity and gives a complexity analysis of the different inconsistency-tolerant inferences. Section 5.6 discusses the different results. Section 5.7 concludes the chapter. Finally, additional propositions and counterexamples can be found in the appendix B.

5.2 Reasoning from MBox knowledge bases

This section proposes how to reason from a *DL-Lite* knowledge base with a multiple ABox, called an MBox *DL-Lite* knowledge base. We discuss three main elementary changes or modifiers on an MBox. We provide different ways to compose them in order to obtain a composite modifier and propose inference strategies for querying MBox *DL-Lite* knowledge bases.

5.2.1 MBox: Multiple ABox

We first introduce the concept of *DL-Lite* knowledge bases with multiple ABox, called MBox *DL-Lite* knowledge bases.

Definition 5.1. A *DL-Lite* knowledge base with an MBox, called MBox *DL-Lite* knowledge base, is of the form $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$ where \mathcal{T} is a standard *DL-Lite* TBox and $\mathcal{M} = \{\mathcal{A}_1, \dots, \mathcal{A}_n\}$ is a multi-set of facts, called MBox, where each \mathcal{A}_i is a standard *DL-Lite* ABox.

In the above definition, there is no additional assumption regarding the consistency of each $\langle \mathcal{T}, \mathcal{A}_i \rangle$. However, in general, $\langle \mathcal{T}, \mathcal{A}_i \rangle$'s are often assumed to be consistent while $\langle \mathcal{T}, \bigcup_{\mathcal{A}_i \in \mathcal{M}} \mathcal{A}_i \rangle$ is unlikely to be consistent. An MBox may be viewed as a convenient way to represent a multiple-sources of information, where each ABox \mathcal{A}_i is assumed to be provided by a distinct source. An MBox may also be the result of revising an inconsistent standard *DL-Lite* knowledge base. In this case, each element of the MBox reflects a possible repair of the inconsistent *DL-Lite* knowledge base. This view of an MBox will be assumed in large in Section 5.4.

In the rest of the chapter, an MBox *DL-Lite* knowledge base $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \{\mathcal{A}_1, \dots, \mathcal{A}_n\} \rangle$ is said to be consistent if each $\langle \mathcal{T}, \mathcal{A}_i \rangle$ is individually consistent. A standard knowledge base will be indifferently represented by $\langle \mathcal{T}, \mathcal{A} \rangle$ or by an MBox knowledge base $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} = \{\mathcal{A}\} \rangle$ with only a single ABox in \mathcal{M} .

Example 5.1. Let $\mathcal{T} = \{C \sqsubseteq B\}$ be a *DL-Lite* TBox. Let $\mathcal{M} = \{\{C(a), B(a)\}, \{B(c), B(a)\}\}$ and $\mathcal{M}' = \{\{C(a), B(a), B(c)\}\}$ be two MBox's. Then $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$ is a *DL-Lite* knowledge base with an MBox containing a set of ABox and $\mathcal{K}'_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M}' \rangle$ is a *DL-Lite* knowledge base with an MBox that is only composed with one single ABox. $\mathcal{K}'_{\mathcal{M}}$ encodes a standard *DL-Lite* knowledge base. ■

5.2.2 Elementary modifiers on MBox

We now introduce elementary modifiers that, applied on MBox, define new modifiers.

Expansion modifiers

The first considered modifier operators concern expansion of ABox's. It consists in adding to each ABox \mathcal{A}_i some assertions. One natural way to define an expansion modifier on MBox is to use the notion of a deductive closure of *DL-Lite* knowledge bases. There are different definitions of deductive closures in *DL-Lite* (e.g. [Lembo et al., 2010]). Here, we propose the one that is appropriate for the context of inconsistency handling.

Note that the use of deductive closure of an ABox fully makes sense in DL languages, while for instance in propositional logic the closure of an inconsistent knowledge base trivially leads to produce the whole language. We denote by $\circ_{cl}(\mathcal{M})$ the expansion modifier on MBox \mathcal{M} , defined as:

Let us denote by $\circ_{cl}(\mathcal{M})$ the deductive closure operator of an MBox \mathcal{M} , defined as follows:

Definition 5.2. Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$ be an MBox *DL-Lite* knowledge base. Let \mathcal{T}_p be the set of all positive inclusion axioms of \mathcal{T} ¹, D_I be the set of all individuals in all ABox's; D_C and D_R be the set of concepts respectively roles in $\mathcal{K}_{\mathcal{M}}$.

- Standard *DL-Lite*: the deductive closure of a given ABox \mathcal{A}_i with respect to \mathcal{T} is defined as:

$$\begin{aligned} Cl_{\mathcal{T}}(\mathcal{A}_i) &= \{B(a) : \langle \mathcal{T}_p, \mathcal{A}_i \rangle \models B(a), B \in D_C, a \in D_I\} \\ &\cup \\ &\{R(a, b) : \langle \mathcal{T}_p, \mathcal{A}_i \rangle \models R(a, b), R \in D_R, a \in D_I, b \in D_I\} \end{aligned}$$

- *DL-Lite* with an MBox: the deductive closure of \mathcal{M} is defined as:

$$\circ_{cl}(\mathcal{M}) = \{Cl(\mathcal{A}_i) : \mathcal{A}_i \in \mathcal{M}\}$$

The expansion modifier $\circ_{cl}(\mathcal{M})$ takes as input an MBox $\mathcal{M} = \{\mathcal{A}_1, \dots, \mathcal{A}_n\}$ and produces as output an MBox $\circ_{cl}(\mathcal{M})$ obtained by replacing each $\mathcal{A}_i \in \mathcal{M}$ by its deductive closure.

Example 5.2. Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$ be a *DL-Lite* knowledge base where $\mathcal{T} = \{\exists P \sqsubseteq B, A \sqsubseteq B, A \sqsubseteq \neg C\}$ and $\mathcal{M} = \{\{A(a), P(c, b)\}, \{C(d)\}\}$. Using Definition 5.2, the deductive closure of \mathcal{M} is : $\circ_{cl}(\mathcal{M}) = \{\{A(a), B(a), P(c, b), B(c)\}, \{C(d)\}\}$. ■

The deductive closure represents one natural way to define an expansion of an MBox. Another natural way to define an expansion is to add to each $\mathcal{A}_i \in \mathcal{M}$ the set of common assertions that can be derived from each $\langle \mathcal{T}, \mathcal{A}_i \rangle$.

$$\begin{aligned} \mathcal{A}_i &= \mathcal{A}_i \cup \\ &\{B(a) : \forall \mathcal{A}_j \in \mathcal{M}, \langle \mathcal{T}_p, \mathcal{A}_j \rangle \models B(a), a \in \mathbb{D}_I, B \in \mathbb{D}_C\} \cup \\ &\{R(a, b) : \forall \mathcal{A}_j \in \mathcal{M}, \langle \mathcal{T}_p, \mathcal{A}_j \rangle \models R(a, b), R \in \mathbb{D}_R, a \in \mathbb{D}_I, b \in \mathbb{D}_I\} \end{aligned}$$

Where \mathbb{D}_C (*resp.* $\mathbb{D}_R, \mathbb{D}_I$) is the set of all concepts (*resp.* roles, individuals) used in \mathcal{M} .

In the rest of the chapter, by an expansion modifier, we refer to the use of deductive closure modifier \circ_{cl} given in Definition 5.2.

Splitting modifiers

The second class of modifiers is called splitting modifiers. The idea is to replace some \mathcal{A}_i of an MBox by one or several of their subsets. This typically happens when some \mathcal{A}_i is inconsistent with respect to

¹Positive inclusion axioms are of the form $B_1 \sqsubseteq B_2$.

\mathcal{T} . Then it may be reasonable to replace each \mathcal{A}_i by some of its consistent subsets. Here, we choose as a splitting modifier the widely used inclusion-based maximally consistent subsets, defined by:

Definition 5.3. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a standard *DL-Lite* knowledge base. $\mathcal{R} \subseteq \mathcal{A}$ is an inclusion-based maximally consistent subset of \mathcal{A} with respect to \mathcal{T} if and only if:

1. $\langle \mathcal{T}, \mathcal{R} \rangle$ is consistent,
2. if $\mathcal{R} \neq \mathcal{A}$ then $\forall \mathcal{R}' : \mathcal{R} \subsetneq \mathcal{R}', \langle \mathcal{T}, \mathcal{R}' \rangle$ is inconsistent.

According to Definition 5.3, adding any fact f from $\mathcal{A} \setminus \mathcal{R}$ to \mathcal{R} entails the inconsistency of $\langle \mathcal{T}, \mathcal{R} \cup \{f\} \rangle$. Note that if \mathcal{K} is consistent, then \mathcal{K} admits a unique maximally consistent subset which is $\mathcal{R} = \mathcal{A}$. The splitting modifier on a single ABox \mathcal{A} , denoted indifferently by $\circ_{incl}(\mathcal{A})$ or $\circ_{incl}(\{\mathcal{A}\})$, is the set of all maximally inclusion-based consistent subsets of \mathcal{A} with respect to \mathcal{T} .

Definition 5.3 is extended to MBox as follows:

Definition 5.4. Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$ be an MBox *DL-Lite* knowledge base. An inclusion-based modifier on \mathcal{M} , denoted $\circ_{incl}(\mathcal{M})$, is defined by:

$$\circ_{incl}(\mathcal{M}) = \bigcup_{\mathcal{A}_i \in \mathcal{M}} \{X : X \in \circ_{incl}(\mathcal{A}_i)\}$$

Namely, $\circ_{incl}(\mathcal{M})$ consists in replacing each inconsistent \mathcal{A}_i of \mathcal{M} by its maximally consistent sub-bases.

Example 5.3. Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$ be a *DL-Lite* knowledge base where $\mathcal{T} = \{C \sqsubseteq \neg B\}$ and $\mathcal{M} = \{\{B(a), C(a), B(b)\}, \{C(e), B(e)\}\}$.

Using Definition 5.4, $\circ_{incl}(\mathcal{M}) = \{\{B(a), B(b)\}, \{C(a), B(b)\}, \{C(e)\}, \{B(e)\}\}$. ■

Selection-based modifiers

The last elementary modifiers considered in this chapter are selection-based modifiers which consist in considering only some subsets of \mathcal{M} to make inferences for instance. An example of a selection modifier simply consists in keeping only ABox's issued from the most reliable sources and getting rid those that are not enough reliable. Another natural way to define such a selection function is to only keep the largest ABox's. This selection function, adopted in this chapter, is called cardinality-based selection, denoted by $\circ_{card}(\mathcal{M})$ and is defined as follows:

Definition 5.5. Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$ be an MBox *DL-Lite* knowledge base. A cardinality-based selection on MBox \mathcal{M} is an MBox, denoted $\circ_{card}(\mathcal{M})$, defined by:

$$\circ_{card}(\mathcal{A}) = \{A_i : A_i \in \mathcal{M} \text{ such that } \nexists A_j \in \mathcal{M}, |A_j| > |A_i|\}.$$

Namely, $\circ_{card}(\mathcal{M})$ selects among the ABox's in \mathcal{M} the ones with maximal assertion number.

Example 5.4. Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$ be a *DL-Lite* knowledge base where $\mathcal{T} = \{\exists P \sqsubseteq B, B \sqsubseteq \neg C\}$ and $\mathcal{M} = \{\mathcal{A}_1 = \{P(c, b), B(a)\}, \mathcal{A}_2 = \{C(a), B(b)\}, \mathcal{A}_3 = \{B(c)\}\}$.

One can check that $\circ_{card}(\mathcal{M}) = \{\{P(c, b), B(a)\}, \{C(a), B(b)\}\}$. Indeed, $|\mathcal{A}_1| = |\mathcal{A}_2| = 2$, while $|\mathcal{A}_3| = 1$. Hence, only \mathcal{A}_1 and \mathcal{A}_2 are kept. ■

5.2.3 Composite modifiers on MBox

In the above section, we presented three main elementary modifiers² that operate on an MBox. These modifiers can be combined and composed to define composite modifiers.

The following lemma first shows that the elementary modifiers \circ_{cl} , \circ_{incl} and \circ_{card} are idempotent. Besides, it also shows that expansion and splitting modifiers only need to be applied once. These properties considerably reduce the number of combinations that can be done on elementary modifiers.

Lemma 5.1. *Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$ be an MBox DL-Lite knowledge base.*

1. *Let \circ_{cl} , \circ_{incl} , \circ_{card} be the three elementary modifiers on MBox presented in Section 5.2.2. Then:*

- (a) $\circ_{incl}(\circ_{incl}(\mathcal{M})) = \circ_{incl}(\mathcal{M})$
- (b) $\circ_{card}(\circ_{card}(\mathcal{M})) = \circ_{card}(\mathcal{M})$, and
- (c) $\circ_{cl}(\circ_{cl}(\mathcal{M})) = \circ_{cl}(\mathcal{M})$.

2. *Let \circ_d be a composite modifier (i.e. a combination of elementary modifiers). Then:*

- (a) $\circ_{cl}(\circ_d(\circ_{cl}(\mathcal{M}))) = \circ_d(\circ_{cl}(\mathcal{M}))$, and
- (b) $\circ_{incl}(\circ_d(\circ_{incl}(\mathcal{M}))) = \circ_d(\circ_{incl}(\mathcal{M}))$.

Proof of Lemma 5.1. The proof of the item (1.a) follows from the facts that:

- $\forall \mathcal{A}_i \in \circ_{incl}(\mathcal{M})$, $\langle \mathcal{T}, \mathcal{A}_i \rangle$ is consistent,
- if $\langle \mathcal{T}, \mathcal{A}_i \rangle$ is consistent, then $\circ_{incl}(\mathcal{A}_i) = \{\mathcal{A}_i\}$.

The proof of the item (1.b) follows from the facts that:

- $\forall \mathcal{A}_i \in \circ_{card}(\mathcal{M})$, $\forall \mathcal{A}_j \in \circ_{card}(\mathcal{M})$, we have $|\mathcal{A}_i| = |\mathcal{A}_j|$
- if $\forall \mathcal{A}_i \in \circ_{card}(\mathcal{M})$, $\forall \mathcal{A}_j \in \circ_{card}(\mathcal{M})$, $|\mathcal{A}_i| = |\mathcal{A}_j|$ then $\circ_{card}(\mathcal{M}) = \mathcal{M}$.

For item (1.c), it is enough to show that for a given $\mathcal{A} \in \mathcal{M}$, $\circ_{cl}(\circ_{cl}(\mathcal{A})) = \circ_{cl}(\mathcal{A})$. From the definition of \circ_{cl} , clearly we have $\circ_{cl}(\mathcal{A}) \subseteq \circ_{cl}(\circ_{cl}(\mathcal{A}))$. Now assume that $f \in \circ_{cl}(\circ_{cl}(\mathcal{A}))$ but $f \notin \circ_{cl}(\mathcal{A})$. Let $B_f \subseteq \circ_{cl}(\mathcal{A})$ be the subset that allows to derive f , namely $\langle \mathcal{T}_p, B_f \rangle \models f$. Now for each element x of B_f , we have $\langle \mathcal{T}_p, \mathcal{A} \rangle \models x$. Then clearly, $\langle \mathcal{T}_p, \mathcal{A} \rangle \models f$.

Regarding item (2.a), if \circ_d is an elementary modifier then it can be either \circ_{cl} , \circ_{card} , or \circ_{incl} . If $\circ_d = \circ_{cl}$ then the result holds thanks to item (1.c). If $\circ_d = \circ_{card}$ then the selected elements from $\circ_{card}(\circ_{cl}(\mathcal{M}))$ are closed set of assertions since \circ_{card} only discards some elements of $\circ_{cl}(\mathcal{M})$ but does not change the content of remaining elements. Lastly, let us consider the case where $\circ_d = \circ_{incl}$. Again $\forall \mathcal{A}' \in \circ_{incl}(\circ_{cl}(\mathcal{M}))$, $\mathcal{A}' = \circ_{cl}(\mathcal{A}')$. Let us recall that \mathcal{A}' is maximally consistent subset of $\mathcal{A} \in \circ_{cl}(\mathcal{M})$, with $\mathcal{A} = \circ_{cl}(\mathcal{A})$. If $\mathcal{A}' \neq \circ_{cl}(\mathcal{A})$ this means that $\exists f \in \circ_{cl}(\mathcal{A}')$ (hence $f \in \mathcal{A}$) such that $f \notin \mathcal{A}'$ despite the fact that $\langle \mathcal{T}, \mathcal{A}' \rangle \models f$. This is impossible since \mathcal{A}' should be a maximal consistent subbase of \mathcal{A} . Since each $\circ_d \in \{\circ_{cl}, \circ_{card}, \circ_{incl}\}$ applied on closed ABox preserves the closeness property, then clearly a composite modifier also preserves this closeness property.

The proof of item (2.b) follows immediately from the fact that

- $\forall \mathcal{A}_i \in \circ_{incl}(\mathcal{M})$, $\langle \mathcal{T}, \mathcal{A}_i \rangle$ is consistent,

²One can add other elementary modifiers as a concatenate modifier or a merging modifier (using different strategies). This will lead to obtaining other composite modifiers which are not considered in this chapter and are left to future works

- if \mathcal{M} is consistent, then $\forall \circ_d \in \{\circ_{cl}, \circ_{card}, \circ_{incl}\}$ gives a consistent subbase, and
- $\circ_{incl}(\mathcal{M}) = \mathcal{M}$ if \mathcal{M} is consistent.

□

Figure 5.1 presents the set of all possible composite modifiers that can be applied on MBox. The starting point is an MBox DL-Lite knowledge base $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$.

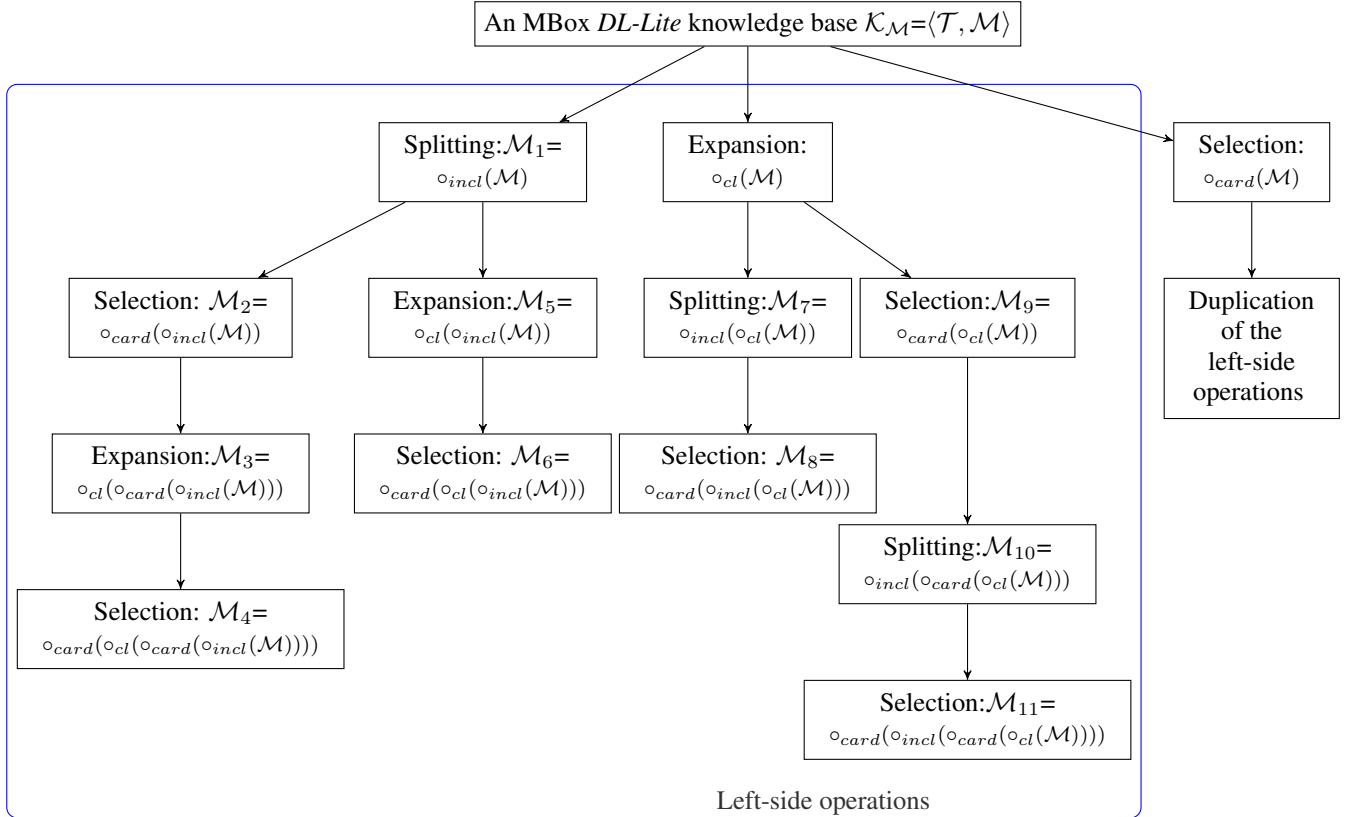


Figure 5.1: Composite modifiers on MBox

At the beginning, one can either use expansion modifier ($\circ_{cl}(\mathcal{M})$), selection modifier ($\circ_{card}(\mathcal{M})$) or a splitting modifier ($\circ_{incl}(\mathcal{M})$). Expansion can only be followed either by a splitting modifier ($\circ_{incl}(\circ_{cl}(\mathcal{M}))$) or by a selection modifier ($\circ_{card}(\circ_{cl}(\mathcal{M}))$). From the splitting operation one can only make selection ($\circ_{card}(\circ_{incl}(\circ_{cl}(\mathcal{M})))$) thanks to Lemma 5.1. From the selection operation one can only make splitting operation ($\circ_{incl}(\circ_{card}(\circ_{cl}(\mathcal{M})))$) thanks to Lemma 5.1. This splitting operation allows again the reuse of a selection operation ($\circ_{card}(\circ_{incl}(\circ_{card}(\circ_{cl}(\mathcal{M}))))$).

From the MBox $\circ_{incl}(\circ_{cl}(\mathcal{M}))$ only a selection can be applied, thanks to Lemma 5.1, where $\circ_{incl}(\mathcal{M})$ and $\circ_{cl}(\mathcal{M})$ only needs to be applied once. Similarly, if one starts with a splitting operation followed by a selection operation, then only an expansion can be applied (thanks to Lemma 5.1 where $\circ_{incl}(\mathcal{M})$ needs only to be applied). From $\circ_{cl}(\circ_{card}(\circ_{incl}(\mathcal{M})))$ only a selection can be applied, since again from Lemma 5.1, $\circ_{incl}(\mathcal{M})$ and $\circ_{cl}(\mathcal{M})$ needs only to be applied once. If one starts by a splitting modifier followed by an expansion, then only a selection needs to be applied. If one starts with a selection operation, then one can either apply an expansion or a splitting operation, and thus, we duplicate the same operations presented in the left-side box of the figure.

Note that if one starts with an MBox *DL-Lite* knowledge base that only contains one ABox, then there is no need to add a selection child to the root. Similarly for such singleton MBox, applying an expansion modifier followed by a selection modifier leads to same results as if one start with an expansion modifier followed by a splitting modifier ($\mathcal{M}_9 = \circ_{cl}(\mathcal{M})$). Hence we have $\mathcal{M}_{10} = \mathcal{M}_7$ and $\mathcal{M}_{11} = \mathcal{M}_8$. The case where the starting point is a singleton MBox will be discussed in Section 5.4.

5.3 Inference-based strategies from MBox

This section addresses the issue of query answering from MBox *DL-Lite* knowledge bases. It presents several inference-based strategies that can be applied on an MBox *DL-Lite* knowledge base. An inference-based strategy takes as input an MBox \mathcal{M} , a TBox \mathcal{T} (*i.e.* an ontology) and a query and aims to find if there exists an answer for such a query over the set of ABox's of the MBox *DL-Lite* knowledge base $\mathcal{K} = \langle \mathcal{T}, \mathcal{M} \rangle$.

Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$ be a *DL-Lite* knowledge base with an MBox \mathcal{M} . The following subsections provide main inference-based strategies that can be applied on $\mathcal{K}_{\mathcal{M}}$.

5.3.1 Universal inference

The universal inference-based strategy states that a conclusion is valid if and only if it can be obtained (in a standard way) from every ABox \mathcal{A}_i of a given MBox \mathcal{M} . More precisely,

Definition 5.6. Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$ be a *DL-Lite* knowledge base with an MBox $\mathcal{M} = \{\mathcal{A}_1, \dots, \mathcal{A}_n\}$. A query q is said to be a universal consequence of $\mathcal{K}_{\mathcal{M}}$, denoted by $\mathcal{K}_{\mathcal{M}} \models_{\forall} q$, if and only if $\forall \mathcal{A}_i \in \mathcal{M}, \langle \mathcal{T}, \mathcal{A}_i \rangle \models q$ ³.

Example 5.5. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{M} \rangle$ be a *DL-Lite* knowledge base where $\mathcal{T} = \{D \sqsubseteq B, C \sqsubseteq B, D \sqsubseteq \neg C\}$ and $\mathcal{M} = \{\{D(a), C(b)\}, \{C(a)\}\}$. One can check that $\mathcal{K}_{\mathcal{M}} \models_{\forall} B(a)$, since

- $\langle \mathcal{T}, \{D(a), C(b)\} \rangle \models B(a)$, and
- $\langle \mathcal{T}, \{C(a)\} \rangle \models B(a)$.

■

The universal inference is a standard way to derive conclusions from different sources. It is also known as a skeptical inference, used for instance in default reasoning [Reiter, 1987], where one only accepts conclusions derived from each extension of a default theory.

In Definition 5.6, q in general represents a first order formula. Now, when we deal with *DL-Lite* framework, q may (and often) represents a conjunctive query.

When q is a boolean query, then q holds universally from an MBox *DL-Lite* knowledge base $\mathcal{K}_{\mathcal{M}}$ if and only if q holds in each standard *DL-Lite* knowledge base $\mathcal{K}_i = \langle \mathcal{T}, \mathcal{A}_i \rangle$, with $\mathcal{A}_i \in \mathcal{M}$.

If q is a general conjunctive query of the form

$$q(x_1, \dots, x_n) \leftarrow (x_1, \dots, x_n) : \exists y_1, \dots, y_l. B_1 \wedge \dots \wedge B_m$$

then $q(a_1, \dots, a_n)$ universally follows from an MBox *DL-Lite* knowledge base $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$ if $q(a_1, \dots, a_n)$ follows from each standard *DL-Lite* knowledge base $\mathcal{K}_i = \langle \mathcal{T}, \mathcal{A}_i \rangle$ with $\mathcal{A}_i \in \mathcal{M}$, where (a_1, \dots, a_n) is an instance of the distinguished variables (x_1, \dots, x_n) .

³ \models is the inference relation from a standard *DL-Lite* knowledge base presented in Section 1.4

Said differently, using the universal consequence relation, answers to a query q from an MBox *DL-Lite* knowledge base $\mathcal{K}_{\mathcal{M}}$ is defined as the intersection of answers to q obtained from each standard *DL-Lite* \mathcal{K}_i of $\mathcal{K}_{\mathcal{M}}$.

Example 5.6. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{M} \rangle$ be a *DL-Lite* knowledge base where $\mathcal{T} = \{\exists S \sqsubseteq B, \exists S^- \sqsubseteq C, S \sqsubseteq \neg H, \exists H \sqsubseteq B, \exists H^- \sqsubseteq D\}$, and $\mathcal{M} = \{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3\}$ where $\mathcal{A}_1 = \{S(a, c), H(c, b), S(b, f), H(f, k)\}$, $\mathcal{A}_2 = \{S(a, d), H(d, b), S(b, t)\}$ and $\mathcal{A}_3 = \{S(a, e), H(g, b), S(b, y)\}$

Let us illustrate universal inferences over the different forms of queries: instance query (or instance checking), atomic ground queries, boolean queries, general conjunctive queries and first order formulas.

1. Let $q \leftarrow B(a)$ be an instance query. One can check that $\mathcal{K}_{\mathcal{M}} \models_{\forall} B(a)$, since $B(a)$ holds respectively from $\mathcal{A}_1, \mathcal{A}_2$, and \mathcal{A}_3 using \mathcal{T} .
2. Let $q \leftarrow B(b) \wedge D(b)$ be an atomic ground query. One can check that $\mathcal{K}_{\mathcal{M}} \models_{\forall} q$.
3. Let $q \leftarrow \exists z. D(b) \wedge S(b, z)$ be a boolean query with existential variables. One can verify that $\mathcal{K}_{\mathcal{M}} \models_{\forall} q$.
4. Let $q(x_1, x_2) \leftarrow (x_1, x_2) \exists y. S(x_1, y) \wedge H(y, x_2)$ be a conjunctive query. The answer sets that can be computed from each ABox in \mathcal{M} are :
 Using \mathcal{A}_1 : $\{\langle a, b \rangle, \langle b, k \rangle\}$
 Using \mathcal{A}_2 : $\{\langle a, b \rangle\}$
 Using \mathcal{A}_3 : $\{\langle a, b \rangle\}$
 The certain answers to q that universally hold from $\mathcal{K}_{\mathcal{M}}$ are: $\{\langle a, b \rangle\}$.
5. Let $q \leftarrow C(c) \vee C(d) \vee C(e)$ be a first order query. One can verify that $\mathcal{K}_{\mathcal{M}} \models_{\forall} q$.

■

5.3.2 Existential inference

The existential inference-based strategy is an inference strategy that only checks if a conclusion holds from at least one ABox of a given MBox. More formally,

Definition 5.7. Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$ be a *DL-Lite* knowledge base with an MBox $\mathcal{M} = \{\mathcal{A}_1, \dots, \mathcal{A}_n\}$. A query q is said to be an existential consequence of $\mathcal{K}_{\mathcal{M}}$, denoted by $\mathcal{K}_{\mathcal{M}} \models_{\exists} q$, if and only if $\exists \mathcal{A}_i \in \mathcal{M}, \langle \mathcal{T}, \mathcal{A}_i \rangle \models q$.

The existential inference, called also credulous inference, is a very adventurous inference relation. It only makes sense in some decision problems when one is only looking for a possible solution of a set of constraints or preferences. It is often considered as undesirable when $\mathcal{K}_{\mathcal{M}}$ represents available knowledge base on some problem. The existential consequence relation is so adventurous that it may lead to an inconsistent set of conclusions (with respect to \mathcal{T}).

Example 5.7. Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$ be a *DL-Lite* where $\mathcal{T} = \{C \sqsubseteq B\}$ and $\mathcal{M} = \{\{C(a), B(a)\}, \{C(b), B(d)\}, \{B(c)\}\}$. One can check that $\mathcal{K}_{\mathcal{M}} \models_{\exists} B(b)$, since $\langle \mathcal{T}, \{C(b), B(d)\} \rangle \models B(b)$.

■

5.3.3 Safe inference

The safe inference-based strategy considers as valid conclusions those that are only derived from facts belonging to the intersection of all ABox's. More formally,

Definition 5.8. Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$ be a *DL-Lite* knowledge base with an MBox $\mathcal{M} = \{\mathcal{A}_1, \dots, \mathcal{A}_n\}$. A query q is said to be a safe consequence of $\mathcal{K}_{\mathcal{M}}$, denoted by $\mathcal{K}_{\mathcal{M}} \models_{\cap} q$, if and only if $\langle \mathcal{T}, \bigcap_{\mathcal{A}_i \in \mathcal{M}} \mathcal{A}_i \rangle \models q$.

Obviously, the safe inference is a very sound and conservative inference relation since it only considers common assertions between the different ABox, to perform inferences.

Example 5.8. Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$ be a *DL-Lite* knowledge base where $\mathcal{T} = \{C \sqsubseteq B\}$ and $\mathcal{M} = \{\{C(a)\}, \{C(a), B(b)\}, \{C(a), B(c)\}\}$. We have $\bigcap_{\mathcal{A}_i \in \mathcal{M}} \mathcal{A}_i = \{C(a)\}$, and thus, $\mathcal{K}_{\mathcal{M}} \models_{\cap} B(a)$, since $\langle \mathcal{T}, \{C(a)\} \rangle \models B(a)$. ■

5.3.4 Other inferences

We now provide additional inference-based strategies called: Majority-based inference, Proportional-based inference and Non-objection inference. These inference relations offer a good compromise between universal or safe inference relations and existential inference relations.

Majority-based inference

The majority-based inference relation (*maj* for short) considers a conclusion as valid if it is confirmed by the majority of ABox's. More formally:

Definition 5.9. Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$ be a *DL-Lite* knowledge base with an MBox $\mathcal{M} = \{\mathcal{A}_1, \dots, \mathcal{A}_n\}$. A query q is said to be a majority-based consequence of $\mathcal{K}_{\mathcal{M}}$, denoted $\mathcal{K}_{\mathcal{M}} \models_{maj} q$, if and only if:

$$\frac{|\mathcal{A}_i : \mathcal{A}_i \in \mathcal{M}, \langle \mathcal{T}, \mathcal{A}_i \rangle \models q|}{|\mathcal{M}|} > 1/2.$$

Definition 5.9 simply states that a query q is a majority-based consequence of $\mathcal{K}_{\mathcal{M}}$ if and only if it can be deduced from more than the half of ABox's in \mathcal{M} .

Example 5.9. Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$ be a *DL-Lite* knowledge base where $\mathcal{T} = \{C \sqsubseteq B\}$ and $\mathcal{M} = \{\{C(a)\}, \{C(a), B(b)\}, \{C(c), B(c)\}\}$.

We have $\langle \mathcal{T}, \{C(a)\} \rangle \models B(a)$ and $\langle \mathcal{T}, \{C(a), B(b)\} \rangle \models B(a)$ and $|\mathcal{M}| = 3$. Hence $\mathcal{K}_{\mathcal{M}} \models_{maj} B(a)$. ■

Proportional- α -based inference

The Proportional- α -based inference (*prop* for short) requires that a conclusion is valid if it can be derived from a proportion α of ABox's of an MBox. More formally:

Definition 5.10. Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} \rangle$ be a *DL-Lite* knowledge base with an MBox $\mathcal{M} = \{\mathcal{A}_1, \dots, \mathcal{A}_n\}$ and $\alpha \geq 0$. A query q is said to be a proportional- α -based consequence of $\mathcal{K}_{\mathcal{M}}$, denoted $\mathcal{K}_{\mathcal{M}} \models_{\alpha} q$, if and only if

$$\frac{|\mathcal{A}_i : \mathcal{A}_i \in \mathcal{M}, \langle \mathcal{T}, \mathcal{A}_i \rangle \models q|}{|\mathcal{M}|} \geq \alpha.$$

Following Definition 5.10, the proportional- α -based inference generalizes the above inference-based strategies as follows:

- if $\alpha = 1$, then we recover the universal inference.
- if $\alpha > 1/2$, then we recover the majority-based inference.
- if $\alpha = \epsilon$, then we recover the existential inference.

Note that, the major difference of Proportional- α -based inference relation is how to fit the parameter α .

Non-objection inference

The non-objection inference or argued inference (*obj* for short) is an inference strategy where a conclusion q is accepted if it can be obtained using at least one ABox while there is no ABox that implies $\neg q$. More formally:

Definition 5.11. Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a DL-Lite knowledge base with an MBox $\mathcal{M} = \{\mathcal{A}_1, \dots, \mathcal{A}_n\}$. A query q is said to be a non-objection consequence relation of $\mathcal{K}_{\mathcal{M}}$, denoted by $\mathcal{K}_{\mathcal{M}} \models_{obj} q$, if and only if

1. $\exists \mathcal{A}_i \in \mathcal{M} : \langle \mathcal{T}, \mathcal{A}_i \rangle \models q$, and
2. there exists no $\mathcal{A}_j \in \mathcal{M}$ such that $\langle \mathcal{T}, \mathcal{A}_j \rangle \models \neg q$.

Example 5.10. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{M} \rangle$ be a DL-Lite where $\mathcal{T} = \{C \sqsubseteq \neg B, B \sqsubseteq D\}$ and $\mathcal{M} = \{\{C(a)\}, \{B(a)\}\}$. One can check that $\mathcal{K} \models_{obj} D(a)$. ■

One can easily check that, if q is a first order formula then the non-objection inference is more cautious than the existential inference relation but it is more productive than the universal inference relation.

As it is said before, within DL-Lite framework, q is often restricted to conjunctive queries. In this case, the second item of Definition 5.11 does not really make sense, since negation is not allowed in the definition of a conjunctive query. Besides, if one restricts inferences to boolean queries, and interprets $\langle \mathcal{T}, \mathcal{A}_i \rangle \models \neg q$ in definition as $\langle \mathcal{T}, \mathcal{A}_i \rangle \models q$ does not hold, then the non-objection is simply equivalent to existential inference.

In the rest of the chapter, we will not make reference to non-objection inference, since it comes down to existential inference for boolean query. Of course, one may weaken item 2 of Definition 5.11 for atomic grounded queries of the form $(\bigwedge_{i=1}^n A_i(a)) \wedge (\bigwedge_{j=1}^m P_j(a, b))$ where A_i and P_j are concepts and roles respectively. In item 2, one may replace $\langle \mathcal{T}, \mathcal{A}_i \rangle \models \neg q$ by $\langle \mathcal{T}, \mathcal{A}_i \cup q \rangle$ is inconsistent. This weakened form of Definition 5.11 is not considered in the chapter and is left for further research.

Given the limitations of existential, non-objection and proportional-based inferences, in the rest of the chapter, we only focus on universal, safe and majority-based inference.

5.3.5 Comparing inference-based strategies from a fixed MBox

Given a fixed MBox \mathcal{M} , the following figure summarizes the cautiousness relationships between different inference-based strategies, defined in the above subsections.

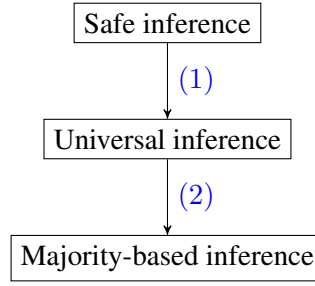


Figure 5.2: Comparison between inference-based strategies

The top of Figure 5.2 corresponds to the most conservative inference relation which is the safe inference relation. The existential inference is the most productive inference. In Figure 5.2, $X \longrightarrow Y$ means that each conclusion of X is also a conclusion of Y .

Proposition 5.1 (Figure 5.2). *Let \mathcal{M} be a consistent MBox with respect to a TBox \mathcal{T} . Let q be a query. Then:*

1. *if $\langle \mathcal{T}, \mathcal{M} \rangle \models_{\cap} q$ then $\langle \mathcal{T}, \mathcal{M} \rangle \models_{\vee} q$.*
2. *if $\langle \mathcal{T}, \mathcal{M} \rangle \models_{\vee} q$ then $\langle \mathcal{T}, \mathcal{M} \rangle \models_{maj} q$.*

Proof of Proposition 5.1. Item 1 holds from the fact that $\forall \mathcal{A}_i \in \mathcal{M}$, we have $(\bigcap_{\mathcal{A}_i \in \mathcal{M}} \mathcal{A}_i) \subseteq \mathcal{A}_i$. Item 2 holds due to the fact that universal consequence requires that q follows from all ABox in \mathcal{M} . Hence, q holds in at least the half of \mathcal{A}_i 's in \mathcal{M} . \square

Example 5.11 (Counter-examples of Proposition 5.1). The following gives counter-examples for items 1-3 of Proposition 5.1.

1. Let $\mathcal{T} = \{D \sqsubseteq B, C \sqsubseteq B, D \sqsubseteq \neg C\}$ and $\mathcal{M} = \{\{D(a), C(b)\}, \{C(a)\}\}$. Let $q \leftarrow B(a)$. We have $\langle \mathcal{T}, \mathcal{M} \rangle \models_{\vee} B(a)$ but $\langle \mathcal{T}, \mathcal{M} \rangle \not\models_{\cap} B(a)$.
2. Let $\mathcal{T} = \{D \sqsubseteq B\}$ and $\mathcal{M} = \{\{D(a)\}, \{D(a), B(b)\}, \{D(c), B(c)\}\}$. Let $q \leftarrow B(a)$. We have $\langle \mathcal{T}, \mathcal{M} \rangle \models_{maj} B(a)$ but $\langle \mathcal{T}, \mathcal{M} \rangle \not\models_{\vee} B(a)$, since $\langle \mathcal{T}, \{D(c), B(c)\} \rangle$ does not allow to entail $B(a)$.

The following lemma gives cautiousness relations between different inference strategies from two MBox's where one MBox is included (or equal) in the other. This lemma and the next one will be helpful for next sections.

Lemma 5.2. *Let \mathcal{M}_1 and \mathcal{M}_2 be two consistent MBox with respect to a TBox \mathcal{T} s.t $\mathcal{M}_1 \subseteq \mathcal{M}_2$ (namely, $\forall \mathcal{A}_i \in \mathcal{M}_1$, we have $\mathcal{A}_i \in \mathcal{M}_2$). Let q be a query. Then:*

1. *If $\langle \mathcal{T}, \mathcal{M}_2 \rangle \models_{\vee} q$ then $\langle \mathcal{T}, \mathcal{M}_1 \rangle \models_{\vee} q$.*
2. *If $\langle \mathcal{T}, \mathcal{M}_2 \rangle \models_{\cap} q$ then $\langle \mathcal{T}, \mathcal{M}_1 \rangle \models_{\cap} q$.*
3. *The majority-based inference from \mathcal{M}_1 is incomparable with the one obtained from \mathcal{M}_2 .*

Proof of Lemma 5.2. The proof is immediate. For item 1, if q holds in all \mathcal{A}_i of \mathcal{M}_2 then trivially it holds in all \mathcal{A}_j of \mathcal{M}_1 (since $\mathcal{M}_1 \subseteq \mathcal{M}_2$). Item 2 holds due to the fact that $\mathcal{M}_1 \subseteq \mathcal{M}_2$ implies that $\bigcap_{\mathcal{A}_i \in \mathcal{M}_2} \mathcal{A}_i \subseteq \bigcap_{\mathcal{A}_j \in \mathcal{M}_1} \mathcal{A}_j$. \square

Example 5.12 (Counter-examples of Lemma 5.2). The converse of Items 1 and 2 does not hold. As it is shown by the following counter-example (we also provides the counter-example of item 3). Let $\mathcal{T} = \emptyset$, $\mathcal{M}_1 = \{B(a)\}$ and $\mathcal{M}_2 = \{\{B(a)\}, \{B(c)\}, \{B(c)\}\}$. First, note that $\mathcal{M}_1 \subseteq \mathcal{M}_2$. Clearly $\langle \mathcal{T}, \mathcal{M}_1 \rangle \models_{\forall} B(a)$ (resp. $\langle \mathcal{T}, \mathcal{M}_1 \rangle \models_{\cap} B(a)$) holds, while $\langle \mathcal{T}, \mathcal{M}_2 \rangle \models_{\forall} B(a)$ (resp. $\langle \mathcal{T}, \mathcal{M}_2 \rangle \models_{\cap} B(a)$) does not hold.

Regarding majority-based inference, one can check that $\langle \mathcal{T}, \mathcal{M}_1 \rangle \models_{maj} B(a)$ holds while $\langle \mathcal{T}, \mathcal{M}_2 \rangle \models_{maj} B(a)$ does not hold. And $\langle \mathcal{T}, \mathcal{M}_2 \rangle \models_{maj} B(c)$ holds while $\langle \mathcal{T}, \mathcal{M}_1 \rangle \models_{maj} B(c)$ does not hold.

The following inference relations hold between an MBox \mathcal{M} and its expansion $\circ_{cl}(\mathcal{M})$.

Lemma 5.3. *Let \mathcal{M}_1 and \mathcal{M}_2 be two consistent MBox with respect to \mathcal{T} . Let \mathcal{M}_2 be the closure of \mathcal{M}_1 , namely $\mathcal{M}_2 = \{Cl(\mathcal{A}_i) : \mathcal{A}_i \in \mathcal{M}_1\}$. Let q be a query. Then:*

1. $\langle \mathcal{T}, \mathcal{M}_1 \rangle \models_{\forall} q$ if and only if $\langle \mathcal{T}, \mathcal{M}_2 \rangle \models_{\forall} q$.
2. $\langle \mathcal{T}, \mathcal{M}_1 \rangle \models_{maj} q$ if and only if $\langle \mathcal{T}, \mathcal{M}_2 \rangle \models_{maj} q$.
3. if $\langle \mathcal{T}, \mathcal{M}_1 \rangle \models_{\cap} q$ then $\langle \mathcal{T}, \mathcal{M}_2 \rangle \models_{\cap} q$.

Proof of Lemma 5.3. The proof is again immediate. Item 1 and 2 follow from the fact that in standard *DL-Lite*, if \mathcal{A} is a consistent ABox with \mathcal{T} , then $\langle \mathcal{T}, \mathcal{A} \rangle \models q$ iff $\langle \mathcal{T}, Cl_{\mathcal{T}}(\mathcal{A}) \rangle \models q$. Item 3 follows from the fact that $\mathcal{A}_i \subseteq Cl(\mathcal{A}_i)$ for each $\mathcal{A}_i \in \mathcal{M}_1$. Hence $\bigcap_{\mathcal{A}_i \in \mathcal{M}_1} \mathcal{A}_i \subseteq \bigcap_{\mathcal{A}_i \in \mathcal{M}_1} Cl(\mathcal{A}_i) = \bigcap_{\mathcal{A}_j \in \mathcal{M}_2} \mathcal{A}_j$. \square

5.4 Handling inconsistency = Composite modifiers + inference strategies

As it is said in the introduction, an MBox may be issued from multiple-sources of information or may be resulted from handling inconsistent standard *DL-Lite* knowledge base \mathcal{K} , where each element of the MBox represents a possible repair of \mathcal{K} . In the context of multiple-sources of information, it may make sense to merge the ABox's of an MBox, in order to get a single ABox. As it is suggested in the title of this chapter, this work is not oriented towards merging ABox's but rather on the use of MBox as a way to represent and reason about inconsistent *DL-Lite* knowledge bases. We view the problem of repairing an inconsistent *DL-Lite* knowledge base as composed of a composite modifier on a given MBox followed by an inference-based strategy.

As it was said before, from Figure 5.1, if one starts with a standard *DL-Lite* knowledge base (i.e. a single MBox), there are only eight main composite modifiers useful for handling inconsistency (Lemma 5.4). These composite modifiers are given in Figure 5.1, and summarized in Table 5.1.

MBox	Combination
\mathcal{M}_1	$\circ_{\mathcal{M}_1} = \circ_{incl}(\mathcal{M})$
\mathcal{M}_2	$\circ_{\mathcal{M}_2} = \circ_{card}(\circ_{incl}(\mathcal{M}))$
\mathcal{M}_3	$\circ_{\mathcal{M}_3} = \circ_{cl}(\circ_{card}(\circ_{incl}(\mathcal{M})))$
\mathcal{M}_4	$\circ_{\mathcal{M}_4} = \circ_{card}(\circ_{cl}(\circ_{card}(\circ_{incl}(\mathcal{M}))))$
\mathcal{M}_5	$\circ_{\mathcal{M}_5} = \circ_{cl}(\circ_{incl}(\mathcal{M}))$
\mathcal{M}_6	$\circ_{\mathcal{M}_6} = \circ_{card}(\circ_{cl}(\circ_{incl}(\mathcal{M})))$
\mathcal{M}_7	$\circ_{\mathcal{M}_7} = \circ_{incl}(\circ_{cl}(\mathcal{M}))$
\mathcal{M}_8	$\circ_{\mathcal{M}_8} = \circ_{card}(\circ_{incl}(\circ_{cl}(\mathcal{M})))$

 Table 5.1: Composite modifiers on MBox with $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} = \{\mathcal{A}\} \rangle$

Lemma 5.4. Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} = \{\mathcal{A}\} \rangle$ be an inconsistent DL-Lite knowledge base. Let $\{\circ_{\mathcal{M}_1}, \dots, \circ_{\mathcal{M}_8}\}$ be the eight composite modifiers summarized in Table 5.1. Then for each composite modifier \circ_d (obtained by a finite combination of elementary modifiers $\circ_{incl}, \circ_{card}, \circ_{cl}$), there exists $\circ_c \in \{\circ_{\mathcal{M}_1}, \dots, \circ_{\mathcal{M}_8}\}$ such that $\circ_d(\mathcal{M}) = \circ_c(\mathcal{M})$.

Let us now provide the set inclusion relations between the different MBox's resulting from applying composite modifiers $\circ_{\mathcal{M}_1}, \dots, \circ_{\mathcal{M}_8}$ on an initial inconsistent DL-Lite knowledge base $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} = \{\mathcal{A}\} \rangle$. Figure 5.3 gives different relations between MBox's issued from applying the main composite modifiers:

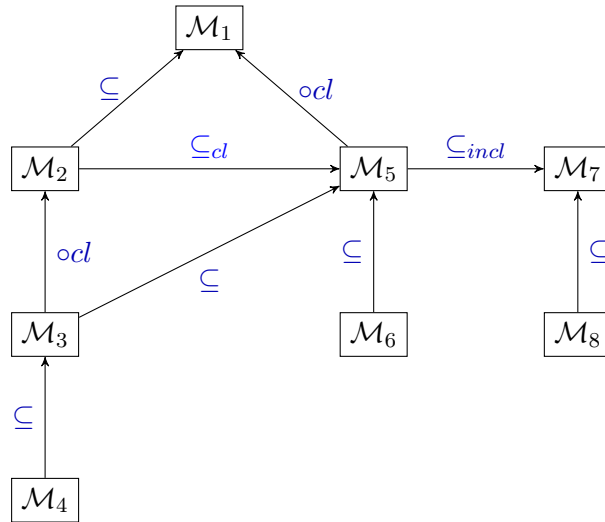


Figure 5.3: Inclusion relations between MBox's where " $X \xrightarrow{\subseteq} Y$ " means that the MBox X is included (set inclusion) in Y , " $X \xrightarrow{\subseteq_{incl}} Y$ " means that $\forall A \in X, \exists B \in Y$ s.t. $A \subseteq B$. " $X \xrightarrow{\subseteq_{cl}} Y$ " means that for each $A \in X, \exists B \in Y$ s.t. $B = \circ_{cl}(A)$ and " $X \xrightarrow{\subseteq_{incl}} Y$ " means that $X = \circ_{incl}(Y)$.

Let us briefly explain why the relations given in Figure 5.3 hold. First, note that by definition of the elementary modifier \circ_{card} , $\circ_{card}(\mathcal{M}) \subseteq \mathcal{M}$ (since \circ_{card} is a selection operation), hence the relations:

$$\mathcal{M}_4 \subseteq \mathcal{M}_3, \mathcal{M}_2 \subseteq \mathcal{M}_1, \mathcal{M}_6 \subseteq \mathcal{M}_5, \text{ and } \mathcal{M}_8 \subseteq \mathcal{M}_7$$

trivially hold.

Similarly, using the definition of the elementary modifier $\circ_{cl}()$, we also have:

$$\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1) \text{ and } \mathcal{M}_3 = \circ_{cl}(\mathcal{M}_2).$$

As consequence of the above relations, we have:

$$\mathcal{M}_3 \subseteq \mathcal{M}_5.$$

Regarding $\mathcal{M}_2 \circ_{cl} \mathcal{M}_5$, we have $\mathcal{M}_2 \subseteq \mathcal{M}_1$, hence $\forall A \in \mathcal{M}_2$, we also have $A \in \mathcal{M}_1$. Recall that $\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1)$. This means that $\forall A \in \mathcal{M}_2, \exists B \in \mathcal{M}_5$ such that $B = Cl(A)$.

It remains now to show that $\mathcal{M}_5 \circ_{cl} \mathcal{M}_7$. Let $B \in \circ_{incl}(\{\mathcal{A}\})$ and let us show that there exists a set of assertions X such that $\circ_{cl}(\{B\}) \subseteq X$ and $X \in \mathcal{M}_7$. Since $B \in \circ_{incl}(\{\mathcal{A}\})$, this means by definition that $B \subseteq A$ and hence $B \subseteq \circ_{cl}(A)$. Now, B is consistent, this means that there exists $R \in \circ_{incl}(\circ_{cl}(A)) = \mathcal{M}_7$ such that $B \in R$. From Lemma 5.1, R is a closed set of assertions, then this means that $Cl(B) \subseteq R$.

5.5 Comparative studies

We now compare main inference-based strategies (Section 5.3) applied on the eight identified composite modifiers for handling inconsistent standard DL-Lite knowledge base. The studies concern productivity relations and computational complexity.

5.5.1 Productivity

This section provides an exhaustive study of productivity results for modifier plus inference strategies combinations.

Composite modifiers + universal inference

The following figure summarizes existing cautiousness relations between universal consequence relations (\forall -entailment) applied on $\{\mathcal{M}_1, \dots, \mathcal{M}_8\}$ identified in Section 5.2.3.

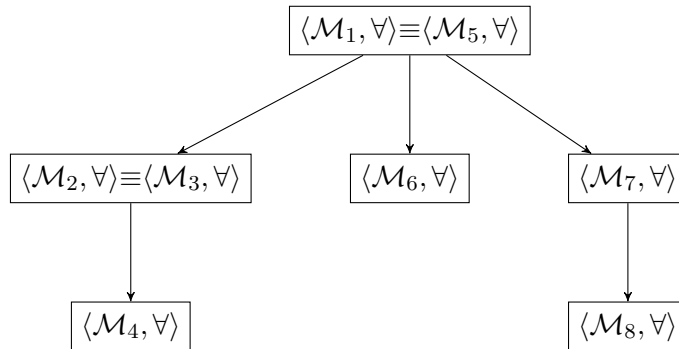


Figure 5.4: Relationships between \forall -entailment from \mathcal{M}_1 - \mathcal{M}_8

In Figure 5.4, the arrow $n1 \rightarrow n2$ means that each conclusion that can be derived using $n1$ is also a conclusion using $n2$. Proofs of different links and counter-examples of the converse relations are given in the Appendix.

Composite modifiers + safe inference

Figure 5.5 gives existing cautiousness relations between safe inferences (\cap -entailment) applied on $\{\mathcal{M}_1, \dots, \mathcal{M}_8\}$.

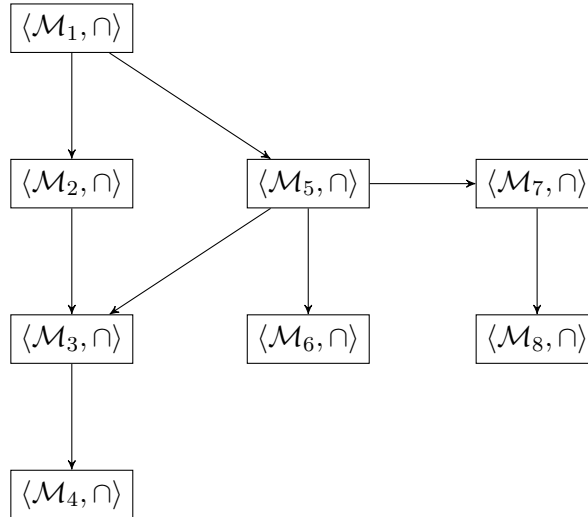


Figure 5.5: Relationships using safe entailment from \mathcal{M}_1 - \mathcal{M}_8

In Figure 5.5: the bold arrow $n1 \rightarrow n2$ means that each conclusion that can be derived using $n1$ is also a conclusion using $n2$.

Composite modifiers + majority-based inference

Figure 5.6 considers the case of majority-based consequence relations.

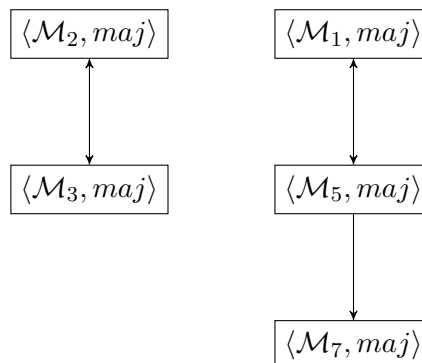


Figure 5.6: Relationships using majority-based inference from \mathcal{M}_1 - \mathcal{M}_8

Global schema

Figure 5.7 pairwise compares inference-based strategies given in Figure 5.2. Dashed arrow only holds for instance checking.

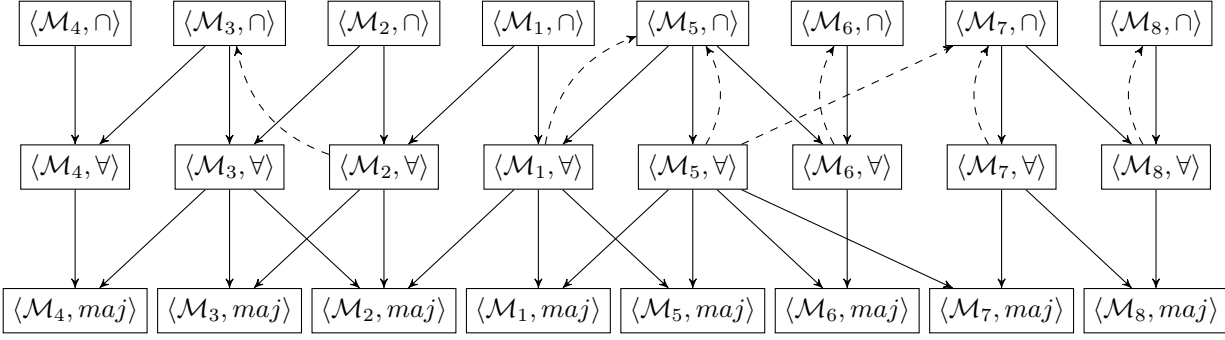


Figure 5.7: Relationships between different inference relations

Lastly, when the initial knowledge base is consistent, then all inference relations collapse with standard inferences, namely:

Proposition 5.2. *Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{M} \rangle$ be a consistent standard DL-Lite knowledge base. Then: $\forall s \in \{\forall, maj, \cap\}, \forall \mathcal{M}_i \in \{\mathcal{M}_1, \dots, \mathcal{M}_8\}, \langle \mathcal{T}, \mathcal{M}_i \rangle \models_s q$ if and only if $\langle \mathcal{T}, \mathcal{M} \rangle \models q$.*

Proof of Proposition 5.2. The proof basically follows from three facts:

1. when $\langle \mathcal{T}, \mathcal{M} \rangle$ is consistent then $\circ_{incl}(\mathcal{M}) = \mathcal{M}$ and $\circ_{card}(\mathcal{M}) = \mathcal{M}$,
2. $\mathcal{M}_1 = \mathcal{M}_2 = \mathcal{M}$ and for $i = 3, \dots, 8$: $\mathcal{M}_i = \circ_{cl}(\mathcal{M})$, and
3. $\forall q, \langle \mathcal{T}, \mathcal{M} \rangle \models q$ if and only if $\langle \mathcal{T}, \circ_{cl}(\mathcal{M}) \rangle \models q$.

□

5.5.2 Complexity analysis

This section discusses computational complexity of inference relations presented in the chapter. We first give the main ideas behind the complexity of inference relation $\langle \mathcal{M}_2 = \mathcal{L}^{\circ_{incl}}(\mathcal{M}), \forall \rangle$, and then provide different tables summarizing complexity results of inference relations studied in this chapter.

Let \mathcal{K} be an inconsistent knowledge base, we first define the notion of conflict which is a minimal inconsistent subset of \mathcal{A} , more formally:

Definition 5.12. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an inconsistent DL-Lite knowledge base. A conflict set C is a set of membership assertions such that:

- $C \subseteq \mathcal{A}$,
- $\langle \mathcal{T}, C \rangle$ is inconsistent,
- $\forall C', C' \subset C, \langle \mathcal{T}, C' \rangle$ is consistent.

We denote by $\mathcal{C}(\mathcal{A})$ the collection of conflicts in \mathcal{K} . Since \mathcal{K} is assumed to be finite, if \mathcal{K} is inconsistent then $\mathcal{C}(\mathcal{A}) \neq \emptyset$ is also finite. Moreover, as mentioned in Lemma 3.6 and 3.7 in possibilistic DL-Lite, $\forall C \in \mathcal{C}(\mathcal{A})$, it holds that $|C| = 2$. Note that this definition is in the spirit of 4.5 in prioritized case.

Let us now consider a simple case of instance-based checking, namely what is the complexity of the decision problem:

"Does $\langle \mathcal{T}, \mathcal{M}_2 \rangle \models_{\forall} B(a)$ (resp.. $R(a, b)$) holds ?"

where $B \in D_C$ (resp. $R \in D_R$), and $a \in D_I$ (resp. $a, b \in D_I$).

The complexity analysis can be easily generalized to any conjunctive query q and also to the inference relation $\langle \mathcal{M}_8, \forall \rangle$. To reach this aim, we will use complexity results which are known in graph theory regarding the problem of Maximum Independent Sets (MIS). Let us recall k -MIS the following decision problem:

"Given a symmetric graph G , is there an independent set of size (at least) k ?"

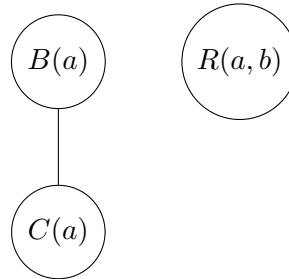
The computational complexity of k -MIS is known to be NP complete. [Garey and Johnson, 1979]. The following gives transformations between graphs and *DL-Lite* knowledge bases.

A transformation from an inconsistent *DL-Lite* knowledge base to G

Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an inconsistent *DL-Lite* knowledge base. Let $\mathcal{C}(\mathcal{A})$ be the set of all conflicts in \mathcal{A} . Recall that when \mathcal{T} is coherent, then all conflicts of \mathcal{C} are pairs of \mathcal{A} and are computed in PTime. We define a graph associated with \mathcal{K} as follows:

1. The set of nodes is simply the set of assertions in \mathcal{A} (one assertion = one different node), and
2. A non-oriented arc is drawn from f to g if there is $f \in \mathcal{A}$, $g \in \mathcal{A}$ such that (f, g) is a conflict of $\langle \mathcal{T}, \mathcal{A} \rangle$.

Example 5.13. Let us consider $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ a *DL-Lite* knowledge base where $\mathcal{T} = \{B \sqsubseteq \neg C\}$ and $\mathcal{A} = \{B(a), C(a), R(b, c)\}$. The graph associated with \mathcal{K} is :



■

Then we have the following result:

Proposition 5.3. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a *DL-Lite* knowledge base, and G be its associated graph as it is defined above. Let $\mathcal{R} \subseteq \mathcal{A}$ be a subset of \mathcal{A} and $G_{\mathcal{R}}$ be the set of nodes associated to \mathcal{R} . Then \mathcal{R} is a maximal consistent subset of \mathcal{A} if and only if $G_{\mathcal{R}}$ is a maximal independent set of G .

Proof of Proposition 5.3. The proof is immediate.

Assume that \mathcal{R} is a maximal consistent subbase of \mathcal{A} but $G_{\mathcal{R}}$ is not a maximal independent set of G . This means that there exists a node f (namely an assertion of \mathcal{A}) such that $f \notin G_{\mathcal{R}}$ and $\forall g \in G_{\mathcal{R}}$, there is no arc between f and g . Said differently, there exists an assertion $f \in \mathcal{A}$ such that $f \notin \mathcal{R}$ and $\forall g \in \mathcal{R}$, there is no conflict \mathcal{C} of the form (f, g) . This means that $\mathcal{R} \cup \{f\}$ is consistent and this contradicts the fact that \mathcal{R} is a maximally consistent subbase of \mathcal{A} .

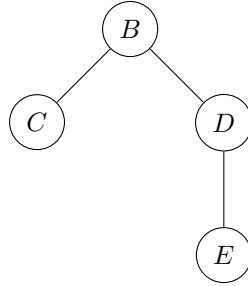
Similarly, assume that G_R is a maximal independent set of G and let us show that R (the subset of assertions present in G_R) is a maximally consistent subset of \mathcal{A} . Clearly, R is consistent, since $\forall f \in R, \forall g \in R$, we have $(f, g) \notin \mathcal{C}$ where \mathcal{C} is conflict (otherwise, there would be an arc between f and g). R is maximal, since $\forall h \notin R$ there is an arc between h and a node from G_R . Hence there is a conflict between h and an element of R , namely $R \cup \{h\}$ is inconsistent. Hence R is maximal. \square

Let us now give the converse transformation

Let G be a non-oriented graph. The DL-Lite knowledge base associated with G is defined as follows:

1. We associate to each node e a concept also denoted by e (two different nodes have two distinct associated concepts),
2. We use "a" as the unique individual used in \mathcal{A} ,
3. For each non-oriented arc $e \longrightarrow f$, we add $(e \sqsubseteq \neg f)$ to \mathcal{T} , namely the TBox associated with G is defined by: $\mathcal{T} = \{e \sqsubseteq \neg f : e \longrightarrow f \text{ is an arc of } G\}$, and
4. The ABox is simply the set of nodes with the same individual "a", namely $\mathcal{A} = \{e(a) : a \text{ is an individual and } e \text{ is a node of } G\}$.

Example 5.14. Let G be the following graph:



Then the DL-Lite knowledge base associated with G is :
 $\mathcal{T} = \{B \sqsubseteq \neg C, B \sqsubseteq \neg D, D \sqsubseteq \neg E\}$ and $\mathcal{A} = \{B(a), C(a), D(a), E(a)\}$. \blacksquare

The DL-Lite knowledge base associated with a graph only involves one individual. It neither contains positive axioms nor relation symbols.

Proposition 5.4. Let G be a non-oriented graph, and $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be the DL-Lite knowledge base associated with G , as it is defined above. Then, $\forall e(a) \in \mathcal{A}, \forall f(a) \in \mathcal{A}$, $(e(a), f(a)) \in \mathcal{C}$ if and only if there is a non-oriented arc between f and e .

Proof of Proposition 5.4. The proof is immediate. Since there is no relation symbols nor positive axioms, then the negative closure of \mathcal{T} is simply \mathcal{T} . Besides, for each $e \sqsubseteq \neg f$ of \mathcal{T} (namely, an arc from G by construction), there exists exactly one conflict $(e(a), f(a))$ from \mathcal{A} (since there is exactly one individual a). \square

Using the two above propositions, the following proposition gives the complexity of computing the cardinality of the largest maximal consistent subbase of \mathcal{A} .

Proposition 5.5. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an inconsistent DL-Lite knowledge base. The complexity of computing the cardinality of the largest maximal consistent subset of \mathcal{K} is $\mathcal{O}(\log_2(|\mathcal{A}|) * k\text{-MIS})$.

Proof of Proposition 5.5. The proof is immediate. It is enough to apply a dichotomy search between 1 and $|\mathcal{A}|$, and for each value $1 \leq k \leq |\mathcal{A}|$ we call a k -MIS problem. \square

We are now ready to give the complexity of the decision problem, denoted $(\forall\text{-card})$, namely

$(\forall\text{-card})$ "Does $\langle \mathcal{T}, \mathcal{M}_2 \rangle \models_{\forall} B(a)$ (resp. $R(a,b)$) hold?"

Proposition 5.6. *Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an inconsistent DL-Lite knowledge base. Assume that k_{max} is the size of the largest maximal subbase of \mathcal{A} . Then $(\forall\text{-card})$ is NP-complete.*

Proof of Proposition 5.6. Recall first that in standard consistent DL-Lite knowledge base $\mathcal{K}_1 = \langle \mathcal{T}_1, \mathcal{A}_1 \rangle$, checking if $\mathcal{K}_1 \models B(a)$ comes down to check if $\langle \mathcal{T} \cup \{D \sqsubseteq \neg B\}, \mathcal{A}_1 \cup \{D(a)\} \rangle$ is consistent, where a is a new individual and D is a new symbol.

Let D be a new concept symbol and a be a new individual. Let $\mathcal{K}' = \langle \mathcal{T}' = \mathcal{T} \cup \{D \sqsubseteq \neg B\}, \mathcal{A}' = \mathcal{A} \cup \{D(a)\} \rangle$ be a DL-Lite knowledge base. Let G' be the graph associated with \mathcal{K}' . Then one can check that $\langle \mathcal{T}, \circ_{cl}(\circ_{incl}(\mathcal{A})) \rangle \not\models_{\forall} B(a)$ iff G' admits a maximal independent set of size $(k_{max} + 1)$.

Indeed, assume that G' does not admit a maximal independent set of size $(k_{max} + 1)$. This means that $\forall R'$ a maximally consistent subset of \mathcal{A}' is of size k_{max} . This means that $\forall R$ a maximally consistent subset of \mathcal{A} consistent with $D(a)$ with respect to \mathcal{T} , hence $\mathcal{K} \models B(a)$. Similarly, if G' admits a maximal independent set of size $(k_{max} + 1)$. This means that $\exists R'$ a maximally consistent subset of \mathcal{A}' of size $(k_{max} + 1)$. Hence, there exists R a maximally consistent subset of \mathcal{A} with respect to \mathcal{T}' which is consistent with $D(a)$. Hence $R \models B(a)$ and therefore $\mathcal{K} \models_{\forall} B(a)$. \square

Proposition 5.7. *Computing $\langle \mathcal{M}_2, \cap \rangle$ needs $|\mathcal{A}|$ calls to a solver of k -MIS problem.*

Proof. Recall that $\langle \mathcal{M}_2, \cap \rangle = \bigcap_{A_i \in \mathcal{M}_2} A_i$ where $\mathcal{M}_2 = \mathcal{L}_{\circ_{incl}}(\mathcal{A})$.

Hence, one way to compute $\langle \mathcal{M}_2, \cap \rangle$ is to check whether each instance of \mathcal{A} is a universal consequence of \mathcal{M}_2 . Following Proposition 5.6, checking whether an instance of \mathcal{A} follows from $\langle \mathcal{M}_2, \forall \rangle$ needs exactly one call to a solver of k -MIS problem. Hence, computing $\langle \mathcal{M}_2, \cap \rangle$ needs $|\mathcal{A}|$ calls to a solver of k -MIS problem.

Similarly proof can be given for $\langle \mathcal{M}_8, \cap \rangle$, where it is enough to replace $|\mathcal{A}|$ by $|\circ_{cl}(\mathcal{A})|$. \square

As corollary, we have:

- The proof of NP completeness of $\langle \mathcal{M}_3, \forall \rangle$ follows immediately from the result of Proposition 5.6 and from the fact that $\langle \mathcal{M}_2, \forall \rangle$ is equivalent to $\langle \mathcal{M}_3, \forall \rangle$.
- Similarly, we have shown that $\langle \mathcal{M}_1, \forall \rangle$ is equivalent to $\langle \mathcal{M}_5, \forall \rangle$. Hence, the proof of the completeness of $\langle \mathcal{M}_5, \forall \rangle$ follows from the fact that $\langle \mathcal{M}_1, \forall \rangle$ has been shown to be NP-complete in [Lembo et al., 2010].
- The proof of NP-completeness of $\langle \mathcal{M}_7, \forall \rangle$ is exactly the same as the one of $\langle \mathcal{M}_1, \forall \rangle$. The main difference is rather to start with the initial ABox \mathcal{A} , one should start with its closure $\circ_{cl}(\mathcal{A})$. Note that $\langle \mathcal{M}_7, \forall \rangle$ is equivalent to CAR -entailment proposed in [Lembo et al., 2010] where it has been shown that this latter is NP-complete.
- The proof of NP-completeness of $\langle \mathcal{M}_8, \forall \rangle$ is exactly the same as the one of $\langle \mathcal{M}_2, \forall \rangle$. The main difference is rather to start with the initial ABox \mathcal{A} , one should start with its closure $\circ_{cl}(\mathcal{A})$.
- The computational complexity of $\langle \mathcal{M}_4, \forall \rangle$, $\langle \mathcal{M}_6, \forall \rangle$ comes down to enumerate all possible repairs.

We now provide a study of computational complexity of the rest of inference relations.

The majority-based inference $\langle \mathcal{M}_i, maj \rangle$ for $i = 1, \dots, 8$ are in #P (for more details on #P class of complexity, see [Valiant, 1979]), since it comes down to counting models implying the query.

We now give the complexity of safe inference relations. Note that, once $\bigcap_{\mathcal{A}_i \in \mathcal{M}_j} \mathcal{A}_i$ is computed, the query answering has a same complexity as in standard *DL-Lite*.

- The inference relation $\langle \mathcal{M}_1, \cap \rangle$ (*resp.* $\langle \mathcal{M}_7, \cap \rangle$) is in PTime, since it comes down to compute conflict set from \mathcal{A} (*resp.* $\circ_{cl}(\mathcal{A})$) and throw them out in order to obtain $\bigcap_{\mathcal{A}_i \in \mathcal{M}_1} \mathcal{A}_i$ (*resp.* $\bigcap_{\mathcal{A}_i \in \mathcal{M}_7} \mathcal{A}_i$) where the computation of conflict sets in *DL-Lite* is in PTime.
- The computational complexity of computing $\langle \mathcal{M}_2, \cap \rangle$ (*resp.* $\langle \mathcal{M}_8, \cap \rangle$) is $\mathcal{O}(|\mathcal{A}| * k\text{-MIS})$ (*resp.* $\mathcal{O}(|Cl(\mathcal{A})| * k\text{-MIS})$).

5.6 Related works and discussions

This section first gives related works and then discusses the main inference strategies studied in this chapter.

5.6.1 Related works

Handling inconsistency in ontologies is a very important problem that received a particular attention in recent years either on works on debugging or repairing generic knowledge (*e.g.* [Schlobach and Cornet, 2003; Haase *et al.*, 2005; Peñaloza and Sertkaya, 2010]) or revising ontologies (*e.g.* [Qi and Du, 2009; Wang *et al.*, 2010; Zhuang *et al.*, 2014]).

This present work is rather oriented to inconsistency handling within an OBDA setting. In [Lembo *et al.*, 2010] four inconsistency-tolerant semantics called AR, IAR, CAR and ICAR were proposed. An inconsistency-tolerant semantics corresponds in our work to the combination of an MBox composite modifier followed by an inference-based strategy. It is easy to check that AR, IAR, CAR and ICAR semantics correspond respectively to $\langle \mathcal{M}_1, \forall \rangle$, $\langle \mathcal{M}_1, \cap \rangle$, $\langle \mathcal{M}_7, \forall \rangle$ and $\langle \mathcal{M}_7, \cap \rangle$. It was shown in that CQ answering from AR and CAR is co-NP-complete in data complexity and IQ (instance checking) from CAR is in PTime, but it remains co-NP-complete under AR semantics. Besides both IQ and CQ answering under IAR and ICAR are in PTime for data complexity.

In [Bienvenu, 2012] a new semantics, called ICR, was given as a sound approximation for AR semantics. The ICR semantics corresponds to $\langle \mathcal{M}_5, \cap \rangle$ in our work.

In [Bienvenu and Rosati, 2013], two parametrized inconsistency-tolerant semantics, called k -support and k -defeater semantics, were studied for *DL-Lite_{horn}* and *DL-Lite_{core}* logics where it was shown that instance checking (*resp.* CQ answering), within *DL-Lite_{core}* framework, is NLSpace (*resp.* NP) for both k -support and k -defeater semantics. In a nutshell, a query q is said to be a k -support consequence of an inconsistent *DL-Lite* knowledge base \mathcal{K} , if there exist k consistent subsets $\{S_1, \dots, S_n\}$ of \mathcal{A} such that $\forall S_i, \langle \mathcal{T}, S_i \rangle \models q$ and $\forall \mathcal{A}_i \in \mathcal{M}_1$, there is at least an $S_i \subseteq \mathcal{A}_i$. A query is said to be a k -defeater consequence of \mathcal{K} , if there does not exist a consistent subset S of \mathcal{A} with $|S| \leq k$ such that $\langle \mathcal{T}, S \cup C \rangle$ is inconsistent where $C \subseteq \mathcal{A}$ is a minimal support of q (*i.e.* there is no proper subset of C that support q).

Another family of parametrized semantics, called k -lazy semantics, was proposed in [Lukasiewicz *et al.*, 2012b] within Datalog+/- setting. However, as mentioned in [Bienvenu and Rosati, 2013], these semantics are not a sound approximation of consistent query answering and they don't have good computational properties where CQ answering is co-NP-hard in data complexity for every $k \geq 1$.

In [Zhou *et al.*, 2012] a four-valued semantics reasoning approach, was proposed to reason under inconsistent *DL-Lite* knowledge bases. However this approach leads to derive inconsistent conclusion

with respect to generic knowledge.

5.6.2 Majority-based inference within *DL-Lite* framework

In the context of the use of majority-based inference for managing inconsistent set of information, there is one main and major advantage of using *DL-Lite* language rather than expressive DLs or propositional logic: It concerns the consistency of the set of derived conclusions. To illustrate our proposal, let us focus on instance checking problem and let consider $\langle \mathcal{M}_1, \text{maj} \rangle$. which represents majority-based inference over the set of maximally consistent subset of an inconsistent knowledge base. Then we have:

Proposition 5.8. *Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an inconsistent standard *DL-Lite* knowledge base. Then*

$$\begin{aligned} \text{Res}(\mathcal{K}) = & \{A(a) : a \in D_I, A \in D_C, \langle \mathcal{M}_1, \text{Maj} \rangle \models A(a)\} \cup \\ & \{R(a, b) : a, b \in D_I, R \in D_R, \langle \mathcal{M}_1, \text{Maj} \rangle \models R(a, b)\} \end{aligned}$$

is consistent with respect to \mathcal{T} , where D_I , D_C and D_R are given in Def.5.3.

Note that, Proposition 5.8 does not hold with general DLs. Assume that we have a DL language where ' \sqsupseteq ' is allowed in the left-side of inclusion axioms. Consider the following knowledge base

$$\mathcal{K} = \langle \mathcal{T} = \{A \sqcap B \sqsubseteq \neg C, A \sqcap C \sqsubseteq \neg B, B \sqcap C \sqsubseteq \neg A\}, \mathcal{A} = \{A(a), B(a), C(a)\} \rangle.$$

Clearly, \mathcal{K} admits three maximally consistent subsets, namely $\mathcal{M}_1 = \{A(a), B(a)\}$, $\mathcal{M}_2 = \{A(a), C(a)\}$, $\mathcal{M}_3 = \{B(a), C(a)\}$. Clearly $\langle \mathcal{T}, \mathcal{M}_1 \rangle \models_{\text{maj}} A(a)$, $\langle \mathcal{T}, \mathcal{M}_1 \rangle \models_{\text{maj}} B(a)$ and $\langle \mathcal{T}, \mathcal{M}_1 \rangle \not\models_{\text{maj}} C(a)$, while the set $\{A(a), B(a), C(a)\}$ is consistent with \mathcal{T} . A similar example can be provided in a propositional logic setting.

5.6.3 Which inference relations are appropriate to deal with inconsistent *DL-Lite* knowledge bases?

Intuitively, an appropriate method is such that:

- i Outputs a single ABox, or a limited set of ABox's,
- ii Produces as many safe conclusions as possible, and
- iii the size of the produced ABox should be polynomial with respect to the initial ABox.

Clearly majority-based and universal inference do not fit item (i) and (iii). If one is interested in keeping inference relations that avoid deriving contestable conclusions, then inference $\langle \mathcal{M}_7, s \rangle$ and $\langle \mathcal{M}_8, s \rangle$, with $s \in \{\forall, \text{maj}, \cap\}$, should be discarded. Indeed, considering first the closure, may lead to consider as plausible a conclusion with a contestable support. For instance, let $\mathcal{K} = \langle \{A \sqsubseteq \neg B, A \sqsubseteq C\}, \mathcal{M} = \{A(a), B(a)\} \rangle$. Clearly, $\langle \mathcal{T}, \mathcal{M}_7 \rangle \models_s C(a)$ (*resp.* $\langle \mathcal{T}, \mathcal{M}_8 \rangle \models_s C(a)$). This conclusion $C(a)$ is obtained using $A(a)$, but $A(a)$ is contestable and it is not a conclusion of $\langle \mathcal{T}, \mathcal{M}_7 \rangle$ (*resp.* $\langle \mathcal{T}, \mathcal{M}_8 \rangle$).

The inclusion-based criterion (namely \mathcal{M}_1) followed by universal entailment is widely used in the literature. The cardinality-based criterion may make sense in some applications where counting falsified formulas is important. For instance, consider an ABox encoding facts about women and men (distinguished by the TBox) attending a conference. From the point of view of the output, safe inferences $\langle \mathcal{M}_i, \cap \rangle$ are appropriate for handling inconsistency in *DL-Lite*. Safe inferences using different \mathcal{M}_i 's, fit at the requirement (i) and (iii). The inference $\langle \mathcal{M}_1, \cap \rangle$ is considered as safe since it is equivalent to consider a subbase of the initial ABox obtained by ignoring all conflicting facts, namely:

$$\mathcal{A}_r = \bigcap_{\mathcal{A}_i \in \mathcal{M}_1} \mathcal{A}_i = \{f : f \in \mathcal{A} \text{ and } \nexists \mathcal{C} \in \mathcal{C}(\mathcal{A}) \text{ such that } f \in \mathcal{C}\}$$

where $\mathcal{C}(\mathcal{A})$ is the set of conflicts. The main advantage of this approach is that computing \mathcal{A}_r is done in linear time. However, the main limitation is that its associated inference relation is very cautious. To get a more productive one and to better fit requirement (ii), one may take the intersection of largest consistent sets from the ABox, namely, $\mathcal{A}_r = \bigcap_{\mathcal{A}_i \in \mathcal{M}_j} \mathcal{A}_i, j = 2, \dots, 6$. The advantage of this approach is that the obtained inference relation is productive and its space complexity is $|D_I| \times |D_C| + |D_I|^2 \times |D_R|$, in the worst case.

5.7 Conclusion

This chapter considered an MBox as a result of modifying inconsistent standard *DL-Lite* knowledge base. We generalize techniques for non-merge inconsistency management approaches in flat knowledge bases by introducing the notions of (1) modifier and (2) inference strategy. The combination modifiers plus inference strategies can be mapped out in order to provide a principled and exhaustive list of techniques for inconsistency management. We study the productivity and give a complexity analysis for such techniques in the case of flat knowledge bases and show how our work extends the state of the art.

The last part of this thesis concerns dynamics and handling inconsistency in prioritized assertional *DL-Lite* knowledge bases.

PART IV

**On the revision of prioritized *DL-Lite*
knowledge bases**

PRIORITIZED ASSERTIONAL-BASED REMOVED SETS REVISION OF *DL-Lite* KNOWLEDGE BASES

6.1 Introduction

Dynamics of a DL-based knowledge bases gave rise to increasing interest (e.g. [Qi *et al.*, 2006c; Qi and Du, 2009]) and often concerns the situation where new information should be incorporated while ensuring the consistency of the result. Several works have recently dealt with revising *DL-Lite* TBox with a terminological information (e.g. [Wang *et al.*, 2010; Calvanese *et al.*, 2010]) or with an assertional information (e.g. [Calvanese *et al.*, 2010; Kharlamov and Zheleznyakov, 2011; Gao *et al.*, 2012]) as input.

Besides, data are often provided by several and potentially conflicting sources. Concatenating them gives a prioritized or a stratified ABox. This stratification generally results from two situations as pointed out in [Benferhat *et al.*, 1995; Benferhat *et al.*, 1998b]. The first one is when each source provides its set of data without any priority between them, but there exists a total pre-ordering between different sources reflecting their reliability. The other one is when the sources are considered as equally reliable (i.e. having the same reliability level), but there exists a preference ranking between the set of provided data according to their level of certainty. The role of priorities in belief revision is very important and it was largely studied in the literature where knowledge bases are encoded in a propositional logic setting (e.g. [Benferhat *et al.*, 2002c; Benferhat *et al.*, 2010b]). The notion of priorities in DLs is used in (e.g. [Baader and Hollunder, 1995; Qi *et al.*, 2006a; Qi and Pan, 2007]) to deal with default terminologies while assuming that the ABox is completely sure. However, as far as we know, revising prioritized *DL-Lite* knowledge bases has not been addressed so far.

Belief revision has been largely considered in the literature when knowledge bases are encoded using a propositional language. Among these revision approaches the so-called Removed Sets Revision, also known as a cardinality-based approach, has been proposed in [Papini, 1992; Benferhat *et al.*, 1993a] for revising a set of propositional formulas. This approach stems from removing a minimal number of formulas, called removed set, to restore consistency. The minimality in Removed Sets Revision refers to the cardinality criterion and not to the set-inclusion one. This approach has interesting properties: it has not a high computational complexity, it is not too cautious and satisfies all rational AGM postulates when extended to belief sets revision.

This chapter studies Prioritized Removed Sets Revision (PRSR), when knowledge bases are described in *DL-Lite* logics. One of the motivations in considering PRSR is to take advantage of tractability of *DL-Lite* for the revision process as well as of rational properties satisfied by PRSR. In particular, we investigate the well-known *DL-Lite_R* logic which offers a good compromise between expressive power and computational complexity. We consider different forms of input: a membership assertion, a positive inclusion axiom or a negative inclusion axiom, since they lead to different revision problems, different algorithms and different complexity results. A crucially important problem that arises when revising

a *DL-Lite* knowledge base is how to restore consistency. In this chapter restoring consistency leads to ignoring some assertions, namely we give a priority to TBox over ABox. Another important feature when dealing with *DL-Lite* knowledge base is that computing the set of minimal information responsible of inconsistency can be done in polynomial time. Besides minimal assertional sets that cause inconsistency are either singletons or doubletons. This is helpful in defining removed sets necessary to restore consistency in presence of new information.

The rest of this chapter is organized as follows. Section 6.2 studies Prioritized Removed Sets Revision within this framework when priorities between assertional facts are available. Section 6.3 reformulates the well-known Hansson’s postulates defined for propositional belief bases revision within a *DL-Lite* setting and gives logical properties of PRSR operators. Section 6.4 provides algorithms for computing prioritized removed sets through the use of hitting sets. Finally, Section 6.5 presents some related works and Section 6.6 concludes the chapter.

6.2 Assertional-based revision of *DL-Lite* knowledge bases

In this section, we investigate the revision of *DL-Lite* knowledge bases in the case where priorities are available between assertions in the ABox. We study different forms of the input: An assertion, a positive inclusion axiom or a negative inclusion axiom. We consider a lexicographical strategy where only smallest subsets of assertions should be dropped from the knowledge base in order to restore its consistency and accept the new piece of information. Note that the choice of dropping information only from the ABox is motivated by the fact that in many applications (such as in ontology-based data access applications) a TBox is often seen as a well-formed and coherent ontology whereas the ABox represents data that are not necessarily reliable and consistent with the ontology. In other words, when the input is a terminological information, the revising process comes down to enrich the ontology while preserving the coherence of the resulting TBox. However, in case of inconsistency, the ABox may be modified in order to take into account the input.

Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a consistent *DL-Lite* knowledge base. Let us denote by N a new consistent information to be accepted. The presence of this new information may lead to inconsistency according to the content of the TBox and the nature of the input information.

Within the *DL-Lite* language, the new piece of information N may be :

- A membership assertion of the form $A(a)$ or $P(a, b)$,
- A positive inclusion axiom (PI) of the form $B_1 \sqsubseteq B_2$ or
- A negative inclusion axiom (NI) of the form $B_1 \sqsubseteq \neg B_2$.

We assume that the input is consistent with the terminological base (otherwise, the input will be simply ignored. According to [Calvanese *et al.*, 2007a], every *DL-Lite* knowledge base \mathcal{K} with only PIs in its TBox is always satisfiable (consequence of Lemma 7 in [Calvanese *et al.*, 2007a]). Hence, if N is a membership assertion or a PI axiom, there is no inconsistency. However when the TBox \mathcal{T} contains NI axioms then N may have an undesirable interaction with \mathcal{K} and which leads to an inconsistency.

We use $\mathcal{C}(\mathcal{A})$ the collection of conflicts in \mathcal{K} as defined in Definition 5.12. Recall that when \mathcal{T} is coherent, then $\forall C \in \mathcal{C}(\mathcal{A}), |C| = 2$.

Example 6.1. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an inconsistent knowledge base such that $\mathcal{T} = \{B_1 \sqsubseteq B_2, B_2 \sqsubseteq \neg B_3\}$ and $\mathcal{A} = \{B_1(a), B_3(a), B_2(b), B_3(b), B_1(c)\}$. We have $cln(\mathcal{K}) = \{B_2 \sqsubseteq \neg B_3, B_1 \sqsubseteq \neg B_3\}$. Then $\mathcal{C}(\mathcal{A}) = \{\{B_1(a), B_3(a)\}, \{B_2(b), B_3(b)\}\}$. ■

6.2.1 Prioritized DL-Lite assertional base

We use the notion of a *DL-Lite* prioritized knowledge base, simply denoted by $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$. We assume that \mathcal{T} is coherent and not stratified. Namely, all elements of \mathcal{T} have the same level of importance. On contrast, the ABox is assumed to be stratified, *i.e.* partitioned into n strata, $\mathcal{A} = \mathcal{A}_1 \cup \dots \cup \mathcal{A}_n$ such that:

- The strata are pairwise disjoint, namely $\forall \mathcal{A}_i, \forall \mathcal{A}_j : \mathcal{A}_i \cap \mathcal{A}_j = \emptyset$,
- The assertions in \mathcal{A}_i have the same level of priority,
- The assertions of \mathcal{A}_i have higher priority than the ones in \mathcal{A}_j where $j > i$. Hence assertions in \mathcal{A}_1 are the most important ones, while assertions in \mathcal{A}_n are the least important ones.

We first define the lexicographic preference relation between subsets of the ABox as follows.

Definition 6.1. let X and X' be two subsets of \mathcal{A} . X is strictly preferred to X' , denoted by $X <_{lex} X'$, if and only if, there exists i , $1 \leq i \leq n$ such that:

- $|X \cap \mathcal{A}_i| < |X' \cap \mathcal{A}_i|$, and
- $\forall j, 1 \leq j < i, |X \cap \mathcal{A}_j| = |X' \cap \mathcal{A}_j|$.

Similarly, X is equally preferred to X' , denoted by $X =_{lex} X'$, if and only if $\forall i, 1 \leq i \leq n, |X \cap \mathcal{A}_i| = |X' \cap \mathcal{A}_i|$. Lastly, X is at least as preferred as X' , denoted by $X \leq_{lex} X'$, if and only if $X <_{lex} X'$ or $X =_{lex} X'$. The relation \leq_{lex} is a total pre-order.

Example 6.2. Let \mathcal{A} be a stratified ABox, $\mathcal{A} = \mathcal{A}_1 \cup \mathcal{A}_2 \cup \mathcal{A}_3$ where $\mathcal{A}_1 = \{B_1(a)\}$, $\mathcal{A}_2 = \{B_2(b)\}$ and $\mathcal{A}_3 = \{B_3(a), B_3(b)\}$. Let $X = \{B_3(a), B_3(b)\}$ and $X' = \{B_3(a), B_2(b)\}$ be two subsets of \mathcal{A} , we have $X <_{lex} X'$. ■

6.2.2 Prioritized Removed Sets Revision of DL-Lite knowledge bases

We now investigate the revision of *DL-Lite* knowledge bases according to the nature of the input information. We consider an approach using a lexicographical strategy well-known as "Prioritized Removed Sets Revision" (PRSR) [Benferhat *et al.*, 2010a] proposed within a propositional logic setting.

Within the *DL-Lite* framework, in order to restore consistency while keeping new information, the Prioritized Removed Sets Revision strategy removes exactly one assertion in each conflict minimizing the minimum number of assertions from \mathcal{A}_1 , then the minimum number of assertions in \mathcal{A}_2 , and so on. Using lexicographic criterion instead of set inclusion one, will reduce the set of potential conflicts.

Note that taking the stratification of the ABox into account has not been considered before for revising or repairing *DL-Lite* knowledge bases. Next Chapter (Chapter 7) investigates repairing *DL-Lite* knowledge bases when the ABox is layered.

Revision by a membership assertion

We first consider the case where N is an ABox assertion which corresponds to the revision by a fact or by an observation. In this case, N is added to a new stratum having the highest and a new priority. However, in order to avoid heavy notations, we simply write $\mathcal{K} \cup \{N\}$ or $\langle \mathcal{T}, \mathcal{A} \cup \{N\} \rangle$ where \mathcal{A} is a prioritized ABox, to denote the fact that N is added to a new and highest priority stratum of \mathcal{A} .

The following definition introduces the concept of prioritized removed sets.

Definition 6.2. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a consistent stratified knowledge base and N be a membership assertion. A *prioritized removed set*, denoted by X , is a set of membership assertions such that:

- $X \subseteq \mathcal{A}$,
- $\langle \mathcal{T}, (\mathcal{A} \setminus X) \cup \{N\} \rangle$ is consistent,
- $\forall X' \subseteq \mathcal{A}$, if $\langle \mathcal{T}, (\mathcal{A} \setminus X') \cup \{N\} \rangle$ is consistent then $X \leq_{lex} X'$.

We denote by $\mathcal{PR}(\mathcal{K} \cup \{N\})$ the set of all prioritized removed sets of $\mathcal{K} \cup \{N\}$. If $\mathcal{K} \cup \{N\}$ is consistent then $\mathcal{PR}(\mathcal{K} \cup \{N\}) = \{\emptyset\}$. Besides, if $\mathcal{K} \cup \{N\}$ is inconsistent then every conflict C of $\mathcal{K} \cup \{N\}$ contains N . More formally.

Lemma 6.1. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a consistent prioritized knowledge base and N be an assertion. If $\mathcal{K} \cup \{N\}$ is inconsistent then $\forall C \in \mathcal{C}(\mathcal{A})$ it holds that $N \in C$.

Proof of Lemma 6.1. The proof is immediate. If $\mathcal{K} \cup \{N\}$ is inconsistent then $\mathcal{C}(\mathcal{K} \cup \{N\}) \neq \emptyset$. This means that there exists at least a conflict $C = (\alpha, \beta) \in \mathcal{C}(\mathcal{K} \cup \{N\})$ (recall that $|C| = 2$). Let C be a conflict of $\mathcal{K} \cup \{N\}$. Suppose that $N \notin C$. This means that $\alpha \in \mathcal{A}$ and $\beta \in \mathcal{A}$. This is a contradiction since \mathcal{K} is assumed to be consistent, namely $\mathcal{C}(\mathcal{A}) = \emptyset$. \square

As consequence, there exists exactly one prioritized removed set. More formally.

Proposition 6.1. Let \mathcal{K} be a consistent stratified knowledge base and N be a membership assertion. If $\mathcal{K} \cup \{N\}$ is inconsistent then $|\mathcal{PR}(\mathcal{K} \cup \{N\})| = 1$.

Proof of Proposition 6.1. Suppose that there are two prioritized removed sets X and X' such that $X \neq X'$. By Definition 6.2, $X \subseteq \mathcal{A}$, $X' \subseteq \mathcal{A}$ and $X =_{lex} X'$. Since $(\mathcal{T} \cup \{N\}) \cup (\mathcal{A} \setminus X)$ and $(\mathcal{T} \cup \{N\}) \cup (\mathcal{A} \setminus X')$ are consistent, we have $\forall C \in \mathcal{C}(\mathcal{K} \cup \{N\})$ on one hand $C \cap X \neq \emptyset$ and $|C \cap X| = 1$ and on the other hand $C \cap X' \neq \emptyset$ and $|C \cap X'| = 1$. Moreover, since N is a single assertion, by Lemma 6.1, $|C \cap N| = 1$. Therefore there are three elements in C namely N , one element of X and one element of X' . Hence, this contradicts Lemma 3.6 that states that $|C \cap \mathcal{A}| \leq 2$. \square

Definition 6.3. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a consistent stratified knowledge base and N be a membership assertion. The revised knowledge base $\mathcal{K} \circ_{PRSR} N$ is such that $\mathcal{K} \circ_{PRSR} N = \langle \mathcal{T}, \mathcal{A} \circ_{PRSR} N \rangle$ where $\mathcal{A} \circ_{PRSR} N = (\mathcal{A} \setminus X) \cup \{N\}$ with $X \in \mathcal{PR}(\mathcal{K} \cup \{N\})$.

Example 6.3. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a consistent stratified knowledge base such that:

$\mathcal{T} = \{B_1 \sqsubseteq B_2, B_2 \sqsubseteq \neg B_3, B_3 \sqsubseteq \neg B_4\}$ and $\mathcal{A} = \mathcal{A}_1 \cup \mathcal{A}_2 \cup \mathcal{A}_3$ where $\mathcal{A}_1 = \{B_1(a)\}$, $\mathcal{A}_2 = \{B_3(b)\}$, and $\mathcal{A}_3 = \{B_4(a)\}$.

Let $N = B_3(a)$ then $\mathcal{K} \cup \{N\}$ is inconsistent. By Definition 5.12, $\mathcal{C}(\mathcal{K} \cup \{N\}) = \{\{B_1(a), B_3(a)\}, \{B_3(a), B_4(a)\}\}$.

Hence by Definition 6.2, $\mathcal{PR}(\mathcal{K} \cup \{N\}) = \{\{B_1(a), B_4(a)\}\}$. Therefore $\mathcal{A} \circ_{PRSR} N = \{B_3(a), B_3(b)\}$ and $\mathcal{A} \circ_{PRSR} N = \mathcal{A}'_1 \cup \mathcal{A}'_2 \cup \mathcal{A}'_3$ where $\mathcal{A}'_1 = \{B_3(a)\}$, $\mathcal{A}'_2 = \{B_3(b)\}$ and $\mathcal{A}'_3 = \emptyset$. \blacksquare

As detailed in Section 6.4 (precisely, subsection 6.4.1) computing the set of conflicts is polynomial. Moreover when the input information is a membership assertion, as stated by Proposition 6.1 and illustrated in the above example, there is only one prioritized removed set. Next subsection investigates the case where the input information is a positive or a negative inclusion axiom.

Revision by a positive or a negative axiom

We now consider the case where the input N is a PI axiom or a NI axiom. This new axiom should be added to the TBox and since we gave priority to the TBox over the ABox, the input is kept in the revised knowledge base. In this case, $\mathcal{K} \cup \{N\}$ denotes $\langle \mathcal{T} \cup \{N\}, \mathcal{A} \rangle$. Since \mathcal{T} is considered as non prioritized, then $\mathcal{T} \cup \{N\}$ simply denotes a simple addition of N to \mathcal{T} .

Definition 6.4. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a consistent stratified knowledge base and N be a PI or a NI axiom. A prioritized removed set, denoted by X , is a set of assertions such that:

- $X \subseteq \mathcal{A}$,
- $\langle \mathcal{T} \cup \{N\}, (\mathcal{A} \setminus X) \rangle$ is consistent and
- $\forall X' \subseteq \mathcal{A}$, if $\langle \mathcal{T} \cup \{N\}, (\mathcal{A} \setminus X') \rangle$ is consistent then $X \leq_{lex} X'$.

Let us point out that Definition 6.4 is similar to Definition 6.2, except that new information is not added to the ABox but to the TBox. However, the revision process still considers the TBox as a stable knowledge, and hence to restore consistency assertional elements from ABox should be removed. We denote again by $\mathcal{PR}(\mathcal{K} \cup \{N\})$ the set of prioritized removed sets of $\mathcal{K} \cup \{N\}$.

Example 6.4. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a consistent stratified knowledge base such that

$\mathcal{T} = \{B_1 \sqsubseteq B_2, B_3 \sqsubseteq \neg B_4\}$ and

$\mathcal{A} = \mathcal{A}_1 \cup \mathcal{A}_2 \cup \mathcal{A}_3$ where $\mathcal{A}_1 = \{B_1(a)\}$, $\mathcal{A}_2 = \{B_2(b)\}$, and $\mathcal{A}_3 = \{B_3(a), B_3(b)\}$.

Let $N = B_2 \sqsubseteq \neg B_3$ then $\mathcal{K} \cup \{N\}$ is inconsistent. $\mathcal{C}(\mathcal{K} \cup \{N\}) = \{\{B_1(a), B_3(a)\}, \{B_2(b), B_3(b)\}\}$.

The four possible candidates to be removed are:

$X_1 = \{B_1(a), B_2(b)\}$,

$X_2 = \{B_1(a), B_3(b)\}$,

$X_3 = \{B_3(a), B_2(b)\}$, and

$X_4 = \{B_3(a), B_3(b)\}$.

There is only one prioritized removed set X_4 as illustrated in Table 6.1.

\mathcal{A}_i	$ X_1 \cap \mathcal{A}_i $	$ X_2 \cap \mathcal{A}_i $	$ X_3 \cap \mathcal{A}_i $	$ X_4 \cap \mathcal{A}_i $
\mathcal{A}_3	0	1	1	2
\mathcal{A}_2	1	0	1	0
\mathcal{A}_1	1	1	0	0

Table 6.1: One prioritized removed set.

If the stratification of \mathcal{A} , now is $\mathcal{A}_1 = \{B_1(a), B_3(a)\}$, $\mathcal{A}_2 = \{B_2(b)\}$, $\mathcal{A}_3 = \{B_3(b)\}$, then there are two prioritized removed sets X_2 and X_4 as illustrated in Table 6.2.

\mathcal{A}_i	$ X_1 \cap \mathcal{A}_i $	$ X_2 \cap \mathcal{A}_i $	$ X_3 \cap \mathcal{A}_i $	$ X_4 \cap \mathcal{A}_i $
\mathcal{A}_3	0	1	0	1
\mathcal{A}_2	1	0	1	0
\mathcal{A}_1	1	1	1	1

Table 6.2: Two prioritized removed sets.

■

We have seen that when the input is a membership assertion then there exists exactly one prioritized removed set. However, when the input information is a NI or a PI axiom there may exist one or several prioritized removed sets, as illustrated in the previous example. The first case to consider, which is also the easiest one, is when each conflict intersects two distinct strata: then, there exists only one prioritized removed set. More formally,

Proposition 6.2. *If for each $C \in \mathcal{C}(\mathcal{K} \cup \{N\})$ there exists i and j , $i \neq j$, such that $C \cap \mathcal{A}_i \neq \emptyset$ and $C \cap \mathcal{A}_j \neq \emptyset$ then $|\mathcal{PR}(\mathcal{K} \cup \{N\})| = 1$.*

Proof of proposition 6.2. Suppose there are two prioritized removed sets, X and X' and $X \neq X'$. By Definition 6.4, $X \subseteq \mathcal{A}$, $X' \subseteq \mathcal{A}$, and $X =_{lex} X'$. Since $(\mathcal{T} \cup \{N\}) \cup (\mathcal{A} \setminus X)$ and $(\mathcal{T} \cup \{N\}) \cup (\mathcal{A} \setminus X')$ are consistent, $\forall C \in \mathcal{C}(\mathcal{K} \cup \{N\})$ we have $C \cap X \neq \emptyset$ and $C \cap X' \neq \emptyset$. If $|C \cap X| = 2$ (resp. $|C \cap X'| = 2$) then X (resp. X') is not a prioritized removed set, since C is a minimal inconsistent subset with two elements by Lemma 3.6. If $|C \cap X| = 1$ and $|C \cap X'| = 1$ two cases hold. If $C \cap X \neq C \cap X'$ since there exists i and j , $i \neq j$, such that $C \cap \mathcal{A}_i \neq \emptyset$ and $C \cap \mathcal{A}_j \neq \emptyset$ it contradicts $X =_{lex} X'$. If $C \cap X = C \cap X'$, since C intersects two strata, and $|C \cap X| = |C \cap X'| = 1$ then $X = X'$ which contradicts the hypothesis. \square

This situation holds when each stratum is consistent with $\mathcal{T} \cup \{N\}$ for example, when the stratification comes from several experts with different degrees of reliability. In this case, as detailed in Section 6.4.2 computing the unique prioritized removed set is polynomial.

There may be several prioritized removed sets as soon as there are conflicts included in a stratum where each conflict may lead to two prioritized removed sets. Namely, let NC be the number of conflicts such that each one is included in a stratum, the number of prioritized removed sets is bounded by 2^{NC} . In such case, each prioritized removed set leads to a possible revised knowledge base: $\mathcal{K}_i = \langle \mathcal{T} \cup \{N\}, (\mathcal{A} \setminus X_i) \rangle$ with $X_i \in \mathcal{PR}(\mathcal{K} \cup \{N\})$.

In *DL-Lite* language it is not possible to find a knowledge base that represents the disjunction of such possible revised knowledge bases. If we want to keep the result of revision in *DL-Lite* one can define a selection function that selects from $\mathcal{PR}(\mathcal{K} \cup \{N\})$ one or several prioritized removed sets. More formally,

Definition 6.5. A selection function f is a mapping from $\mathcal{PR}(\mathcal{K} \cup \{N\})$ to \mathcal{A} such that:

- $f(\mathcal{PR}(\mathcal{K} \cup \{N\})) \subseteq \mathcal{A}$
- $\exists X_i \in \mathcal{PR}(\mathcal{K} \cup \{N\})$ such that $X_i \subseteq f(\mathcal{PR}(\mathcal{K} \cup \{N\}))$
- $f(\mathcal{PR}(\mathcal{K} \cup \{N\})) \subseteq \bigcup_{X_i \in \mathcal{PR}(\mathcal{K} \cup \{N\})} X_i$

The first item in Definition 6.5 simply states that $f(\mathcal{PR}(\mathcal{K} \cup \{N\}))$ should only contain elements of \mathcal{A} . This condition guarantees that the result of revision will be within the *DL-Lite* language. The second item states that at least one prioritized removed set should be in $f(\mathcal{PR}(\mathcal{K} \cup \{N\}))$. This guarantees that $\langle \mathcal{T}, \mathcal{A} \setminus f(\mathcal{PR}(\mathcal{K} \cup \{N\})) \rangle$ is consistent. The last item states that only elements from $\bigcup_{X_i \in \mathcal{PR}(\mathcal{K} \cup \{N\})} X_i$ should be removed and ignored to restore consistency. Hence, elements which are not responsible of conflicts will not be removed.

We now define the revised knowledge base as follows.

Definition 6.6. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a consistent and stratified knowledge base and N be a PI or a NI axiom. Let f be a selection function, the revised knowledge base $\mathcal{K} \circ_{PRSR} N$ is such that $\mathcal{K} \circ_{PRSR} N = \langle \mathcal{T} \cup \{N\}, \mathcal{A} \circ_{PRSR} N \rangle$ where $\mathcal{A} \circ_{PRSR} N = (\mathcal{A} \setminus f(\mathcal{PR}(\mathcal{K} \cup \{N\})))$.

In the next subsection, we present some examples of selection functions.

6.2.3 Examples of selection functions

The definition of selection functions can be seen as the application of modifiers (presented in Chapter 5) on an MBox (resulting from the revision process) followed by inference based strategy.

Let us first start with two basic selection functions, denoted simply by f_1 and f_2 . The first selection function f_1 consists in taking all prioritized removed sets. More formally,

$$f_1(\mathcal{PR}(\mathcal{K} \cup \{N\})) = \bigcup_{X_i \in \mathcal{PR}(\mathcal{K} \cup \{N\})} X_i$$

which corresponds to the intersection of all possible revised knowledge bases. In this case $\mathcal{K} \circ_{PRSR} N = \langle \mathcal{T} \cup \{N\}, \mathcal{A} \circ_{PRSR} N \rangle$ where $\mathcal{A} \circ_{PRSR} N = \mathcal{A} \setminus f_1(\mathcal{PR}(\mathcal{K} \cup \{N\})) = \bigcap_{i=1}^n (\mathcal{A} \setminus X_i)$. This first selection function may be too cautious since it could remove too many assertions and is not in agreement with the minimal change principle.

Another option is to choose a selection function that only picks one prioritized removed set, more formally,

$$f_2(\mathcal{PR}(\mathcal{K} \cup \{N\})) = X_i$$

which corresponds to the choice of only one revised knowledge base. This option is less cautious than the previous one and captures, in some sense, the existence of a possibility for restoring consistency.

Example 6.5. Let us consider the knowledge base of Example 4. We have

$\mathcal{T} = \{B_1 \sqsubseteq B_2, B_3 \sqsubseteq \neg B_4\}$ and

$\mathcal{A} = \mathcal{A}_1 \cup \mathcal{A}_2 \cup \mathcal{A}_3$ where $\mathcal{A}_1 = \{B_1(a), B_3(a)\}$, $\mathcal{A}_2 = \{B_2(b)\}$ and $\mathcal{A}_3 = \{B_3(b)\}$.

Let $N = B_2 \sqsubseteq \neg B_3$ be a new piece of information. We have $\mathcal{K} \cup \{N\}$ is inconsistent. The prioritized removed sets are: $X_1 = \{B_1(a), B_3(b)\}$ and $X_2 = \{B_3(a), B_3(b)\}$. We have:

$f_1(\mathcal{PR}(\mathcal{K} \cup \{N\})) = \{B_1(a), B_3(b), B_3(a)\}$ and

$f_2(\mathcal{PR}(\mathcal{K} \cup \{N\}))$ can be either $\{B_1(a), B_3(b)\}$ or $\{B_3(a), B_3(b)\}$. ■

The third example of selection function, denoted by f_3 , is strongly related to the notion of universal or skeptical inference that can be defined from $\mathcal{PR}(\mathcal{K} \cup \{N\})$. Namely, we first need to define the set of all possible assertions that can be derived from each $\mathcal{A} \setminus X_i$ with $X_i \in \mathcal{PR}(\mathcal{K} \cup \{N\})$.

More precisely, let D_C be the set of concepts of \mathcal{T} , D_R be the set of roles of \mathcal{T} and D_I be the set of individuals of \mathcal{A} . Then we define the set of universal assertional consequences, denoted $UAC(\mathcal{K} \cup \{N\})$ as :

$$\begin{aligned} UAC(\mathcal{K} \cup \{N\}) = & \{A(a) : a \in D_I, A \in D_C \text{ and} \\ & \forall X_i \in \mathcal{PR}(\mathcal{K} \cup \{N\}), \langle \mathcal{T}, \mathcal{A} \setminus X_i \rangle \models A(a)\} \\ & \cup \\ & \{R(a, b) : a \in D_I, b \in D_I, R \in D_R \text{ and} \\ & \forall X_i \in \mathcal{PR}(\mathcal{K} \cup \{N\}), \langle \mathcal{T}, \mathcal{A} \setminus X_i \rangle \models R(a, b)\} \end{aligned}$$

The selection function f_3 is then simply defined by :

$$f_3(\mathcal{PR}(\mathcal{K} \cup \{N\})) = \mathcal{A} \setminus UAC(\mathcal{K} \cup \{N\})$$

Example 6.6. Let us consider $\mathcal{T} = \{A \sqsubseteq B, C \sqsubseteq B\}$ and $\mathcal{A} = \mathcal{A}_1$ where $\mathcal{A}_1 = \{B(b), A(a), C(a)\}$. Let $N = A \sqsubseteq \neg C$ be a new piece of information. We have $\mathcal{K} \cup \{N\}$ is inconsistent. The two possible

prioritized removed sets that can be computed are: $X_1 = \{A(a)\}$ and $X_2 = \{C(a)\}$.
 One can check that $UAC(\mathcal{K} \cup \{N\}) = \{B(a), B(b)\}$. Hence $f_3 = \{A(a), C(a)\}$. ■

The last selection function uses the notion of deductive closure using Definition 5.2. Using the notion deductive closure, one can refine the set of prioritized removed sets in which a selection function operates. This new subset, denoted $\mathcal{CPR}(\mathcal{K} \cup \{N\})$, is made by keeping only prioritized removed sets X in $\mathcal{PR}(\mathcal{K} \cup \{N\})$ such that the deductive closure of the set $\mathcal{A} \setminus X$ is maximal with respect to lexicographical criterion. More formally,

Definition 6.7. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a DL-Lite knowledge base and N be a new information. Let $\mathcal{PR}(\mathcal{K} \cup \{N\})$ be the set of prioritized removed sets. The set $\mathcal{CPR}(\mathcal{K} \cup \{N\})$ is composed of prioritized removed sets X from $\mathcal{PR}(\mathcal{K} \cup \{N\})$ such that $\nexists Y \in \mathcal{PR}(\mathcal{K} \cup \{N\}), |\circ_{cl}(\mathcal{A} \setminus Y)| \geq |\circ_{cl}(\mathcal{A} \setminus X)|$.

Then the last selection function, denoted by f_4 and based on the deductive closure, is simply defined by :

$$f_4(\mathcal{PR}(\mathcal{K} \cup \{N\})) = \bigcup_{X_i \in \mathcal{CPR}(\mathcal{K} \cup \{N\})} X_i$$

Clearly, $\mathcal{CPR}(\mathcal{K} \cup \{N\}) \subseteq \mathcal{PR}(\mathcal{K} \cup \{N\})$ then we have $f_2(\mathcal{PR}(\mathcal{K} \cup \{N\})) \subseteq f_3(\mathcal{PR}(\mathcal{K} \cup \{N\})) \subseteq f_1(\mathcal{PR}(\mathcal{K} \cup \{N\}))$. $f_4(\mathcal{PR}(\mathcal{K} \cup \{N\}))$ offers a good compromise between an arbitrary choice of the prioritized removed set to be ignored from the ABox \mathcal{A} , and a skeptical choice where all prioritized removed sets are ignored from the ABox.

Example 6.7. From Example 6.5, one can check that:

$\circ_{cl}(\mathcal{A} \setminus X_1) = \{B_3(a), B_2(b)\}$, and
 $\circ_{cl}(\mathcal{A} \setminus X_2) = \{B_1(a), B_2(a), B_2(b)\}$.
 Then $\mathcal{CPR}(\mathcal{K} \cup \{N\}) = \{X_2\}$. ■

6.2.4 Multiple revision

In the pervious sections, it is assumed that the input information is only composed of a single element: An assertional fact, a positive axiom or a negative axiom. This section briefly discusses the case where the input contains more than one element. This problem is known as multiple revision and has been addressed for instance in [Hansson, 1992; Fuhrmann and Hansson, 1994] in a propositional setting.

Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a DL-Lite prioritized knowledge base. Let us start with the situation where the input, simply denoted again by N , is a set of assertional facts. If $\langle \mathcal{T}, N \rangle$ is consistent, then our approach can be applied straightforwardly. The definition of prioritized removed set is exactly the same. Definitions 6.2 to 6.7 can be used as it is except that N is a set of assertional facts instead of a single one. The same holds for Lemma 6.1 as well as Propositions 6.1-6.2.

Example 6.8. Let us consider $\mathcal{T} = \{A \sqsubseteq \neg B\}$ and $\mathcal{A} = \mathcal{A}_1 \cup \mathcal{A}_2$ where $\mathcal{A}_1 = \{A(a), B(c)\}$ and $\mathcal{A}_2 = \{B(b)\}$.

Let $N = \{A(b), B(a)\}$ where $\langle \mathcal{T}, N \rangle$ is consistent. Now, $\langle \mathcal{T}, \mathcal{A} \cup N \rangle$ is inconsistent. There only exists one prioritized removed set: $X_1 = \{A(a), B(b)\}$ and $\mathcal{A} \circ_{PRSR} N = \{A(b), B(a), B(c)\}$. ■

Now assume that $\langle \mathcal{T}, N \rangle$ is inconsistent. In this case, if we still consider that \mathcal{T} as a stable knowledge, then the input cannot be completely accepted. In this case, the prioritized removed set will both contain elements from \mathcal{A} and also from N , with elements of N being preferred to all elements of \mathcal{A} . Definition 6.2 needs the following adaptation :

Definition 6.8. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a consistent stratified knowledge base and N be a set of membership assertions. A *prioritized removed set*, denoted by X , is a set of membership assertions such that:

- $X \subseteq \mathcal{A} \cup N$,
- $\langle \mathcal{T}, (\mathcal{A} \cup N) \setminus X \rangle$ is consistent,
- $\forall X' \subseteq \mathcal{A} \cup N$, if $\langle \mathcal{T}, (\mathcal{A} \cup N) \setminus X' \rangle$ is consistent then $X \leq_{lex} X'$.

Note that $\mathcal{A} \cup N$ is a new prioritized ABox, where elements of N are put in a new important stratum. Namely, let $\mathcal{A} = \mathcal{A}_1 \cup \dots \cup \mathcal{A}_n$ be a prioritized ABox. Then $\mathcal{A} \cup N = \mathcal{A}'_1 \cup \dots \cup \mathcal{A}'_{n+1}$ where $\mathcal{A}'_1 = N$, and $\mathcal{A}'_i = \mathcal{A}_{i-1}$ for $i = 1, \dots, n + 1$.

The remaining definitions are valid, however Lemma 3.6 and Proposition 6.1 do not hold as it is shown the following counter-example.

Example 6.9. Let us consider $\mathcal{T} = \{A \sqsubseteq \neg B\}$ and $\mathcal{A} = \mathcal{A}_1 \cup \mathcal{A}_2$ where $\mathcal{A}_1 = \{A(a)\}$ and $\mathcal{A}_2 = \{B(b)\}$. Let $N = \{A(b), B(b), B(a)\}$ where $\langle \mathcal{T}, N \rangle$ is inconsistent. We have $\langle \mathcal{T}, N \cup \mathcal{A} \rangle$ is also inconsistent. The conflict sets are: $C_1 = \{A(a), B(a)\}$ and $C_2 = \{A(b), B(b)\}$. The two prioritized removed sets are: $X_1 = \{A(a), A(b)\}$ and $X_2 = \{A(a), B(b)\}$.

One can check that there exist more than one prioritized removed set which both contain elements from \mathcal{A} and N . ■

When the input N is a set of PI axioms or NI axioms. We assume that $\mathcal{T} \cup N$ is coherent, since the TBox of the knowledge base is assumed to be stable. Of course $\langle \mathcal{T} \cup N, \mathcal{A} \rangle$ may be inconsistent. In this case PRSR behaves in the same way as simple revision by a single input. In both cases (set of assertions or axioms), the most noticeable difference is that the number of conflicts may be higher and by consequence the size of prioritized removed sets may be higher.

Lastly, if the input contains both membership assertions and PI axioms or NI axioms, then this comes down to revise the *DL-Lite* prioritized knowledge base $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ with another knowledge base $\langle \mathcal{T}', \mathcal{A}' \rangle$. One way to achieve such revision is to apply PRSR on $\langle \mathcal{T} \cup \mathcal{T}', \mathcal{A} \cup N \rangle$.

Example 6.10. Let us consider $\mathcal{T} = \{A \sqsubseteq \neg B\}$ and $\mathcal{A} = \mathcal{A}_1 \cup \mathcal{A}_2$ where $\mathcal{A}_1 = \{A(a)\}$ and $\mathcal{A}_2 = \{B(b)\}$. Let $N = \{C \sqsubseteq A, C(a), A(b)\}$. We have $\langle \mathcal{T} \cup \mathcal{T}', \mathcal{A} \cup N \rangle$ is inconsistent. There exists only one conflict set $C_1 = \{A(b), B(b)\}$, and then one prioritized removed set $X_1 = \{B(b)\}$. ■

6.3 Logical properties

In this section we go a step further in the characterization of Prioritized Removed Sets Revision for *DL-Lite* knowledge bases by presenting logical properties of the proposed operator through a set of postulates.

As mentioned in the Introduction, the AGM postulates [Alchourrón *et al.*, 1985] have been formulated to characterize belief revision in a propositional logic setting. Flouris *et al.* [Flouris *et al.*, 2004; Flouris *et al.*, 2005; Flouris *et al.*, 2006b] have studied which logics are AGM-compliant, that is, DLs where the revision operation satisfies AGM postulates. Indeed, the problem is that AGM postulates are defined for belief sets, i.e. deductively closed sets of formulas, possibly infinite. Qi *et al.* [Qi *et al.*, 2006c] focused on revising a finite representation of belief sets. They used a semantic reformulation of AGM postulates, done by Katsuno and Mendelzon [Katsuno and Mendelzon, 1991], to extend it to DLs knowledge bases. However, as pointed out in [Calvanese *et al.*, 2010] known model-based approaches of revision are not expressible in *DL-Lite*. AGM postulates are defined for belief sets, however efficient implementation and computational tractability require finite representations. Moreover, cognitive realism stems from finite structures [Hansson, 2008] since infinite structures are cognitively inaccessible. Revision within the framework of DLs, particularly, *DL-Lite*, requires belief bases, i.e. finite sets of formulas. Postulates have been proposed for characterizing belief bases revision in a propositional logic setting [Fuhrmann, 1997; Hansson, 1998].

In order to give logical properties of PRSR operators, we first rephrase Hansson’s postulates within the *DL-Lite* framework. We then analyze to what extent our operators satisfy these postulates.

6.3.1 Hansson’s postulates reformulated

Let $\mathcal{K}, \mathcal{K}'$ be *DL-Lite* knowledge bases, N and M be either membership assertions or a positive or a negative axiom, \circ be a revision operator. $\mathcal{K} + N$ denotes the non closing expansion, i.e. $\mathcal{K} + N = \mathcal{K} \cup \{N\}$. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a knowledge base. When N is a membership assertion $\mathcal{K} + N = \langle \mathcal{T}, \mathcal{A} \cup \{N\} \rangle$ and when N is a positive or a negative axiom $\mathcal{K} + N = \langle \mathcal{T} \cup \{N\}, \mathcal{A} \rangle$. We rephrase the Hansson’s postulates as follows.

<i>Success</i>	$N \in \mathcal{K} \circ N$
<i>Inclusion</i>	$\mathcal{K} \circ N \subseteq \mathcal{K} + N$.
<i>Consistency</i>	$\mathcal{K} \circ N$ is consistent.
<i>Vacuity</i>	If $\mathcal{K} \cup \{N\}$ is consistent then $\mathcal{K} \circ N = \mathcal{K} + N$.
<i>Pre-expansion</i>	$(\mathcal{K} + N) \circ N = \mathcal{K} \circ N$.
<i>Internal exchange</i>	If $N, M \in \mathcal{K}$ then $\mathcal{K} \circ N = \mathcal{K} \circ M$.
<i>Core retainment</i>	If $M \in \mathcal{K}$ and $M \notin \mathcal{K} \circ N$ then there exists \mathcal{K}' such that $\mathcal{K}' \subseteq \mathcal{K} + N$ and \mathcal{K}' is consistent but $\mathcal{K}' \cup \{M\}$ is inconsistent.
<i>Relevance</i>	If $M \in \mathcal{K}$ and $M \notin \mathcal{K} \circ N$ then there exists \mathcal{K}' such that $\mathcal{K} \circ N \subseteq \mathcal{K}' \subseteq \mathcal{K} + N$, and \mathcal{K}' is consistent but $\mathcal{K}' \cup \{M\}$ is inconsistent.

Success and *Consistency* express the basic principles of revision. *Inclusion* states that the union of the initial knowledge bases is the upper bound of any revision operation. *Vacuity* says that if the new information is consistent with the knowledge base then the result of revision equals the non closing expansion. *Pre-expansion* states that expanding first by an assertion does not change the result of revision by the same assertion. *Internal exchange* says that revising by two different assertions from the knowledge base does not change the result of revision. *Core-retainment* and *Relevance* express the intuition that nothing is removed from the original knowledge bases unless its removal in some way contributes to make the result consistent.

6.3.2 Prioritized Removed Sets Revision: logical properties

We now present the logical properties of Prioritized Removed Sets operators.

Proposition 6.3. *Let \mathcal{K} be a consistent stratified DL-Lite knowledge base and N be a membership assertion. The revision operator \circ_{PRSR} satisfies Success, Inclusion, Consistency, Vacuity, Pre-expansion, Internal exchange, Core retainment and Relevance.*

Proof of proposition 6.3. Since N is a membership assertion, $\mathcal{K} \cup \{N\} = \langle \mathcal{T}, \mathcal{A} \cup \{N\} \rangle$. By Definition 6.3, $\mathcal{K} \circ_{PRSR} N = \langle \mathcal{T}, \mathcal{A} \circ_{PRSR} N \rangle$ with $\mathcal{A} \circ_{PRSR} N = (\mathcal{A} \setminus X) \cup \{N\}$ and the postulates *Success*, *Inclusion*, *Consistency* are satisfied.

Vacuity: If $\mathcal{K} \cup \{N\}$ is consistent, then $\mathcal{PR}(\mathcal{K} \cup \{N\}) = \emptyset$ and $\mathcal{A} \circ_{PRSR} N = \mathcal{A} \cup \{N\}$, therefore the postulate holds.

Pre-expansion: $(\mathcal{A} \cup \{N\}) \circ_{PRSR} N = ((\mathcal{A} \cup \{N\}) \setminus X) \cup \{N\} = (\mathcal{A} \setminus X) \cup \{N\}$, therefore the postulate is satisfied.

Internal exchange: If $N, M \in \mathcal{A}$, $\mathcal{A} \cup \{M\} = \mathcal{A} \cup \{N\} = \mathcal{A}$ and $\mathcal{PR}(\mathcal{K} \cup \{N\}) = \mathcal{PR}(\mathcal{K} \cup \{M\}) = \emptyset$, therefore the postulate is satisfied.

Core retainment: The case where $M \in \mathcal{T}$ is impossible since the \circ_{PRSR} operator may only modify the ABox. When M is a membership assertion, if $M \in \mathcal{K}$ and $M \notin \mathcal{K} \circ_{PRSR} N$ then there exists X such that $M \in X$ and $X \in \mathcal{PR}(\mathcal{K} \cup \{N\})$. Let $\mathcal{K}' = \langle \mathcal{T}, \mathcal{A} \setminus X \rangle$, we have $\mathcal{K}' \subseteq \mathcal{K} \cup \{N\}$ et \mathcal{K}' is consistent but $\mathcal{K}' \cup \{M\}$ is inconsistent, therefore the postulate is satisfied.

Relevance: Since the postulate *Core retainment* is satisfied, and by Proposition 6.1 we have $|\mathcal{PR}(\mathcal{K} \cup \{N\})| = 1$, so $\mathcal{K} \circ_{PRSR} N \subseteq \mathcal{K}'$ and thus the postulate holds. \square

This proposition states that PRSR with a membership assertion as input satisfies all postulates. The situation is slightly different when N is a PI or a NI axiom.

Proposition 6.4. *Let \mathcal{K} be a consistent stratified DL-Lite knowledge base. If N is a PI or a NI axiom then for any selection function, the revision operator \circ_{PRSR} satisfies Success, Inclusion, Consistency, Vacuity, Pre-expansion, Internal exchange, Core retainment but does not satisfy Relevance.*

Proof of proposition 6.4. Since N is a positive or a negative axiom, $\mathcal{K} \cup \{N\} = \langle \mathcal{T} \cup \{N\}, \mathcal{A} \rangle$. By Definition 6.6, $\mathcal{K} \circ_{PRSR} N = \langle \mathcal{T} \cup \{N\}, \mathcal{A} \circ_{PRSR} N \rangle$ with $\mathcal{A} \circ_{PRSR} N = (\mathcal{A} \setminus f(\mathcal{R}(\mathcal{K} \cup \{N\})))$ and the postulates *Success*, *Inclusion*, *Consistency* are satisfied.

Vacuity: If $\mathcal{K} \cup \{N\}$ is consistent, $\mathcal{PR}(\mathcal{K} \cup \{N\}) = \emptyset$ and $\mathcal{A} \circ_{PRSR} N = \mathcal{A}$, therefore the postulate holds.

Pre-expansion: $(\mathcal{K} \cup \{N\}) \circ_{PRSR} N, (\langle \mathcal{T} \cup \{N\}, \mathcal{A} \rangle) \circ_{PRSR} N = \langle \mathcal{T} \cup \{N\}, \mathcal{A} \circ_{PRSR} N \rangle$, therefore the postulate is satisfied.

Internal exchange: If $N, M \in \mathcal{T}$, $\mathcal{T} \cup \{M\} = \mathcal{T} \cup \{N\} = \mathcal{T}$ and $\mathcal{PR}(\mathcal{K} \cup \{N\}) = \mathcal{PR}(\mathcal{K} \cup \{M\}) = \emptyset$, therefore the postulate is satisfied.

Core retainment: The case where $M \in \mathcal{T}$ is impossible since the \circ_{PRSR} operator may only modify the ABox. When M is a membership assertion, if $M \in \mathcal{K}$ and $M \notin \mathcal{K} \circ_{PRSR} N$, then for any selection function used for defining \circ_{PRSR} , there exists $X \in \mathcal{PR}(\mathcal{K} \cup \{N\})$ such that $M \in X$ and $X \subseteq f(\mathcal{R}(\mathcal{K} \cup \{N\}))$ by Definition 6.5. Let $\mathcal{K}' = \langle \mathcal{T} \cup \{N\}, \mathcal{A} \setminus X \rangle$, we have $\mathcal{K}' \subseteq \mathcal{K} \cup \{N\}$ and \mathcal{K}' is consistent but $\mathcal{K}' \cup \{M\}$ is inconsistent, therefore the postulate is satisfied.

Relevance: Since the postulate *Core retainment* is satisfied, there exists $\mathcal{K}' = \langle \mathcal{T} \cup \{N\}, \mathcal{A} \setminus X \rangle$ that is consistent. Since there may exist several prioritized removed sets, let X and X' be two prioritized removed sets such that $X' \neq X$, suppose that $f(\mathcal{PR}(\mathcal{K} \cup \{N\})) = X'$, we have $\mathcal{K} \circ_{PRSR} N = \langle \mathcal{T} \cup \{N\}, \mathcal{A} \setminus X' \rangle$ therefore $\mathcal{K} \circ_{PRSR} N \not\subseteq \mathcal{K}'$, therefore the postulate *Relevance* is not satisfied. We now give a counter-example where \mathcal{K} and N come from Example 6.4. Let $M = B_3(b)$, $X = \{B_3(a), B_3(b)\}$ and $X' = \{B_1(a), B_3(b)\}$ be two prioritized removed sets, suppose that $f(\mathcal{PR}(\mathcal{K} \cup \{N\})) = X'$ we have $\mathcal{A} \circ_{PRSR} N = \{B_3(a), B_2(b)\}$ and $\mathcal{A} \setminus X = \{B_1(a), B_2(b)\}$. \square

In fact, *Relevance* requires the existence of only one prioritized removed set which is the case where N is a membership assertion. However, when N is a PI or a NI axiom, in general, there may exist several prioritized removed sets.

6.4 Computing the revision operation outcome

As stated before, when trying to revise a *DL-Lite* knowledge base by a membership assertion, a PI axiom or a NI axiom, we want to withdraw only ABox assertions in order to restore consistency, *i.e.* prioritized removed sets will only contain elements from the ABox.

From the computational point of view, we have to distinguish several cases depending on the nature of the input N and the content of the knowledge base.

First of all, if the TBox \mathcal{T} only contains PI axioms, and if the input N is a PI axiom or a membership assertion, no inconsistency can occur, so the revision operation PRSR trivially becomes a simple union. Among the remaining cases, we distinguish two different situations:

- (i) N is a membership assertion: the computation of conflicts and the overall revision algorithm is a very simple task, thanks to Proposition 6.1, and is detailed below.
- (ii) N is a PI axiom or a NI axiom : this is the most complicated case, as several prioritized removed sets may exist. Moreover, we will see that this case has to be splitted into two subcases. Whatever case we consider, we first need to compute the conflicts of $\mathcal{K} \cup \{N\}$.

In what follows, we use the following notations: $\mathcal{K}' = \langle \mathcal{T}', \mathcal{A}' \rangle = \mathcal{K} \cup \{N\}$ where:

- $\mathcal{T}' = \mathcal{T} \cup \{N\}$ and $\mathcal{A}' = \mathcal{A}$, if N is a PI or NI axiom, and
- $\mathcal{T}' = \mathcal{T}$ and $\mathcal{A}' = \mathcal{A} \cup \{N\}$, if N is an ABox assertion.

6.4.1 Computing the conflicts

This step follows from the algorithm given in [Calvanese *et al.*, 2007a] for checking the consistency of a DL-Lite knowledge base. The main difference is that in [Calvanese *et al.*, 2007a] the aim is to check whether a DL-Lite knowledge base is consistent or not. Here, we have to perform one step further, as we need to enumerate all assertional pairs involved in conflicts. Hence, we need to adapt the algorithm.

Computing $\mathcal{C}(\mathcal{K} \cup \{N\})$ first requires to obtain the negative closure $cln(\mathcal{T}')$, using the rules recalled in the refresher on DL-Lite logic in Section 1.4. We suppose that this is performed by a NEG-CLOSURE function. Then the computation of the conflicts proceeds with the evaluation over \mathcal{A}' of each NI axiom in $cln(\mathcal{T}')$ in order to exhibit whether \mathcal{A}' contains pairs of assertions that contradict the NI axioms. Intuitively, for each $X \sqsubseteq \neg Y$ belonging to $cln(\mathcal{T}')$, the evaluation of $X \sqsubseteq \neg Y$ over the \mathcal{A}' simply amounts to return all $(X(x), Y(x))$ such that $X(x)$ and $Y(x)$ belong to \mathcal{A}' . Note $X(x)$ (*resp.* $Y(x)$) may be a basic concept assertion, or a role assertion of the form $R(x, y)$ if $X = \exists R$ (*resp.* $Y = \exists R$) or $R(y, x)$ if $X = \exists R^-$ (*resp.* $Y = \exists R^-$). The result of the evaluation of a NI axiom is a collection of sets containing two elements, or one element if N is a membership assertion). Algorithm 3 describes the algorithm of the function COMPUTECONFLICTS, which computes $\mathcal{C}(\mathcal{K} \cup \{N\})$.

```

1: function COMPUTECONFLICTS( $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle, N$ )
2:    $\mathcal{K}' = \langle \mathcal{T}', \mathcal{A}' \rangle \leftarrow \mathcal{K} \cup \{N\}$ 
3:    $\mathcal{C}(\mathcal{K}') \leftarrow \emptyset$ 
4:    $cln(\mathcal{T}') \leftarrow \text{NEG-CLOSURE}(\mathcal{T}')$ 
5:   for all  $X \sqsubseteq \neg Y \in cln(\mathcal{T}')$  do
6:     for all  $\{\alpha_t, \alpha_j\} \in \mathcal{A}'$  do
7:       if  $\langle X \sqsubseteq \neg Y, \{\alpha_t, \alpha_j\} \rangle$  is inconsistent then
8:          $\mathcal{C}(\mathcal{K}') \leftarrow \mathcal{C}(\mathcal{K}') \cup \{\{\alpha_t, \alpha_j\}\}$ 
9:   Return  $\mathcal{C}(\mathcal{K}')$ 
    
```

Algorithm 3: COMPUTECONFLICTS(\mathcal{K})

The set $\mathcal{C}(\mathcal{K}')$ stores the conflict sets. The first step of the algorithm consists in the computation of the negative closure of \mathcal{T}' . Then, for each NI axiom $X \sqsubseteq \neg Y$ of $cln(\mathcal{T}')$ the algorithm looks for the existence of a contradiction in the ABox. This is done by checking whether $\langle X \sqsubseteq \neg Y, \{\alpha_t, \alpha_j\} \rangle$ is consistent or not. Note that this step can be performed by a boolean query expressed from $X \sqsubseteq \neg Y$ to

look whether $\{\alpha_t, \alpha_j\}$ contradicts the query, or not. If the ABox is consistent with $X \sqsubseteq \neg Y$, then the result of the query is an empty set.

It is important to note that if N is a membership assertion, then in each conflict $\{\alpha_t, \alpha_j\}$ either α_t or α_j belongs to \mathcal{A} (but not both), and that either α_t or α_j is equal to N (but not both). This special case is detailed in the next subsection.

6.4.2 Computing the PRSR outcome

Based on the computation of conflict sets, we propose in what follows algorithms for computing the results of revision according to the different cases.

Revision by an assertion

When the input N is a membership assertion (namely a fact), then there exists only one prioritized removed set, and the priorities are not involved. The computation of this single prioritized removed set amounts in picking in each conflict the membership assertion which is different from the new information N . One can easily check that every conflict set $\{\alpha_t, \alpha_j\}$ that contradicts a NI axiom is of the form $\{x, N\}$ where $x \in \mathcal{A}$. This means that there exists exactly one prioritized removed set. Hence, in this case the prioritized removed set computation can be performed in polynomial time: when returning from the call to COMPUTECONFLICTS, the only prioritized removed set is $\bigcup_{c_i \in \mathcal{C}(\mathcal{K} \cup \{N\})} (c_i \setminus \{N\})$.

Algorithm 4 describes the algorithm of the function COMPUTEPRSR1 as a special case of Algorithm 3. It computes directly the single prioritized removed set when revising by a membership assertion.

```

1: function COMPUTEPRSR1( $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ ,  $N = A(a)$  or  $N = R(a, b)$ )
2:    $R \leftarrow \emptyset$ 
3:    $cln(\mathcal{T}) \leftarrow \text{NEG CLOSURE}(\mathcal{T})$ 
4:   for all  $X \sqsubseteq \neg Y \in cln(\mathcal{T})$  do
5:     for all  $\alpha \in \mathcal{A}$  do
6:       if  $\langle X \sqsubseteq \neg Y, \{\alpha, N\} \rangle$  is inconsistent then
7:          $R \leftarrow R \cup \{\alpha\}$ 
8:   return  $R$ 

```

Algorithm 4: COMPUTEPRSR1(\mathcal{K}, N)

Revision by an axiom

Now, we detail the case where N is a PI or a NI axiom. According to Definition 6.4, the computation of $\mathcal{PR}(\mathcal{K} \cup \{N\})$ starts with the computation of $\mathcal{PR}((\mathcal{T} \cup \{N\}) \cup \mathcal{A}_1)$, followed by the computation of $\mathcal{PR}((\mathcal{T} \cup \{N\}) \cup (\mathcal{A}_1 \cup \mathcal{A}_2))$, and so on. A prioritized removed set is formed by picking in each conflict the least priority element. However, according to the form of conflicts, two situations hold, as pointed out in Section 6.2.2. The first one is when *each conflict* involves two elements having different levels of priority. In this case, Proposition 6.2 ensures that there exists only one prioritized removed set. We provide algorithm COMPUTEPRSR2 which computes this single prioritized removed set $PR \in \mathcal{PR}(\mathcal{K} \cup \{N\})$.

The algorithm COMPUTEPRSR2 proceeds from a current layer to all the other less preferred layers and selects the assertions that conflict with the ones in the current layer. Here we increment from a layer to another in order to ensure the minimality of the prioritized removed set with respect to lexicographic ordering. Note that this algorithm is based on inconsistency checking and its computational complexity is polynomial.

```

1: function COMPUTEPRSR2 ( $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle, N$ )
2:    $Res \leftarrow \mathcal{A}_1$ 
3:    $PR \leftarrow \emptyset$ 
4:    $\mathcal{C} \leftarrow \text{COMPUTECONFLICTS}(\mathcal{K}, N)$ 
5:   for  $i \leftarrow 2, n$  do
6:     for all  $\alpha \in \mathcal{A}_i$  do
7:       if  $\exists C \in \mathcal{C}, Res \cap C \neq \emptyset$  and  $\alpha \in C$  then
8:          $PR \leftarrow PR \cup \{\alpha\}$ 
9:          $\mathcal{A}_i \leftarrow \mathcal{A}_i \setminus \{\alpha\}$ 
10:     $Res \leftarrow Res \cup \mathcal{A}_i$ 
11:  Return  $PR$ 
    
```

Algorithm 5: COMPUTEPRSR2

Now, we detail the second case where there exists at least a conflict involving two elements having the same priority level. In such situation there exists several prioritized removed sets, as pointed out in section 6.2.2. To compute them, we use the hitting set notion [Reiter, 1987] and adapt it to the stratified structure of the knowledge base.

A hitting set is a set which intersects each set in a collection. A minimal hitting set, with respect to set inclusion, is called a kernel. Moreover, kernels which are minimal according to cardinality correspond to the definition of a removed set [Würbel *et al.*, 2000]. The same result holds for *DL-Lite* knowledge bases where the computation of the kernels of $\mathcal{C}(\mathcal{K} \cup \{N\})$ is performed using Reiter’s algorithm [Reiter, 1987], modified in [Wilkerson *et al.*, 1989]. We recall this algorithm.

Definition 6.9. A tree T is an HS-tree of $\mathcal{C}(\mathcal{K} \cup \{N\})$ if and only if it is the smallest tree having the following properties:

1. Its root is labeled by an element from $\mathcal{C}(\mathcal{K} \cup \{N\})$. If $\mathcal{C}(\mathcal{K} \cup \{N\})$ is empty, its root is labeled by ‘ \surd ’.
2. If m is a node from T , let $H(m)$ be the set of branch labels on the path going from the root to T to m . If m is labeled by ‘ \surd ’, it has no successor in T .
3. If m is labeled by a set $C \in \mathcal{C}(\mathcal{K} \cup \{N\})$, then, for each $c \in C$, m has a successor node m_c in T , joined to m by a branch labeled by c . The label of m_c is a set $C' \in \mathcal{C}(\mathcal{K} \cup \{N\})$ such that $C' \cap H(m_c) = \emptyset$, if such a set exists. Otherwise, m_c is labeled by ‘ \surd ’.

The kernels correspond to the leaves labeled by \surd . For each such node m , $H(m)$ is a kernel of $\mathcal{C}(\mathcal{K} \cup \{N\})$. We use the same pruning techniques as in [Wilkerson *et al.*, 1989].

Prioritized removed sets are not necessarily minimal with respect to cardinality, but they are minimal with respect to lexicographic ordering (\leq_{lex} for short). So, a naive algorithm for computing $\mathcal{PR}(\mathcal{K} \cup \{N\})$ could be : (i) compute the kernels of $\mathcal{C}(\mathcal{K} \cup \{N\})$. (ii) keep only minimal ones with respect to \leq_{lex} . However, we can improve this algorithm. As we said before, a prioritized removed set is computed from a layer to another. The idea of the enhancement of the algorithm is as follows: Compute conflicts in the first layer, *i.e.* in $\langle \mathcal{T} \cup \{N\}, \mathcal{A}_1 \rangle$. Then, build the hitting set tree on this collection of conflicts. This tree allows for the computation of the kernels of $\langle \mathcal{T} \cup \{N\}, \mathcal{A}_1 \rangle$, which are minimal with respect to \leq_{lex} . From these kernels, we continue the construction of the tree using conflicts in $\langle \mathcal{T} \cup \{N\}, \{\mathcal{A}_1 \cup \mathcal{A}_2\} \rangle$ if they exist, and so on until reaching a fixed point where no conflict will be generated. Then, the kernels of the final hitting set tree — *i.e.* those built using the conflicts in $\langle \mathcal{T} \cup \{N\}, \{\mathcal{A}_1 \cup \mathcal{A}_2 \cup \dots \cup \mathcal{A}_n\} \rangle$

```

1: function COMPUTEPRSR3 ( $\mathcal{K}=\langle \mathcal{T}, \mathcal{A} \rangle, N$ )
2:    $\mathcal{T}' \leftarrow \mathcal{T} \cup \{N\}, \mathcal{K}' = \langle \mathcal{T}', \mathcal{A} \rangle$ 
3:    $cln(\mathcal{T}') \leftarrow \text{NEGCLOSURE}(\mathcal{T}')$ 
4:    $\mathcal{PR}(\mathcal{K}') \leftarrow \emptyset, \mathcal{C} \leftarrow \emptyset, \text{TREE} \leftarrow \emptyset, i \leftarrow 1$ 
5:   while  $i \leq n$  do
6:     for all  $X \sqsubseteq \neg Y \in cln(\mathcal{T}')$  do
7:       for all  $(\alpha, \beta)$  s.t.  $\alpha \in \mathcal{A}_1, \beta \in \mathcal{A}_1 \cup \dots \cup \mathcal{A}_i$  do
8:         if  $\langle X \sqsubseteq \neg Y, \{\alpha, \beta\} \rangle$  is inconsistent then
9:            $\mathcal{C} \leftarrow \mathcal{C} \cup \{\alpha, \beta\}$ 
10:         $\text{TREE} \leftarrow \text{TREE.ADDFROMLEXKERNEL}(\text{HS}(\mathcal{C}))$ 
11:         $\mathcal{C} \leftarrow \emptyset,$ 
12:         $i \leftarrow i + 1$ 
13:    $\mathcal{PR}(\mathcal{K}') \leftarrow \text{LEXKERNEL}(\text{TREE})$ 
14:   return  $\mathcal{PR}(\mathcal{K}')$ 

```

Algorithm 6: COMPUTEPRSR3

— which are minimal with respect to \leq_{lex} are the prioritized removed sets. Algorithm 6 describes the algorithm of the function COMPUTEPRSR3, which computes $\mathcal{PR}(\mathcal{K} \cup \{N\})$.

In this algorithm, the function HS(\mathcal{C}) takes as input the conflicts computed in each strata (if they exist) and builds the corresponding hitting sets tree (TREE) using the algorithm presented in [Reiter, 1987; Wilkerson *et al.*, 1989]. From one layer to another, we resume the construction of (TREE) from its current kernels minimal with respect to \leq_{lex} . Namely, the function ADDFROMLEXKERNEL((HS(\mathcal{C}))) builds the hitting set tree out of a collection of conflicts \mathcal{C} , starting from the branches of the current TREE which are minimal with respect to \leq_{lex} . Finally, $\mathcal{PR}(\mathcal{K} \cup \{N\})$ corresponds to the kernels of TREE obtained using function LEXKERNEL(TREE) which are minimal with respect to \leq_{lex} . Note that COMPUTEPRSR3 is a generalization of COMPUTEPRSR2, since when all conflicts involve elements from distinct layers, then the final tree will only contain one prioritized removed set. The following example illustrates this algorithm.

Example 6.11. Consider $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$, with $\mathcal{T} = \{A \sqsubseteq B, C \sqsubseteq B\}$ and $\mathcal{A} = \mathcal{A}_1 \cup \mathcal{A}_2 \cup \mathcal{A}_3 \cup \mathcal{A}_4$ where $\mathcal{A}_1 = \{A(a), D(a)\}, \mathcal{A}_2 = \{C(a), B(b)\}, \mathcal{A}_3 = \{D(b)\}$ and $\mathcal{A}_4 = \{D(c), C(c)\}$. We want to revise \mathcal{K} with $N = B \sqsubseteq \neg D$. Then, we have $cln(\mathcal{T} \cup \{B \sqsubseteq \neg D\}) = \{B \sqsubseteq \neg D, A \sqsubseteq \neg D, C \sqsubseteq \neg D\}$.

- The set of conflicts obtained from $\langle cln(\mathcal{T}'), \mathcal{A}_1 \rangle$ is $\{\{A(a), D(a)\}\}$. The HS tree built by calling HS($\{\{A(a), D(a)\}\}$) will contain two branches labeled respectively by $A(a)$ and $D(a)$ which are kernels minimal with respect to \leq_{lex} (\leq_{lex} -kernel).
- We go on with $\langle cln(\mathcal{T}'), \mathcal{A}_1 \cup \mathcal{A}_2 \rangle$ where $\{C(a), D(a)\}$ is a newly identified conflict. We resume the construction of the tree from its current \leq_{lex} -kernel branches labeled by $A(a)$ and $D(a)$, and we obtain three HS-tree branches: $\{A(a), C(a)\}, \{A(a), D(a)\}$ and $D(a)$, where only $D(a)$ is a \leq_{lex} -kernel.
- Now, we go to the next strata, that is, we use $\langle cln(\mathcal{T}'), \mathcal{A}_1 \cup \mathcal{A}_2 \cup \mathcal{A}_3 \rangle$. This produces a new conflict $\{B(b), D(b)\}$ and we continue the construction of the Tree from $D(a)$. We potentially obtain $\{D(a), D(b)\}$ and $\{D(a), B(b)\}$ as new prioritized removed sets, but only $\{D(a), D(b)\}$ is a \leq_{lex} kernel.
- Finally, we identify a new conflict $\{D(c), C(c)\}$ from $\langle cln(\mathcal{T}'), \mathcal{A}_1 \cup \mathcal{A}_2 \cup \mathcal{A}_3 \cup \mathcal{A}_4 \rangle$. We continue the construction of the tree from the branch labeled by $\{D(a), D(b)\}$. We obtain two other

branches labeled respectively by $\{D(a), D(b), C(c)\}$ and $\{D(a), D(b), D(c)\}$ which are two \leq_{lex} kernels. Hence, $\mathcal{PR}(\mathcal{K} \cup \{N\}) = \{\{D(a), D(b), C(c)\}, \{D(a), D(b), D(c)\}\}$.

■

6.5 Discussions and related works

In [Calvanese *et al.*, 2010], the authors study the problem of knowledge base *evolution* in *DL-Lite*. Under the word evolution, they encompass both revision and update operations. Note that the update focuses on the changes of the actual state whereas revision focuses on the integration of new information [Wang *et al.*, 2010]. In this chapter, we focus on revision. The part of the article [Calvanese *et al.*, 2010] dedicated to formula-based approaches is closely related to our work. They define several operators which perform revision of a knowledge base expressed in *DL-Lite* at a syntactical level. The first difference concerns the form of the input. In our case, new information is a membership assertion, a single PI axiom or NI axiom, that is a single formula. In [Calvanese *et al.*, 2010], the input is a set of formulas. The second difference is that in [Calvanese *et al.*, 2010] they develop two operators whose strategy is to non-deterministically choose some maximal consistent subset. The first one, called *BoldEvol*, starts with the input, and incrementally and non-deterministically adds as many formulas as possible from the closure of the knowledge base. The algorithm for computing such set is polynomial. However, in the case where the input is a set of membership assertions, they give a result similar to our operator. Namely, the result only gives one maximal consistent subset, which corresponds to Proposition 6.1.

The selected maximal subset is a subset of the *consequences* of the knowledge base, which is very different from our point of view. Prioritized Removed Sets Revision relies only on the explicit content of the knowledge base. The resulting knowledge base will not contain formulas which are not present in the original knowledge base. Only working with explicitly given information, we follow Hansson’s point of view [Hansson, 2008].

Following this line, extensions of belief bases revision to DLs have been proposed, however these approaches differ from ours in several aspects. Within the general framework of DLs, in [Qi *et al.*, 2008a] the authors extend kernel-based revision [Hansson, 1994] for revising flat terminologies. Our approach is very different since we deal with knowledge bases which are prioritized and expressed in a lightweight DLs. Furthermore, our revision operators do not modify the TBox but revise the prioritized ABox according to a lexicographical strategy.

In [Halaschek-wiener *et al.*, 2006], the authors focus on *SHOIN* DL, they extend kernel revision and semi-revision operators [Hansson, 1997] to *SHOIN* knowledge bases. Moreover, they propose an algorithm for revision stemming from the computation of kernels. This algorithm shares several common points with our algorithm for the computation of prioritized removed sets. What they call justification of the inconsistency is very similar to our notion of conflict. But in their case, the generation of conflicts has a higher computational cost than in our case, as they work with *SHOIN* logic. In order to lower this extra-complexity, they rely on an optimized version of the Pellet consistency checker which uses properties of the *SHOIN* logic, allowing them to define an incremental version of their consistency checking tableau algorithm.

In [Ribeiro and Wassermann, 2007], the authors propose another extension of kernel-based revision and semi-revision operators to DLs, namely external kernel revision and semi-revision with weak success. Again, their logical framework is richer than ours, since they consider *SHOIN* and *SHIF* logics in order to capture all the *OWL-DL* and *OWL-Lite* languages. Our revision operators can be viewed as restrictions of the operator they define under the name *kernel revision without negation*. The restrictions are : (i) our knowledge bases are prioritized and expressed in *DL-Lite*; (ii) the minimality of the result of the incision function is defined in terms of lexicographic criterion in our case.

Following another idea, the authors in [Qi *et al.*, 2006a] extend weakening-based revision to *ACC* knowledge bases. Instead of removing conflicting assertions, the proposed revision operators weaken terminological axioms or assertions by adding exceptions which drop individuals responsible of the conflicts. Furthermore, this weakening-based revision is generalized to stratified knowledge bases. Our revision operators differ from this approach since our prioritized knowledge bases are expressed in *DL-Lite*. Moreover, the spirit is different since PRSR removes conflicting assertions according to a lexicographical strategy.

6.6 Conclusion

In this chapter, we investigated the problem of revising prioritized *DL-Lite* knowledge bases where the ABox is stratified. We considered several forms of the incorporated information, more precisely, when the input is a membership assertion, a positive or negative inclusion axiom. According to the form of the input we proposed a family of operators, Prioritized Removed Sets (PRSR) operators, stemming from a lexicographical strategy for removing some assertions, namely the prioritized removed sets, in order to restore consistency.

When the input is a membership assertion, the revision process leads to a unique revised knowledge base. However, when the input is a positive or negative inclusion axiom, the revision process may lead to several possible revised knowledge bases. In this case, we defined selection functions in order to keep the result within the *DL-Lite* language and we gave some concrete PRSR operators with examples of selection functions. We studied the logical properties of PRSR operators through Hansson's postulates rephrased within the *DL-Lite* framework. From a computational point of view, we first proposed an algorithm for pinpointing inconsistencies, then according to the nature of the input, we proposed algorithms, some of them using the notion of hitting set, for computing the prioritized removed sets.

Based on the discussion about suitable techniques for inconsistency handling presented in Chapter 5 and the assumption that the ABox is prioritized, next chapter proposes several approaches to select *one* preferred assertional-based repair from inconsistent and prioritized *DL-Lite* knowledge bases.

HOW TO SELECT ONE PREFERRED ASSERTIONAL-BASED REPAIR FROM INCONSISTENT AND PRIORITIZED *DL-Lite* KNOWLEDGE BASES?

7.1 Introduction

In Chapter 5, we addressed the problem of inconsistency handling in flat *DL-Lite* knowledge bases. We showed, in particular, that dealing with inconsistency in an Ontology-Based Data Access setting comes down first to compute the assertional-based repairs, and then, perform inference. However, as pointed out in previous chapter (Chapter 6), assertions are often provided by several and potentially conflicting sources having different reliability levels. In the previous chapter, we were interested in analyzing the problem of adding a new piece of information (assertional fact, positive inclusion axiom and negative inclusion axiom) in *DL-Lite* knowledge bases when the assertional base is prioritized. We followed a revision strategy that consists in throwing out some assertional facts to restore consistency. This chapter goes one step further and studies the problem of inconsistency handling in prioritized *DL-Lite* knowledge bases. The main question addressed in this chapter is how to select one preferred assertional-based repair. Selecting only one repair is important since it allows efficient query answering once the preferred repair is computed.

In this context, several works (e.g. [Martinez et al., 2008; Staworko et al., 2012]) studied the notion of priority when querying inconsistent databases. In [Du et al., 2013], a maximal repair with respect to set inclusion was introduced in order to answer queries from an inconsistent *SHIQ* DL knowledge base. In the *DL-Lite* area, there is to the best of our knowledge only one work [Bienvenu et al., 2014] dealing with reasoning under inconsistency using the priorities of assertions within the OBDA setting.

In this chapter, we first review the main existing inconsistency-tolerant reasoning methods for prioritized knowledge bases. We provide consequence relations based either i) on the selection of one consistent prioritized assertional base which is in general not maximal or ii) on the choice of several prioritized repairs. It is important to note that some inference relations are specific to *DL-Lite* even if they are inspired by other formalisms, such as propositional logic settings.

This chapter also contains different main strategies for computing repairs that are suitable for the *DL-Lite* setting. As mentioned in Chapter 5, a suitable inconsistency-tolerant relation is a one that outputs a consistent *DL-Lite* knowledge base and produces as many safe conclusions as possible. In particular, it should be at least as productive as taking the intersection of all the repairs. Interestingly enough, many of such consequence relations allow an efficient handling of inconsistency in *DL-Lite* knowledge base.

The rest of this chapter is organized as follows: Section 7.2 reviews existing works for computing preferred repairs. Section 7.3 introduces the notion of prioritized deductive closure and studies the properties of existing approaches with respect to the notion of deductive closure. Section 7.4 investigates new inference strategies based on the selection of one preferred repair. Section 7.5 provides a compara-

tive analysis of the main approaches studied in this chapter. Section 7.6 gives experimental studies and Section 7.7 concludes the chapter.

Notations: Let us first summarize in Table 7.1 the different notations of repairs that will be used in the rest of this chapter.

Acronym	Definition	Signification
$MAR(\mathcal{A})$	Definition 5.3	(flat) inclusion-based repairs of \mathcal{A}
$MAR_{card}(\mathcal{A})$	$\circ_{card}(\circ_{incl}(\mathcal{A}))$ (Chapter 5)	(flat) cardinality-based repairs of \mathcal{A}
$PAR(\mathcal{A})$	Definition 7.1	preferred inclusion-based repair of \mathcal{A}
$\pi(\mathcal{A})$	Definition 7.5	possibilistic-based repair of \mathcal{A}
$\ell(\mathcal{A})$	Definition 7.7	linear-based repair of \mathcal{A}
$pr_{lex}(\mathcal{A})$	Definition 7.3	preferred lexicographic-based repair of \mathcal{A}
$nd(\mathcal{A})$	Definition 7.10	non-defeated repair of \mathcal{A}
$free(\mathcal{S})$	Definition 7.11	non-conflicting assertions of a subset \mathcal{S} w.r.t a TBox \mathcal{T}
$clnd(\mathcal{A})$	Equation 7.2	closed non-defeated repair of \mathcal{A}
$\ell nd(\mathcal{A})$	Algorithm 7	linear-based non-defeated repair of \mathcal{A}
$nd(\mathcal{A})_{card}$	Definition 7.13	cardinality-based non-defeated repair of \mathcal{A}
$consnd(\mathcal{A})_{card}$	Definition 7.14	consistent cardinality-based non-defeated repair of \mathcal{A}
$pin d(\mathcal{A})$	Definition 7.15	prioritized inclusion-based non-defeated repair of \mathcal{A}
$pin d_{lex}(\mathcal{A})$	Equation 7.5	prioritized lexicographical-based non-defeated repair of \mathcal{A}

Table 7.1: Notations of repairs using in this Chapter.

7.2 Existing assertional-based preferred repairs

This section reviews approaches dealing with inconsistent *DL-Lite* knowledge bases that either have been proposed in a DLs setting or have been proposed in a propositional logic setting but need a slight adaptation to be suitable for *DL-Lite*.

A *DL-Lite* knowledge base $\mathcal{K}=\langle \mathcal{T}, \mathcal{A} \rangle$ with a prioritized assertional base is a *DL-Lite* knowledge base where \mathcal{A} is partitioned into n layers (or strata) of the form

$$\mathcal{A} = \mathcal{S}_1 \cup \dots \cup \mathcal{S}_n$$

where each layer \mathcal{S}_i contains the set of assertions having the same level of priority i and they are considered as more reliable than the ones present in a layer \mathcal{S}_j when $j > i$. Within the OBDA setting, we assume that \mathcal{T} is stable and hence its elements are not questionable in the presence of conflicts. Throughout this chapter and when there is no ambiguity, we simply use "prioritized *DL-Lite* knowledge base $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ " to refer to a *DL-Lite* knowledge base with a prioritized assertional base of the form $\mathcal{A}=\mathcal{S}_1 \cup \dots \cup \mathcal{S}_n$.

Example 7.1. Let $\mathcal{K}=\langle\mathcal{T}, \mathcal{A}\rangle$ such that $\mathcal{T}=\{A \sqsubseteq \neg B\}$ and that assertional facts of \mathcal{A} come from three distinct sources $\mathcal{A}=\mathcal{S}_1 \cup \mathcal{S}_2 \cup \mathcal{S}_3$ where $\mathcal{S}_1=\{B(a), A(b)\}$, $\mathcal{S}_2=\{A(a)\}$ and $\mathcal{S}_3=\{B(c)\}$. \mathcal{S}_1 contains the most reliable assertions. \mathcal{S}_3 contains the least reliable assertions. ■

In Example 7.1, it is easy to check that the knowledge base is inconsistent. Coping with inconsistency can be done by first computing the set of repairs, then using them to perform inference. In order to compute the repairs, we use the notion of conflict sets presented in Chapter 5 (Definition 5.12).

7.2.1 Preferred inclusion-based repair

In the flat case¹, one of the main strategies for handling inconsistency comes down to computing the ABox repairs of an inconsistent *DL-Lite* knowledge base. A repair is a maximal subbase of the ABox, denoted by *MAR*, that is consistent with the TBox (Definition 5.3 in Section 5.2).

Example 7.2. Consider $\mathcal{T}=\{A \sqsubseteq \neg B\}$ and $\mathcal{A}=\{A(a), B(a), A(b)\}$. We have $\mathcal{C}(\mathcal{A})=\{A(a), B(a)\}$. The set of *MAR* is: $\mathcal{R}_1=\{A(a), A(b)\}$ and $\mathcal{R}_2=\{B(a), A(b)\}$. ■

According to the definition of *MAR*, adding any assertion f from $\mathcal{A} \setminus \mathcal{R}$ to \mathcal{R} entails the inconsistency of $\langle\mathcal{T}, \mathcal{R} \cup \{f\}\rangle$. Moreover, the maximality in *MAR* is used in the sense of set inclusion. We denote by $MAR(\mathcal{A})$ the set of *MAR* of \mathcal{A} with respect to \mathcal{T} . A query is said to be a universal consequence (*i.e.* $\langle\mathcal{M}1, \forall\rangle$ given in Chapter 5) if it can be derived from every *MAR*. The following definition extends the definition of *MAR* when the *DL-Lite* ABox is prioritized.

Definition 7.1. Let $\mathcal{K}=\langle\mathcal{T}, \mathcal{A}\rangle$ be a prioritized *DL-Lite* knowledge base. A preferred inclusion-based repair (*PAR*) $\mathcal{P}=\mathcal{P}_1 \cup \dots \cup \mathcal{P}_n$ of \mathcal{A} is such that there is no a *MAR* $\mathcal{P}'=\mathcal{P}'_1 \cup \dots \cup \mathcal{P}'_n$ of $\mathcal{S}_1 \cup \dots \cup \mathcal{S}_n$, and an integer i where:

- i) \mathcal{P}_i is strictly included in \mathcal{P}'_i , and
- ii) $\forall j=1..(i-1)$, \mathcal{P}_j is equal to \mathcal{P}'_j

This definition of *PAR* has been largely used in a propositional logic setting (*e.g.* [Brewka, 1989; Benferhat *et al.*, 1998a]) and has been recently used in a *DL-Lite* framework [Bienvenu *et al.*, 2014]. A *PAR* of \mathcal{A} is formed by first computing the *MAR* of \mathcal{S}_1 , then enlarging this *MAR* as much as possible by assertions of \mathcal{S}_2 while preserving consistency, and so on.

Example 7.3. Consider $\mathcal{T}=\{A \sqsubseteq \neg B\}$ and $\mathcal{A}=\mathcal{S}_1 \cup \mathcal{S}_2$ where $\mathcal{S}_1=\{A(a)\}$ and $\mathcal{S}_2=\{B(a), A(b)\}$. There is exactly one *PAR* which is: $\mathcal{P}_1=\{A(a), A(b)\}$. ■

Priorities reduce the number of *MAR* as one can see in Example 7.3 in comparison with Example 7.2. Indeed, within a prioritized setting, the notion of *PAR* operates as a selection function among possible *MAR*. Following the definition of ABox conflict (Definition 5.12), an important feature in restoring consistency in *DL-Lite*, when the ABox is layered, is that when there is no conflict in \mathcal{A} involving two assertions having the same priority level, there exists only one *PAR*.

Proposition 7.1. Let $\mathcal{K}=\langle\mathcal{T}, \mathcal{A}\rangle$ be a prioritized *DL-Lite*. Let $\mathcal{C}(\mathcal{A})$ be the set of conflicts in \mathcal{A} . Then if $\forall \mathcal{C}=(f, g) \in \mathcal{C}(\mathcal{A})$ we have $f \in \mathcal{S}_i$, $g \in \mathcal{S}_j$ and $i \neq j$ then there exists exactly one *PAR*.

Proof of Proposition 7.1. The proof is immediate. In every conflict in \mathcal{A} , we throw out only the assertion having the lowest priority level. Therefore there exists only one *PAR* of \mathcal{A} . □

¹By a flat knowledge base, we mean a base where all the assertions have the same priority.

When a conflict involves two assertions having the same priority level, restoring consistency leads to several PAR . From now on, $PAR(\mathcal{A})$ denotes the set of PAR of \mathcal{A} . The following definition introduces universal inference when \mathcal{A} is layered.

Definition 7.2. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized *DL-Lite* knowledge base. A query q is said to be a PAR -consequence of \mathcal{K} , denoted $\mathcal{K} \models_{PAR} q$, if and only if $\forall \mathcal{P} \in PAR(\mathcal{A}), \langle \mathcal{T}, \mathcal{P} \rangle \models^2 q$ where $PAR(\mathcal{A})$ denotes the set of PAR of \mathcal{A} .

Definition 7.2 states that a query q is a universal consequence if and only if it can be deduced from every preferred inclusion-based repair. Note that the PAR -entailment extends the definition of MAR -entailment (i.e. $\langle \mathcal{M}_1, \forall \rangle$ proposed in Section 5) when the ABox is prioritized. Besides, it is argued that priorities simplify the computation of PAR , but it remains a hard task since in the flat case the inference is coNP-complete (see Section 5.5.2 on complexity analysis). When a conflict involves two assertions having the same priority level, restoring consistency often leads to several PAR .

7.2.2 Lexicographic preferred-based repair

This subsection rewrites the cardinality-based or lexicographic inference or prioritized removed set repair, defined in Chapter 6, to the context of inconsistency handling. The lexicographic inference has been widely used in the propositional setting (e.g. [Benferhat *et al.*, 1998a]). In fact, one of the major problems of PAR -entailment is the large number of PAR that can be computed from an inconsistent *DL-Lite* knowledge base. In order to better choose a PAR , one can follow a lexicographic-based approach. We introduce a preferred lexicographic-based repair which is based on the cardinality criterion instead of the set inclusion criterion.

Definition 7.3. Let $PAR(\mathcal{A})$ be the set of PAR of \mathcal{A} . Then $\mathcal{L} = L_1 \cup \dots \cup L_n$ is said to be a lexicographical preferred-based repair, denoted by PAR_{lex} , if and only if:

- i) $\forall \mathcal{P} = P_1 \cup \dots \cup P_n \in PAR(\mathcal{A})$: there is no i such that $|P_i| > |L_i|$,
- ii) $\forall j < i, |P_j| = |L_j|$.

where $|X|$ is the cardinality of the set X .

Clearly, using a lexicographic-based approach comes down to select among the set of repairs in $PAR(\mathcal{A})$ the ones having the maximal number of elements.

Definition 7.4. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized *DL-Lite* knowledge base. A query q is said to be *Lex*-consequence of \mathcal{K} , denoted by $\mathcal{K} \models_{\mathcal{L}} q$, if and only if $\forall \mathcal{L} \in PAR_{lex}(\mathcal{A}) : \langle \mathcal{T}, \mathcal{L} \rangle \models q$.

Example 7.4. Consider $\mathcal{T} = A \sqsubseteq \neg B, B \sqsubseteq \neg C$ and $\mathcal{A} = \mathcal{S}_1 \cup \mathcal{S}_2$ where $\mathcal{S}_1 = \{A(a), B(a)\}$ and $\mathcal{S}_2 = \{C(a)\}$. We have two PAR : $\mathcal{P}_1 = \{A(a), C(a)\}$ and $\mathcal{P}_2 = \{B(a)\}$ and only one PAR_{lex} which is $\mathcal{L} = \{A(a), C(a)\}$. ■

We propose to review in the two next subsections inconsistency-tolerant inferences based only on selecting one preferred repair.

² \models denotes the standard entailment used from flat and consistent *DL-Lite* knowledge base [Calvanese *et al.*, 2007a]

7.2.3 Possibilistic-based repair

This section briefly rewrites possibilistic-based approach of Chapter 3 when the knowledge base is stratified.

One of the interesting aspects of possibilistic knowledge bases, and more generally weighted knowledge bases, is the ability of reasoning with partially inconsistent knowledge [Dubois and Prade, 1991a]. As shown in Chapter 3, entailment in possibilistic *DL-Lite*, an extension of *DL-Lite* within a possibility theory setting, is based on the selection of one consistent, but not necessarily maximal, subbase of \mathcal{K} . This subbase is induced by a level of priority called the inconsistency degree of \mathcal{K} . The following definition reformulates the definition of inconsistency degree to fit the case where \mathcal{A} is prioritized.

Definition 7.5. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an inconsistent prioritized *DL-Lite* knowledge base. The inconsistency degree of \mathcal{K} , denoted $Inc(\mathcal{K})$, is defined as follows: $Inc(\mathcal{K}) = i + 1$ if and only if:

- i) $\langle \mathcal{T}, \mathcal{S}_1 \cup \dots \cup \mathcal{S}_i \rangle$ is consistent and,
- ii) $\langle \mathcal{T}, \mathcal{S}_1 \cup \dots \cup \mathcal{S}_{i+1} \rangle$ is inconsistent.

The subbase $\pi(\mathcal{A})$ is made of the assertions having priority levels that are strictly less than $Inc(\mathcal{K})$, namely $\pi(\mathcal{A}) = \mathcal{S}_1 \cup \dots \cup \mathcal{S}_{(Inc(\mathcal{K})-1)}$. If \mathcal{K} is consistent then we simply let $\pi(\mathcal{A}) = \mathcal{A}$. The following definition extends the possibilistic entailment (π -entailment) to the case where \mathcal{A} is stratified.

Example 7.5. Consider $\mathcal{T} = \{A \sqsubseteq \neg B\}$ and $\mathcal{A} = \mathcal{S}_1 \cup \mathcal{S}_2 \cup \mathcal{S}_3$ where $\mathcal{S}_1 = \{A(a)\}$, $\mathcal{S}_2 = \{B(a), A(b)\}$ and $\mathcal{S}_3 = \{B(b)\}$. One can check that $\pi(\mathcal{A}) = \{A(a)\}$ since $\langle \mathcal{T}, \mathcal{S}_1 \rangle$ is consistent, but $\langle \mathcal{T}, \mathcal{S}_1 \cup \mathcal{S}_2 \rangle$ is inconsistent. ■

Definition 7.6. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized *DL-Lite* knowledge base. A query q is said to be a π -consequence of \mathcal{K} , denoted $\mathcal{K} \models_{\pi} q$, if and only if $\langle \mathcal{T}, \pi(\mathcal{A}) \rangle \models q$.

The π -entailment is cautious in the sense that assertions from $\mathcal{A} \setminus \pi(\mathcal{A})$ that are not involved in any conflict are inhibited because of their low priority levels.

7.2.4 Linear-based repair

One way to recover the inhibited assertions by the possibilistic entailment is to define the linear-based repair from \mathcal{A} . The following definition introduces the notion of linear subset. Linear entailment has been used in a propositional logic setting in [Nebel, 1994] and has been applied for a DL setting (e.g. [Qi et al., 2011]).

Definition 7.7. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized *DL-Lite*. The *linear* assertional-based repair of \mathcal{A} , denoted $\ell(\mathcal{A}) = \mathcal{S}_1 \cup \dots \cup \mathcal{S}_n$, is obtained as follows:

- i) For $i = 1$: $\ell(\mathcal{S}_1) = \mathcal{S}_1$ if $\langle \mathcal{T}, \mathcal{S}_1 \rangle$ is consistent. Otherwise $\ell(\mathcal{S}_1) = \emptyset$.
- ii) For $i > 1$: $\ell(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_i) = \ell(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_{i-1}) \cup \mathcal{S}_i$ if $\langle \mathcal{T}, \ell(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_{i-1}) \cup \mathcal{S}_i \rangle$ is consistent. Otherwise $\ell(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_i) = \ell(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_{i-1})$.

Clearly, $\ell(\mathcal{A})$ is obtained by discarding a layer \mathcal{S}_i when its facts conflict with the ones involved in the previous layer.

Definition 7.8. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized *DL-Lite* knowledge base. A query q is a linear consequence (ℓ -consequence) from \mathcal{K} , denoted $\mathcal{K} \models_{\ell} q$, if and only if $\langle \mathcal{T}, \ell(\mathcal{A}) \rangle \models q$.

Example 7.6. Let us consider again $\mathcal{T} = \{A \sqsubseteq \neg B\}$ and $\mathcal{A} = \mathcal{S}_1 \cup \mathcal{S}_2 \cup \mathcal{S}_3$ where $\mathcal{S}_1 = \{A(a)\}$, $\mathcal{S}_2 = \{B(a), A(b)\}$ and $\mathcal{S}_3 = \{B(b)\}$. One can check that $\ell(\mathcal{A}) = \{A(a), B(b)\}$ since $\langle \mathcal{T}, \mathcal{S}_1 \rangle$ is consistent, $\langle \mathcal{T}, \mathcal{S}_1 \cup \mathcal{S}_2 \rangle$ is inconsistent and $\langle \mathcal{T}, \mathcal{S}_1 \cup \mathcal{S}_3 \rangle$ is consistent. ■

The subbase $\ell(\mathcal{A})$ is unique and consistent with \mathcal{T} . The following proposition gives the complexity of π -entailment and ℓ -entailment which are in P.

Proposition 7.2. *The computational complexity of π -entailment is in $\mathcal{O}(\text{cons})$ where cons is the complexity of consistency checking of standard DL-Lite. The complexity of ℓ -entailment is in $\mathcal{O}(n * \text{cons})$ where n is the number of strata in the knowledge base.*

Proof of Proposition 7.2. The proof of the complexity of π -entailment can be found in Section 3.5. The proof of the complexity of ℓ -entailment is immediate since to see whether a stratum should be kept or not in the result of restoring consistency, one consistency check is needed. □

The ℓ -entailment is more productive than π -entailment as one can see in Example 7.6 in comparison with Example 7.6, but incomparable with *PAR*-entailment and *Lex*-entailment. However from Definitions 7.5 and 7.7, both $\pi(\mathcal{A})$ and $\ell(\mathcal{A})$ are not guaranteed to be maximal.

7.3 Sensitivity to the prioritized closure

Before presenting new strategies that only select one preferred repair, we briefly introduce the concept of a prioritized closure and check which among existing approaches is sensitive to the use of the deductive closure.

The inference relations given in the previous section can be either defined on $\langle \mathcal{T}, \mathcal{A} \rangle$ or on $\langle \mathcal{T}, \circ_{cl}(\mathcal{A}) \rangle$ where \circ_{cl} denotes the deductive closure of a set of assertions. The following definition extends Definition 5.2 to the prioritized case.

Definition 7.9. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized DL-Lite knowledge base. Then, we define the prioritized closure of \mathcal{A} with respect to \mathcal{T} , simply denoted by $\circ_{cl}(\mathcal{A})$, as follows:

$$\circ_{cl}(\mathcal{A}) = \mathcal{S}'_1 \cup \dots \cup \mathcal{S}'_n$$

where:

$$\begin{aligned} \mathcal{S}'_1 &= \circ_{cl}(\mathcal{S}_1), \\ \forall i = 2, \dots, n : \mathcal{S}'_i &= \circ_{cl}(\mathcal{S}'_1 \cup \dots \cup \mathcal{S}'_i) \setminus (\mathcal{S}'_1 \cup \dots \cup \mathcal{S}'_{i-1}) \end{aligned}$$

Example 7.7. Consider $\mathcal{T} = \{A \sqsubseteq B, B \sqsubseteq C, C \sqsubseteq \neg D\}$ and $\mathcal{A} = \mathcal{S}_1 \cup \mathcal{S}_2$ where $\mathcal{S}_1 = \{A(a), D(a)\}$ and $\mathcal{S}_2 = \{B(b)\}$. Using Definition 7.9, we have $\circ_{cl}(\mathcal{A}) = \mathcal{S}'_1 \cup \mathcal{S}'_2$ where $\mathcal{S}'_1 = \{A(a), B(a), C(a), D(a)\}$ and $\mathcal{S}'_2 = \{B(b), C(b)\}$. ■

An important feature of π -inference and ℓ -inference is that they are insensitive to the deductive closure. This is not the case with *PAR*-entailment or *Lex*-entailment, more precisely:

Proposition 7.3. *Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized DL-Lite knowledge base. Then $\forall q$:*

- $\langle \mathcal{T}, \mathcal{A} \rangle \models_{\pi} q$ if and only if $\langle \mathcal{T}, \circ_{cl}(\mathcal{A}) \rangle \models_{\pi} q$.
- $\langle \mathcal{T}, \mathcal{A} \rangle \models_{\ell} q$ if and only if $\langle \mathcal{T}, \circ_{cl}(\mathcal{A}) \rangle \models_{\ell} q$.

Proof of Proposition 7.3. Intuitively, both π -inference and ℓ -inference use a consistency checking of the whole stratum to decide whether this stratum should be kept or not for restoring the consistency of the knowledge base. Besides, one can easily check that in standard *DL-Lite*, $\langle \mathcal{T}, \mathcal{A} \rangle$ is consistent if and only if $\langle \mathcal{T}, \circ_{cl}(\mathcal{A}) \rangle$ is consistent. \square

The following proposition shows that preferred inclusion-based inference and lexicographic-based inference are sensitive to the deductive closure.

Proposition 7.4. *Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized *DL-Lite* knowledge base. Then,*

- *the preferred inclusion-based inference (PAR-entailment) applied to $\langle \mathcal{T}, \mathcal{A} \rangle$ is incomparable with the one applied to $\langle \mathcal{T}, \circ_{cl}(\mathcal{A}) \rangle$.*
- *the lexicographic inference (Lex-entailment) applied to $\langle \mathcal{T}, \mathcal{A} \rangle$ is incomparable with the one applied to $\langle \mathcal{T}, \circ_{cl}(\mathcal{A}) \rangle$.*

Example 7.8 (Counterexample for *PAR-entailment*). Let $\mathcal{T} = \{A \sqsubseteq \neg B, A \sqsubseteq D, D \sqsubseteq \neg E\}$ and $\mathcal{A} = \mathcal{S}_1 \cup \mathcal{S}_2$ where $\mathcal{S}_1 = \{A(a), B(a)\}$ and $\mathcal{S}_2 = \{E(a)\}$. We have $\mathcal{P}_1 = \{A(a)\}$ and $\mathcal{P}_2 = \{B(a), E(a)\}$. Consider now the deductive closure: we have $\circ_{cl}(\mathcal{S}_1) = \{A(a), B(a), D(a)\}$ and $\circ_{cl}(\mathcal{S}_1 \cup \mathcal{S}_2) = \{E(a)\}$. We also have: $\mathcal{P}_1 = \{A(a), D(a)\}$ and $\mathcal{P}_2 = \{B(a), D(a)\}$. One can check that i) $D(a)$ is a *PAR-entailment* of $\langle \mathcal{T}, \circ_{cl}(\mathcal{A}) \rangle$ while it does not follow from $\langle \mathcal{T}, \mathcal{A} \rangle$, ii) $E(a) \vee A(a)$ is a *PAR-entailment* of $\langle \mathcal{T}, \mathcal{A} \rangle$ while it does not follow from $\langle \mathcal{T}, \circ_{cl}(\mathcal{A}) \rangle$. \blacksquare

Example 7.9 (Counterexample *Lex-entailment*). Let us consider the following cases:

- i) $\mathcal{T} = \{A \sqsubseteq \neg B, A \sqsubseteq C\}$ and $\mathcal{A} = \mathcal{S}_1 = \{A(a), B(a)\}$. We have $\langle \mathcal{T}, \mathcal{A} \rangle \not\models_{lex} C(a)$ while $\langle \mathcal{T}, \circ_{cl}(\mathcal{A}) \rangle \models_{lex} C(a)$.
- ii) $\mathcal{T} = \{A \sqsubseteq \neg B, B \sqsubseteq F, F \sqsubseteq \neg A, C \sqsubseteq \neg B\}$ and $\mathcal{S}_1 = \{A(a), B(a)\}$ and $\mathcal{S}_2 = \{C(a)\}$. We only have a lexicographic subbase of $\langle \mathcal{T}, \mathcal{S}_1 \cup \mathcal{S}_2 \rangle$ which is $\mathcal{L} = \{A(a), C(a)\}$ hence $\langle \mathcal{T}, \mathcal{L} \rangle \models_{lex} C(a)$. Besides $\circ_{cl}(\mathcal{S}_1) = \{A(a), B(a), F(a)\}$ and $\circ_{cl}(\mathcal{S}_2) = \{C(a)\}$. We also have one lexicographic subbase of $\langle \mathcal{T}, \circ_{cl}(\mathcal{A}) \rangle$ which is $\mathcal{L} = \{B(a), F(a)\}$ hence $\langle \mathcal{T}, \circ_{cl}(\mathcal{A}) \rangle \not\models_{lex} C(a)$. \blacksquare

7.4 New strategies for selecting one preferred repair

This section presents new strategies that only select one preferred repair. Selecting only one repair is important since it allows efficient query answering once the preferred repair is computed. These strategies are based on the so-called non-defeated entailment, described in the next section, by adding different criteria: deductive closure, cardinality, consistency and priorities.

7.4.1 Non-defeated repair

One way to get one preferred repair is to iteratively apply, layer per layer, the intersection of maximally assertional-based repairs (*i.e.* *MAR*). More precisely:

Definition 7.10. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized *DL-Lite* knowledge base. We define the non-defeated repair, denoted by $nd(\mathcal{A}) = \mathcal{S}'_1 \cup \dots \cup \mathcal{S}'_n$, as follows:

$$\forall i = 1, \dots, n : \mathcal{S}'_i = \bigcap_{\mathcal{R}_i \in \text{MAR}(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_i)} \mathcal{R}_i \quad (7.1)$$

As it will be shown below, the non-defeated entailment corresponds to the definition of non-defeated subbase proposed in [Benferhat *et al.*, 1998a] within a propositional logic setting. However, contrarily to the propositional setting i) the non-defeated repair can be applied on \mathcal{A} or its deductive closure $\circ_{cl}(\mathcal{A})$ which leads to two different inference relations, ii) the non-defeated repair is computed in polynomial time in a DL-Lite setting while its computation is hard in a propositional logic setting. Let us now rephrase non-defeated repair (Equation 7.1) using the concept of free inference. First, we recall the notion of non-conflicting or free elements.

Definition 7.11. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be DL-Lite knowledge base. An assertion $f \in \mathcal{A}$ is said to be *free* if and only if $\forall C \in \mathcal{C}(\mathcal{A}) : f \notin C$.

Intuitively, *free* assertions are those assertions that are not involved in any conflict. Let $\mathcal{S} \in \mathcal{A}$ be a set of assertions, we denote by $free(\mathcal{S})$ the set of *free* assertions in \mathcal{S} . The notions of *free* elements and *free-entailment* are originally proposed in [Benferhat *et al.*, 1992] where knowledge bases are encoded in a propositional logic setting. The definition of *free-entailment* is also equivalent to the MBox \mathcal{M}_1 followed by the safe inference strategy *i.e.* $\langle \mathcal{M}_1, \cap \rangle$ presented in Section 5.5.

The following proposition shows that the notion of $free(\mathcal{A})$ extended to the prioritized case gives a non-defeated repair.

Proposition 7.5. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized DL-Lite knowledge base. The non-defeated repair of \mathcal{A} , given in Definition 7.10, is equivalent to:

$$nd(\mathcal{A}) = free(\mathcal{S}_1) \cup free(\mathcal{S}_1 \cup \mathcal{S}_2) \cup \dots \cup free(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_n)$$

where $\forall i : free(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_i)$ denotes the set of free facts in $(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_i)$.

Proof. The proof is immediate since $\forall i : free(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_i) = \bigcap_{\mathcal{R} \in MAR(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_i)} \mathcal{R}$. □

The non-defeated repair is an extension of the *free* assertional base when \mathcal{A} is prioritized. The following definition introduces non-defeated entailment.

Definition 7.12. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized DL-Lite knowledge base. A query q is a non-defeated consequence (*nd-consequence*) of \mathcal{K} , denoted $\mathcal{K} \models_{nd} q$, if and only if $\langle \mathcal{T}, nd(\mathcal{A}) \rangle \models q$.

In Definition 7.11, a *free* element is not involved in any conflict. Hence the following proposition holds.

Proposition 7.6. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized DL-Lite knowledge base. Let $nd(\mathcal{A})$ be its dominant subbase. Then $\langle \mathcal{T}, nd(\mathcal{A}) \rangle$ is consistent.

Proof of Proposition 7.6. We will use the recurrence to show the proof. Namely, assume that for some i :

$$free(\mathcal{S}_1) \cup free(\mathcal{S}_1 \cup \mathcal{S}_2) \cup \dots \cup free(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_i)$$

is consistent.

This assumption is true for $i = 1$, since by definition $free(\mathcal{S}_1)$ is consistent. Let us show that the assumption holds for rank $(i + 1)$, namely:

$$free(\mathcal{S}_1) \cup free(\mathcal{S}_1 \cup \mathcal{S}_2) \cup \dots \cup free(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_{i+1})$$

is consistent.

Assume that this is not the case. This means that there exists $f \in \text{free}(\mathcal{S}_1) \cup \text{free}(\mathcal{S}_1 \cup \mathcal{S}_2) \cup \dots \cup \text{free}(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_i)$ and $g \in \text{free}(\mathcal{S}_1) \cup \text{free}(\mathcal{S}_1 \cup \mathcal{S}_2) \cup \dots \cup \text{free}(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_{i+1})$ such that (f, g) is conflicting. Since,

$$\text{free}(\mathcal{S}_1) \cup \text{free}(\mathcal{S}_1 \cup \mathcal{S}_2) \cup \dots \cup \text{free}(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_i) \subseteq \text{free}(\mathcal{S}_1) \cup \text{free}(\mathcal{S}_1 \cup \mathcal{S}_2) \cup \dots \cup \text{free}(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_{i+1})$$

this means that $f \in \text{free}(\mathcal{S}_1) \cup \text{free}(\mathcal{S}_1 \cup \mathcal{S}_2) \cup \dots \cup \text{free}(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_{i+1})$ and $g \in \text{free}(\mathcal{S}_1) \cup \text{free}(\mathcal{S}_1 \cup \mathcal{S}_2) \cup \dots \cup \text{free}(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_{i+1})$. Hence, this is a contradiction, since g cannot belong to $\text{free}(\mathcal{S}_1) \cup \text{free}(\mathcal{S}_1 \cup \mathcal{S}_2) \cup \dots \cup \text{free}(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_{i+1})$. \square

Example 7.10. Let us consider again $\mathcal{T} = \{A \sqsubseteq \neg B\}$ and $\mathcal{A} = \mathcal{S}_1 \cup \mathcal{S}_2 \cup \mathcal{S}_3$ where $\mathcal{S}_1 = \{A(a)\}$, $\mathcal{S}_2 = \{B(a), A(b)\}$ and $\mathcal{S}_3 = \{B(b)\}$. One can check that $\text{nd}(\mathcal{A}) = \{A(a), A(b)\}$. \blacksquare

The computational complexity of the computation of the *dominant* subbase of \mathcal{A} is polynomial.

Proposition 7.7. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized DL-Lite knowledge base. The complexity of *nd-entailment* is in P .

Proof of Proposition 7.7. The proof follows from the fact that computing *free* subbase is done in polynomial time. *nd-entailment* proceeds to a linear number of computations of free subbases. \square

7.4.2 Adding the deduction closure

The non-defeated inference, when it is defined on \mathcal{A} , is safe since it only uses elements of \mathcal{A} which are not involved in conflicts. One way to get a more productive inference is to use $\circ_{cl}(\mathcal{A})$ instead of \mathcal{A} . Namely, we define, a closed non-defeated repair, denoted $\text{clnd}(\mathcal{A}) = \mathcal{S}'_1 \cup \dots \cup \mathcal{S}'_n$, such that:

$$\mathcal{S}'_i = \bigcap_{\mathcal{R} \in \text{MAR}(\circ_{cl}(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_i))} \mathcal{R} \quad (7.2)$$

Example 7.11. Consider $\mathcal{T} = \{A \sqsubseteq \neg B, B \sqsubseteq C\}$ and $\mathcal{A} = \mathcal{S}_1 \cup \mathcal{S}_2$ where $\mathcal{S}_1 = \{A(a)\}$ and $\mathcal{S}_2 = \{B(a)\}$. We have $\text{MAR}(\circ_{cl}(\mathcal{S}_1)) = \{A(a)\}$ and $\text{MAR}(\circ_{cl}(\mathcal{S}_1 \cup \mathcal{S}_2)) = \{(A(a), C(a)), (B(a), C(a))\}$. Then $\text{clnd}(\mathcal{A}) = \{A(a), C(a)\}$. \blacksquare

Contrarily to π -entailment and ℓ -entailment, the following proposition shows that *nd*-inference is sensitive to the use of the deductive closure.

Proposition 7.8. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized DL-Lite knowledge base. Then $\forall q$: if $\langle \mathcal{T}, \mathcal{A} \rangle \models_{nd} q$ then $\langle \mathcal{T}, \circ_{cl}(\mathcal{A}) \rangle \models_{nd} q$. The converse is false.

Proof of Proposition 7.8. For the converse it is enough to consider $\mathcal{T} = \{E \sqsubseteq \neg B, B \sqsubseteq C, E \sqsubseteq C\}$ and $\mathcal{A} = \mathcal{S}_1 = \{E(a), B(a)\}$. We have $\text{nd}(\mathcal{A}) = \emptyset$ and $\text{nd}(\circ_{cl}(\mathcal{A})) = \{C(a)\}$. Hence $C(a)$ is an *nd*-consequence of $\langle \mathcal{T}, \circ_{cl}(\mathcal{A}) \rangle$ but it is not an *nd*-consequence of $\langle \mathcal{T}, \mathcal{A} \rangle$. \square

7.4.3 Combining linear entailment and non-defeated entailment: Adding consistency

We now present a new way to select a single preferred assertional-based repair. It consists in slightly improving linear entailment, where rather to ignore a full stratum, in case of inconsistency, one can only ignore conflicting elements.

More precisely, the linear-based non-defeated repair, denoted by $\ell\text{nd}(\mathcal{A})$, is given by the following algorithm:

Input: $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ where $\mathcal{A} = \mathcal{S}_1 \cup \dots \cup \mathcal{S}_n$

Output: $\ell nd(\mathcal{A})$

- 1: $\ell nd(\mathcal{A}) = free(\mathcal{S}_1)$
- 2: **for** $i = 2$ to n **do**
- 3: **if** $\langle \mathcal{T}, \ell nd(\mathcal{A}) \cup \mathcal{S}_i \rangle$ is consistent **then**
- 4: $\ell nd(\mathcal{A}) \leftarrow \ell nd(\mathcal{A}) \cup \mathcal{S}_i$
- 5: **else**
- 6: $\ell nd(\mathcal{A}) \leftarrow \ell nd(\mathcal{A}) \cup free(\mathcal{S}_i \cup \ell nd(\mathcal{A}))$

Algorithm 7: linear-based non-defeated repair

Example 7.12. Let $\mathcal{T} = \{A \sqsubseteq B, B \sqsubseteq \neg C\}$ and $\mathcal{A} = \mathcal{S}_1 \cup \mathcal{S}_2 \cup \mathcal{S}_3$ where $\mathcal{S}_1 = \{A(a)\}$, $\mathcal{S}_2 = \{C(a), C(b)\}$ and $\mathcal{S}_3 = \{B(b), A(c)\}$. We have $\ell nd(\mathcal{A}) = \{A(a), C(b), A(c)\}$. ■

Clearly $\ell nd(\mathcal{A})$ is consistent and it is more productive than $\pi(\mathcal{A})$ and $\ell(\mathcal{A})$, but it remains incompatible with other approaches. Note that $\ell nd(\mathcal{A}) \cup free(\mathcal{S}_i \cup \ell nd(\mathcal{A})) = \bigcap \{\mathcal{R} : \mathcal{R} \in MAR(\mathcal{S}_i \cup \ell nd(\mathcal{A})) \text{ and } \mathcal{R} \cup \ell nd(\mathcal{A})\}$ is consistent. Hence, $\ell nd(\mathcal{A})$ extends $nd(\mathcal{A})$ by only focusing on $MAR(\mathcal{S}_i \cup \ell nd(\mathcal{A}))$ that are consistent with $\ell nd(\mathcal{A})$. The nice feature of ℓnd -entailment is that the extension of ℓ -entailment and nd -entailment is done without extra computational cost. More precisely, computing $\ell nd(\mathcal{A})$ is in P.

7.4.4 Introducing cardinality in non-defeated inference

A natural question is whether one can introduce a cardinality criterion, instead of set inclusion criterion, in the definition of non-defeated repair given by Equation 7.1. Namely, we define the cardinality-based non-defeated repair as follows:

Definition 7.13. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized DL-Lite knowledge base. The cardinality-based non-defeated repair, denoted by $nd(\mathcal{A})_{card} = \mathcal{S}'_1 \cup \dots \cup \mathcal{S}'_n$, is defined as follows:

$$\forall i = 1, \dots, n : \mathcal{S}'_i = \bigcap_{\mathcal{R} \in MAR_{card}(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_i)} \mathcal{R} \quad (7.3)$$

where $MAR_{card}(\mathcal{S}) = \{\mathcal{R} : \mathcal{R} \in MAR(\mathcal{S}) \text{ and } \nexists \mathcal{R}' \in MAR(\mathcal{S}) \text{ such that } |\mathcal{R}'| > |\mathcal{R}|\}$.

One main advantage of this approach is that it produces more conclusions than the standard non-defeated inference relation. Namely, $nd(\mathcal{A}) \subseteq nd(\mathcal{A})_{card}$ where $nd(\mathcal{A})$ and $nd(\mathcal{A})_{card}$ are respectively given by Equations 7.1 and 7.3. The converse is false.

Proposition 7.9. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized DL-Lite knowledge base. Then

$$nd(\mathcal{A}) \subseteq nd(\mathcal{A})_{card}$$

where $nd(\mathcal{A})$ and $nd(\mathcal{A})_{card}$ are respectively given by Equations 7.1 and 7.3. The converse is false.

Proof. The proof follows from the fact that:

$$\forall i = 1, \dots, n : MAR_{card}(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_i) \subseteq MAR(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_i)$$

□

For the converse, consider the following counter-example.

Example 7.13 (counter-example). Let $\mathcal{T} = \{A \sqsubseteq \neg B, B \sqsubseteq \neg C\}$ and $\mathcal{A} = \mathcal{S}_1 \cup \mathcal{S}_2$ where $\mathcal{S}_1 = \{A(a), B(a)\}$ and $\mathcal{S}_2 = \{C(a)\}$. We have $nd(\mathcal{A}) = \emptyset$ while $nd(\mathcal{A})_{card} = \{A(a), C(a)\}$. ■

The main limitation of $nd(\mathcal{A})_{card}$ is that it may be inconsistent with \mathcal{T} as it is illustrated with the following example.

Example 7.14. Consider $\mathcal{T} = \{A \sqsubseteq \neg B, A \sqsubseteq \neg C\}$ and $\mathcal{A} = \mathcal{S}_1 \cup \mathcal{S}_2$ where $\mathcal{S}_1 = \{A(a)\}$ and $\mathcal{S}_2 = \{B(a), C(a)\}$. Using Equation 7.3, we have $\mathcal{S}'_1 = \{A(a)\}$ and $\mathcal{S}'_2 = \{B(a), C(a)\}$. Clearly, $nd(\mathcal{A})_{card} = \mathcal{S}'_1 \cup \mathcal{S}'_2$ contradicts \mathcal{T} . ■

One way to overcome such limitation is to only select MAR_{card} of $(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_i)$ that are consistent with $(\mathcal{S}'_1 \cup \dots \cup \mathcal{S}'_{i-1})$, namely:

Definition 7.14. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an prioritized *DL-Lite* knowledge base. We define the consistent cardinality-based non-defeated repair, denoted by $consnd(\mathcal{A})_{card} = \mathcal{S}'_1 \cup \dots \cup \mathcal{S}'_n$ such that:

$$\mathcal{S}'_1 = \bigcap_{\mathcal{R} \in MAR_{card}(\mathcal{S}_1)} \mathcal{R}$$

$$\forall i = 2, \dots, n : \mathcal{S}'_i = \bigcap \{ \mathcal{R} : \mathcal{R} \in MAR_{card}(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_i) \text{ and } \mathcal{R} \text{ is consistent with } \mathcal{S}'_1 \cup \dots \cup \mathcal{S}'_{i-1} \}$$

Clearly, contrarily to $nd(\mathcal{A})_{card}$, $consnd(\mathcal{A})_{card}$ is always consistent.

Example 7.15. Consider the example where $\mathcal{T} = \{A \sqsubseteq \neg B, A \sqsubseteq \neg C\}$ and $\mathcal{A} = \mathcal{S}_1 \cup \mathcal{S}_2$ where $\mathcal{S}_1 = \{A(a)\}$ and $\mathcal{S}_2 = \{B(a), C(a)\}$. We have $\mathcal{S}'_1 = \{A(a)\}$ and $\mathcal{S}'_2 = \emptyset$. Clearly $consnd(\mathcal{A})_{card}$ is consistent with \mathcal{T} . ■

7.4.5 Adding priorities to non-defeated inference

In the definition of *nd*-inference, given by Equation 7.1, a flat notion of *MAR* (maximally inclusion-based repair) has been used. A natural way to extend the *nd*-entailment is to use a prioritized version of *MAR* (*i.e.* *PAR*), namely:

Definition 7.15. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an prioritized *DL-Lite* knowledge base. We define the prioritized inclusion-based non-defeated repair, denoted by $pind(\mathcal{A}) = \mathcal{S}'_1 \cup \dots \cup \mathcal{S}'_n$, as follows:

$$\forall i = 1, \dots, n : \mathcal{S}'_i = \bigcap_{\mathcal{P} \in PAR(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_i)} \mathcal{P} \quad (7.4)$$

The following proposition shows that there is no need to consider all \mathcal{S}'_i for $i < n$ when computing $pind(\mathcal{A})$, namely:

Proposition 7.10. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized *DL-Lite* knowledge base. Then

$$pind(\mathcal{A}) = \bigcap_{\mathcal{P} \in PAR(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_n)} \mathcal{P}.$$

Proof. The proof follows from the fact that $\forall i < n, \forall A \in PAR(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_{i+1}), \exists B \in PAR(\mathcal{S}_1 \cup \dots \cup \mathcal{S}_i)$ such that $B \subseteq A$. Namely, prioritized inclusion-based maximal repairs from $\mathcal{S}_1 \cup \dots \cup \mathcal{S}_{i+1}$ are obtained from prioritized inclusion-based maximal repairs from $\mathcal{S}_1 \cup \dots \cup \mathcal{S}_i$ by adding some elements from \mathcal{S}_{i+1} . Hence, for $i < n$, we have:

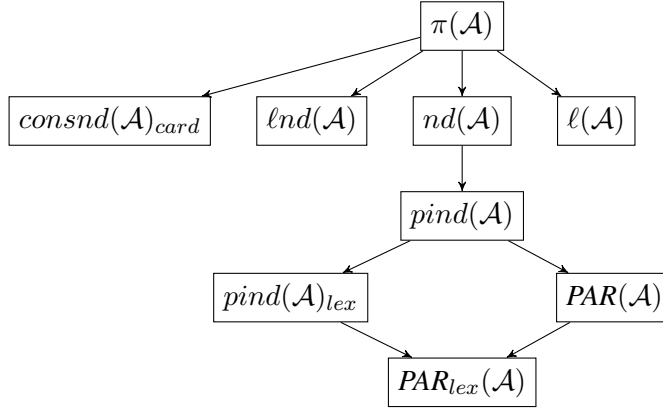


Figure 7.1: Relationships between inferences

$$\bigcap_{B \in \text{PAR}(S_1 \cup \dots \cup S_i)} B \subseteq \bigcap_{B \in \text{PAR}(S_1 \cup \dots \cup S_{i+1})} B$$

Therefore:

$$\text{pind}(\mathcal{A}) = \bigcap_{B \in \text{PAR}(S_1 \cup \dots \cup S_n)} B$$

□

Besides, a cardinality-based version of Equation 7.4, denoted by $\text{pind}(\mathcal{A})_{lex} = S_1 \cup \dots \cup S_n$, can be defined as follows:

$$\forall i = 1, \dots, n : \mathcal{S}'_i = \bigcap_{\mathcal{L} \in \text{PAR}_{lex}(S_1 \cup \dots \cup S_n)} \mathcal{L} \quad (7.5)$$

Lastly, both $\text{pind}(\mathcal{A})$ and $\text{pind}(\mathcal{A})_{lex}$ can be defined on $\circ_{cl}(\mathcal{A})$ instead of \mathcal{A} or be defined on closed repairs instead of repairs themselves. This leads to new inferences strategies that only select one preferred subbase.

7.5 Comparative analysis

From a computational complexity point of view, π -entailment, ℓ -entailment, nd -entailment and ℓnd -entailment and the entailments based on their closures, are the most promising ones since both computing the repair and query answering are tractable. For other strategies based on the nd -inference, computing the repairs is a hard task, but it is done *ONCE*. Answering queries, when the single repair is computed, is efficiently computed since it has the same complexity as in standard *DL-Lite*.

From productivity, Figure 7.1 summarizes the relationships between main entailments considered in the paper when the ABox is prioritized. Note that for the sake of simplicity, we do not make reference in Figure 7.1 to inferences defined on $\circ_{cl}(\mathcal{A})$.

In Figure 7.1, $n1 \rightarrow n2$ means that each conclusion that can be universally derived from repairs in $n1$ is also a conclusion using repairs in $n2$. From Figure 7.1, π -entailment is the most cautious relation. Adding priorities, cardinality and consistency to the definition of nd -entailment allow to provide more productive

inference relations. However ℓ -entailment remains incomparable with the nd -entailment, since layers including non free assertions can be present in $\ell(\mathcal{A})$. Moreover, $\ell nd(\mathcal{A})$ is incomparable with other approaches. Within the prioritized setting, $nd(\mathcal{A})$ plays the same role with respect to PAR as $free(\mathcal{A})$ for MAR in the flat case. As a consequence, each nd -consequence of \mathcal{A} is also a PAR -consequence of \mathcal{A} . The converse is false. Moreover, it is well-known that each PAR -entailment is also a Lex -entailment and the converse is false, since the Lex -entailment only uses subsets of prioritized repair (PAR).

7.6 Experimental evaluation

In this section, we present an experimental evaluation on the computation of main repairs proposed in this Chapter using some algorithms proposed in Chapter 6.

All the experiments are performed on a MacBook Pro laptop with 2.6GHz Intel Core (i7) processor and 16GB 1600 MHz DDR3 RAM. We considered a TBox containing 100 negative inclusion axioms with a proportion of conflicts at least equal to 1/5 per assertion. This TBox is adapted from the *DL-Lite_R* university benchmark proposed in [Lutz *et al.*, 2013]. We use the Extended University Data Generator (EUDG)³ to generate the *ABox* assertions. Once the *ABox* is produced, we fit it to our setting using 4 strata until 7 strata. Moreover the computation of conflicts is performed layer per layer. Note that the time used for computing the conflicts is not included in the time used for computing the repairs, since this is done in a polynomial time. Said differently, computing conflicts is negligible with respect to computing repairs.

# conflict	# <i>MAR</i>	time # <i>MAR</i>	# <i>MAR</i> _{card}	time # <i>MAR</i> _{card}
18	28080	105ms	192	65ms
25	688128	2268ms	256	789ms
37	16815986	206089ms	56	5422ms
75	20160000	272830ms	96	216236ms
105	-	Time-out	2034	8259s

Table 7.2: Number of conflicts, number of *MAR*, time taken to compute *MAR* in *ms* (milliseconds) or *s* (seconds), number of #*MAR*_{card}, time taken to compute #*MAR*_{card}.

Table 7.2 gives the experimental results of the computation of *MAR* and *MAR*_{card}. One can see that using the cardinality criterion instead of the set inclusion one refines the result and improves the computation time of the repairs. Moreover, an important influential parameter when computing the repairs is the number of occurrences of an assertion in conflicts. Namely, the more an assertion is recurring in conflicts the more the conflict resolution has better chances to be achieved. For instance, in Table 7.2 considering the case of 37 conflicts, by increasing the percentage of occurrences of some assertions in conflicts, we obtain 23082 *MAR* in 136ms instead of 16815986 in 206089ms. In such case, the number of *Lex* decreases also where we compute only 24 #*MAR*_{card} having cardinality equal to 14 assertions. Similar results on the effect of the number of occurrences of assertions in conflicts are provided [Pivert and Prade, 2010; Deagustini *et al.*, 2014].

Now, concerning PAR_{lex} , we also use the notion of minimal inconsistent subsets where the minimality refers to a lexicographic ordering. Table 7.3 gives the results on the computation of PAR_{lex} and the main repairs given in this paper. One can first observe that given an *ABox* \mathcal{A} whatever is its size, computing π or ℓ does not need long computation time as needed by inconsistency checking. Regarding now the computation of the non-defeated repair, it depends on the number of conflicts in the *ABox*. Another

³available at <https://code.google.com/p/combo-obda/>

parameter that also influences the results is the number of layers. This can be clearly seen when computing $\#PAR_{lex}$. Indeed, the number of PAR_{lex} decreases as the number of layers increases. Clearly, more the stratification of the ABox is important more the conflicts resolution has better chances to be achieved.

# Conflicts	# Strata	time π	time ℓ	time nd	$\#PAR_{lex}$	time PAR_{lex}
61	4	4ms	7ms	7ms	16	17ms
	7	4ms	8ms	6ms	2	11ms
123	4	5ms	8ms	10ms	16	43ms
	7	4ms	8ms	9ms	4	38ms
502	4	5ms	9ms	24ms	2024	1072ms
	7	5ms	8ms	13ms	128	90ms
1562	4	4ms	8ms	129ms	1392	128:47s
	7	5ms	8ms	64ms	232	34:52s

Table 7.3: Number of conflicts, number of strata, time taken to compute π , ℓ , nd and PAR_{lex} and number of computed PAR_{lex} .

7.7 Conclusion

This chapter focuses on how to produce a single preferred repair from a prioritized inconsistent *DL-Lite* knowledge base based on the notion of the non-defeated inference relation. We first reviewed some well-known approaches that select one repair (such as possibilistic repair or linear-based repair) or several repairs (such as preferred inclusion-based repairs or lexicographic-based repairs). Then, we presented different strategies for selecting one preferred repair. These strategies have as starting point the non-defeated repair and mainly add one/several of the four main criteria: priorities, deductive closure, cardinality and consistency.

CONCLUSION

In this thesis, we studied dynamics of beliefs and uncertainty management in *DL-Lite*. In the first part we investigated the problem of uncertainty handling in *DL-Lite*. We first proposed an extension of the main fragments of *DL-Lite* to deal with the uncertainty associated with axioms using a possibility theory framework. We introduced the syntax and the semantics of such extensions. We provided the properties of π -*DL-Lite* and showed how to compute the inconsistency degree of π -*DL-Lite* knowledge base using query evaluation obtained by defining the π -*DL-Lite* negative closure that extends the one of standard *DL-Lite*. This extension permits to deal with priorities or uncertainty degrees (between *DL-Lite* axioms) without increasing the computational complexity. A tool for representing and reasoning in possibilistic *DL-Lite* framework is implemented. We introduced main reasoning tasks in π -*DL-Lite* and we provided a preliminary result on conditioning π -*DL-Lite* knowledge bases when a new piece of information is available. Lastly, we proposed a min-based assertional merging operator when the assertions of ABox are provided by several sources of information having different levels of priority. We showed that this operator is more productive than the classical one, without increasing the complexity of the merging process.

In the second part of the thesis, we studied the problem of inconsistency handling in flat *DL-Lite* knowledge bases. We generalized techniques for inconsistency handling in flat knowledge bases by introducing the notions of modifier and inference strategy. We showed that the combination of modifiers and inference strategies provides a principled and exhaustive list of techniques for inconsistency management. We studied the productivity and the computational complexity for the proposed techniques and showed how our work extends the state of the art.

The third part of this thesis was dedicated to the revision and inconsistency handling in prioritized *DL-Lite* knowledge bases where the ABox is stratified. We first investigated the problem of revising by considering different forms of input information, namely when the input is an ABox assertion or a TBox axiom. We proposed syntactic revision operators, called Prioritized Removed Sets (PRSR) operators. These operators follow a lexicographical strategy for removing some assertions, namely the prioritized removed sets, in order to restore consistency. We showed in particular that when the input is an ABox axiom, the revision process leads to a unique revised knowledge base. However, when the input is a TBox axiom, the revision process may lead to several possible revised knowledge bases. In this case, we defined selection functions in order to keep the result within the *DL-Lite* language. We provided the logical properties of PRSR operators using Hansson's postulates rephrased within our framework. Finally, we proposed algorithms to computing the prioritized removed sets where some of them are based on the notion of hitting set.

We studied the problem of inconsistency handling in prioritized *DL-Lite* knowledge bases. We introduced several inconsistency-tolerant inference approaches in *DL-Lite* when the ABox is stratified. We first reviewed some well-known approaches that select one assertional-based repair or several assertional-based repairs, and then, proposed different strategies for selecting a single assertional-based repair. These strategies have as starting point the non-defeated assertional-based repair and mainly adding one/several of four main ingredients: priorities, deductive closure, cardinality and consistency. We showed that these inconsistency-tolerant relations are proper to *DL-Lite* logics and some of them allow tractable handling of inconsistency without additional complexity in comparison to standard *DL-Lite*. We provided complexity results and experimental studies showing the efficiency of the proposed entailments.

Future works

Uncertainty management The inconsistency notion used Chapter 3 is fully in the spirit of the one used in standard *DL-Lite*. In presence of certainty degrees, one may consider other forms of inconsistency that depart from the one used in standard *DL-Lite* logic. This depends on the uncertainty framework and the knowledge modeling steps may play a crucial role. For instance, if one uses a probability theory framework, and roughly speaking if $\langle A \sqsubseteq B, \alpha \rangle$ is interpreted as “the conditional probability of an individual to be in B given the fact that he is A is α , then $\langle A \sqsubseteq B, .9 \rangle$ and $\langle A \sqsubseteq \neg B, .1 \rangle$ are closely related. Hence, the way we write uncertain pieces has an important role and the definition of inconsistency degrees should take into account the used uncertainty framework. Note that in a possibility theory setting, an axiom φ and its negated axiom $\neg\varphi$ are only weakly related since we only have $\min(N(\varphi), N(\neg\varphi)) = 0$. Besides, in the future, we plan to explore more parsimonious definitions of inconsistency degrees. For instance, in the rules used for defining the π -negated closure of a *DL-Lite* knowledge base we used the minimum operator for propagating certainty degrees. The question is how to use other operators (such that leximin or descremin [Benferhat *et al.*, 1993a; Dubois *et al.*, 1992] or simply the product operators) that lead to new definitions of inconsistency degrees. The aim is to get more productive relations that only provide safe conclusions and run in a polynomial time.

Another future work is to consider richer extensions of *DL-Lite* logics such as *DL-Lite*_{R,□} and *DL-Lite*_{F,□} [Calvanese *et al.*, 2006] (*DL-Lite*_{horn}^H and *DL-Lite*_{horn}^F presented in Section 1.4.2) that allow conjunction on the left side of a concept inclusion. We are also interested in considering possibilistic *DL-Lite* with disjunctive ABox as it has been done in databases [Molinaro *et al.*, 2009]. Namely, we plan to enrich DL-languages with the propositional disjunction " $A \vee B$ ". The question is then how to extend the concepts of conflicts, associated with negated axioms, to define inconsistency degrees of possibilistic knowledge bases with disjunctive ABox. This extension is useful when one has to merge multiple sources information, where the union of the TBox (issued from each source) is consistent, but the whole set of information is conflicting.

Knowledge base fusion This thesis opens several perspectives regarding the fusion issue. For instance, we focus on a *min* operator for aggregating conflict vectors, in order to preserve possibilistic semantics. Nevertheless, other aggregation operators can be considered (*e.g.* the product operator) or direct comparisons from vectors (*e.g.* G-max based operator). From a postulates point of view, other postulates dedicated to DL knowledge bases could be studied and adapted (*e.g.* arbitration [Konieczny and Pino Pérez, 2002]). Moreover, we plan to investigate other measures of conflicts such as Shapley measure proposed in [Hunter and Konieczny, 2010]. Finally, we will investigate the extension of Removed Sets Fusion [Hué *et al.*, 2008], defined in a propositional setting, to the merging of *DL-Lite* knowledge bases. Another extension is to study merging of uncertain *DL-Lite* knowledge bases in the context where uncertainty scales are incommensurable [Benferhat *et al.*, 2007].

Ontological-based revision vs assertional based revision When revising a prioritized *DL-Lite* knowledge base, we only considered the case where the input is a single assertion or a simple positive or negative axiom. A future work will investigate the case where the input is a set of assertions or a set of axioms. A future work will focus on a deeper study of the computational complexity of the PRSR operators. We plan also to study revision of TBox, when a new axiom (*i.e.* rule) is available. We will study revision in the general case, namely when the revision process comes down to throw out both TBox axioms and ABox assertions to restore consistency.

Regarding inconsistency handling, other modifiers such as merging or concatenating modifiers have to be investigated. Besides, the considered framework may be reapplied easily for other

richer *DL-Lite* logics (or rules-based languages (e.g. [Baget *et al.*, 2009; Baget *et al.*, 2011; Cali *et al.*, 2012]) provided that data is separated from generic knowledge.

Appendix

APPENDIX OF CHAPTER 3

Possibilistic *DL-Lite* ontology tool

In this appendix, we present a tool for representing and reasoning in possibilistic *DL-Lite* framework. This tool is developed in JAVA and it implements the inconsistency check algorithm based on query evaluation presented in Chapter 3. In this tool, the ABox is managed using SQL database engine. As ontology language, we use the *OWL-QL* functional syntax extended with the ability to attach weights to axioms. To this end, we developed our proper parser in order to manage weighted axioms. In what follows, we present some screenshots of this tool followed by a explanation about its features.

Ontology representation

The following figure presents the main interface of the possibilistic *DL-Lite* ontology editor. The main interface is partitioned into two panels: one for specifying the TBox and the other for expressing the ABox.

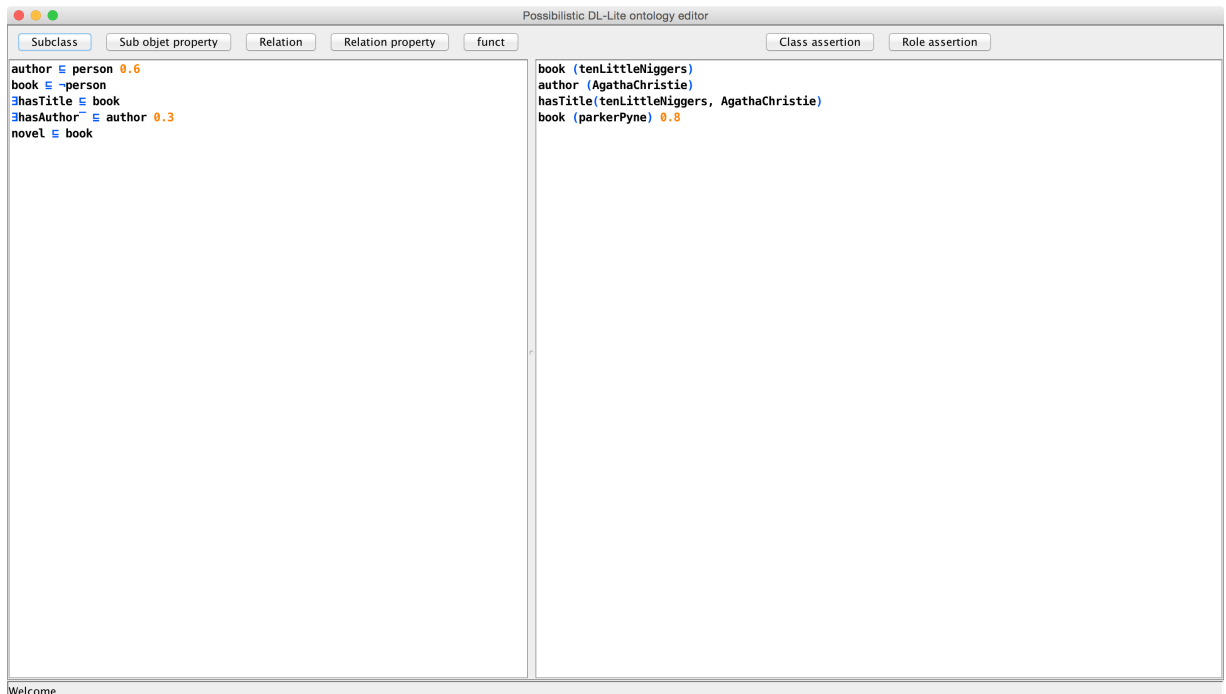


Figure A.1: Main interface

Using the different buttons of the toolbar, one can express either π -*DL-Lite*_{core}, π -*DL-Lite*_F or π -*DL-Lite*_R axioms. As presented in Figure A.2, one can express either flat axioms (*i.e.* when the weight

associated to the axioms is equal to 1. We do not represent in the interface weights equal to 1 or weighted axioms. Note that when all weights are set to 1, then we represent a standard *DL-Lite* knowledge base.

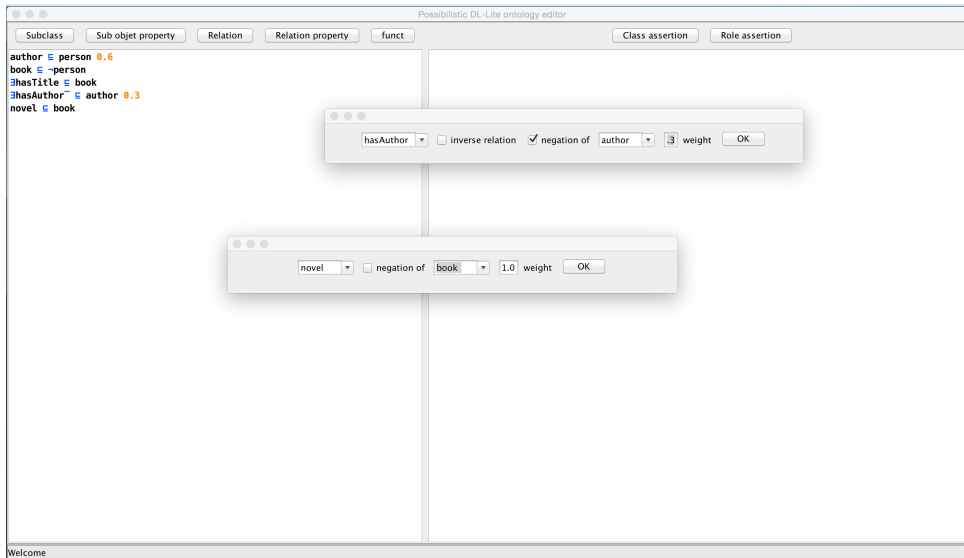


Figure A.2: Example of TBox axioms

Once the TBox is edited, one can export it using extended *OWL-QL* functional syntax as follows:

```
Weighted(SubClassOf(author , person), 0.6)
DisjointClasses(book , person)
ObjectPropertyDomain(hasTitle, book )
Weighted(ObjectPropertyRange(hasAuthor, author ), 0.3)
SubClassOf(novel, book)
```

In fact, when developing this tool, we could not use the existing ontology programming tools such as Jena, OWL API (see Section 1.2). Hence, we developed our proper parser. Note that when all weights are equal to 1, one can manipulate standard *OWL-QL* functional syntax. Clearly, one can read any standard *OWL-QL* file, edited using a standard ontology editor tool such as Protégé (see Section 1.2) and add different weights. Finally, we adapted such encoding with the aim to add other reasoning capabilities to our tool when dealing with uncertainty or priorities.

Similarly, the ABox is expressed using weighted or flat assertions as presented in Figure A.3. Note that ABox is managed using an SQL-lite¹ engine.

¹<https://www.sqlite.org>

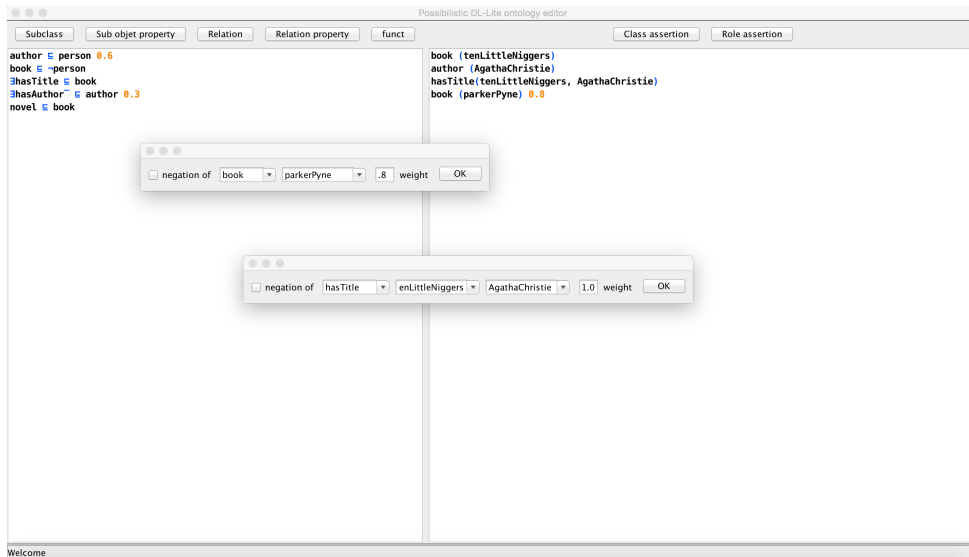


Figure A.3: Example of ABox assertions

Reasoning tasks

The current version of the tool implements two main reasoning algorithms. The first reasoning task concerns the detection of modeling errors when specifying the ontology, namely the incoherency of the TBox (see Definition 1.13). An important feature of our tool, compared with existing ones (*i.e.* flat case) is that incoherency detection is done instantly after adding any axiom to the TBox. Once an incoherency is detected and according to the weights of formulas, the editor points out the source of incoherency using "italic style". Figure A.4 gives an example.

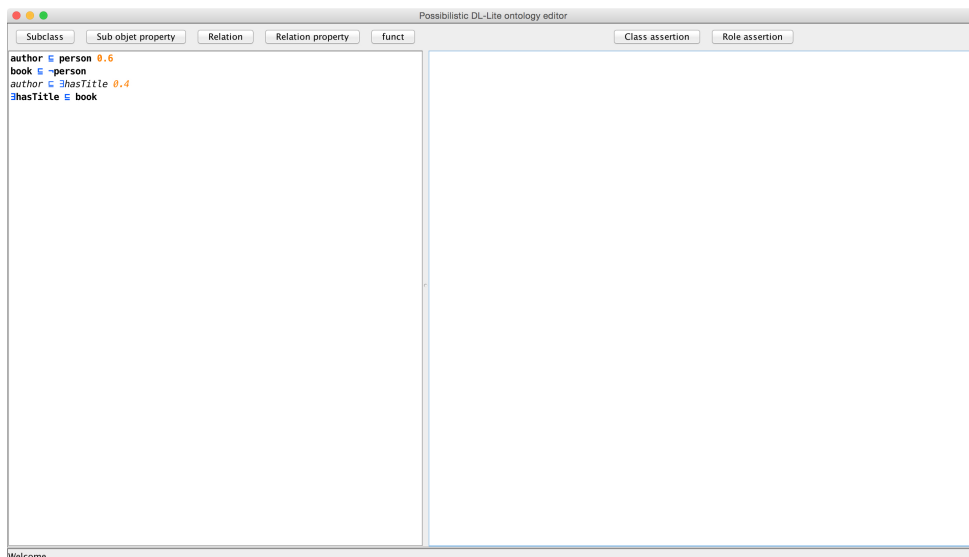


Figure A.4: Incoherency detection

The second main reasoning task is the computation of the inconsistency degree. The tool computes first the negated closure using the rules presented in Section 3.4. As for incoherency checking, the computation of the negated closure is done instantly after adding any axiom. This closure is transformed to

weighted queries performed over the set of individuals in the ABox in order to compute the inconsistency degree. The inconsistency associated with a query and a given tuple of assertions provided as an answer for the query is the maximum weight among all the certainty degrees of the query and this tuple. The maximum among these inconsistency degrees is the inconsistency degree associated with the knowledge base. Figure A.5 shows an example of computation of inconsistency degree. Once the inconsistency degree is computed, all axioms having weights less or equal to this degree are highlighted using "italic style".

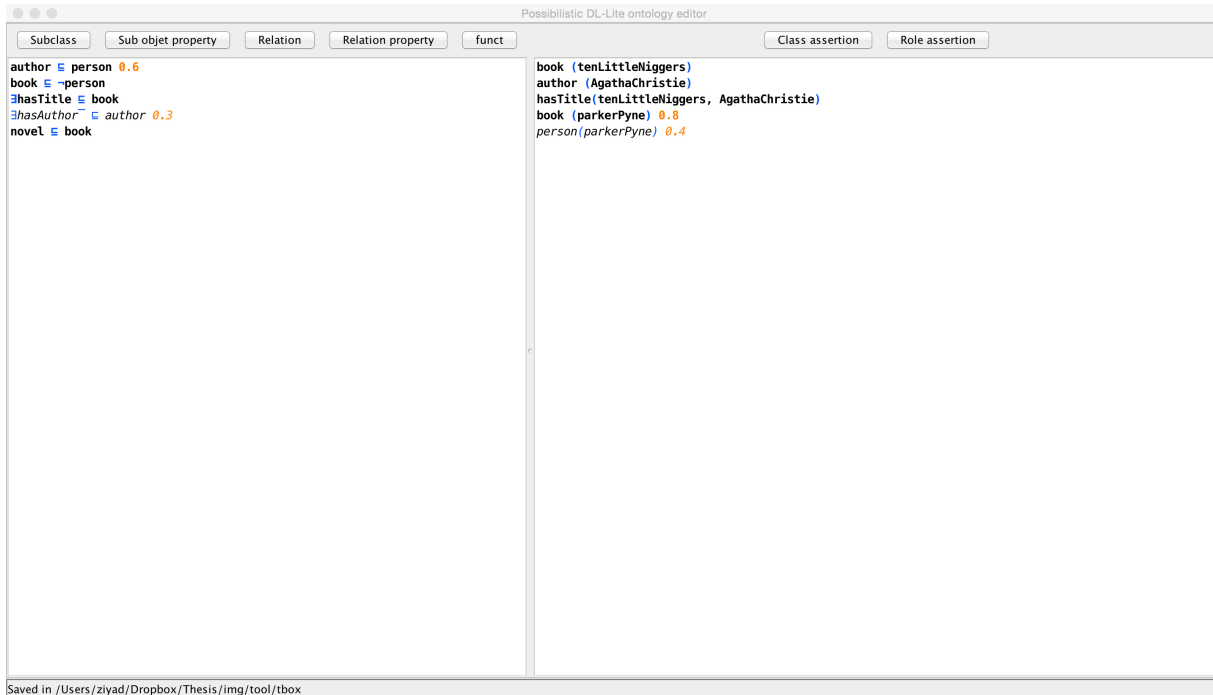


Figure A.5: Inconsistency degree

Lastly, the possibilistic *DL-Lite* tool permits to compute and export ABox conflicts as follows.

```
fconflict (book (pakerpyne) , person (pakerpyne) )
pconflict ( (book (pakerPyne) , 0.8) , (person (pakerPyne) , 0.4) )
```

Note that these conflict sets are used for experimentation in Section 7.6.

APPENDIX OF CHAPTER 5

Handling inconsistent *DL-Lite* knowledge bases = Composite modifiers+inference strategies

Proposition B.1 (Figure 5.3). *Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} = \{\mathcal{A}\} \rangle$ be an inconsistent *DL-Lite* knowledge base. Let $\{\mathcal{M}_1, \dots, \mathcal{M}_8\}$ be the eight composite modifiers summarized in Table 5.1. Then:*

1. $\mathcal{M}_2 \subseteq \mathcal{M}_1$.
2. $\mathcal{M}_4 \subseteq \mathcal{M}_3$.
3. $\mathcal{M}_6 \subseteq \mathcal{M}_5$.
4. $\mathcal{M}_8 \subseteq \mathcal{M}_7$.
5. $\mathcal{M}_3 = \circ_{cl}(\mathcal{M}_2)$.
6. $\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1)$.
7. $\mathcal{M}_2 \subseteq_{cl} \mathcal{M}_5$.
8. $\mathcal{M}_3 \subseteq \mathcal{M}_5$.
9. $\mathcal{M}_5 \subseteq_{incl} \mathcal{M}_7$.

Proof of Proposition B.1. The proof are as follows:

- Items 1-4 follow from the definition of the elementary modifier \circ_{card} . Since \circ_{card} selects subsets of \mathcal{M} having maximal cardinality. Namely, given \mathcal{M} an MBox, we have $\circ_{card}(\mathcal{M}) \subseteq \mathcal{M}$. Hence relations $\mathcal{M}_4 \subseteq \mathcal{M}_3$, $\mathcal{M}_2 \subseteq \mathcal{M}_1$, $\mathcal{M}_6 \subseteq \mathcal{M}_5$, and $\mathcal{M}_8 \subseteq \mathcal{M}_7$ holds.
- Items 5-6 follow immediately from the definition of the elementary modifier \circ_{cl} , hence we trivially have $\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1)$ and $\mathcal{M}_3 = \circ_{cl}(\mathcal{M}_2)$.
- Let us show that $\mathcal{M}_2 \subseteq_{cl} \mathcal{M}_5$, namely $\forall A \in \mathcal{M}_2, \exists B \in \mathcal{M}_5$ such that $B = Cl(A)$. The proof is immediate. Recall that $\mathcal{M}_2 \subseteq \mathcal{M}_1$, hence $\forall A \in \mathcal{M}_2$ we also have $A \in \mathcal{M}_1$. Recall also that $\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1)$. This means that $\forall A \in \mathcal{M}_2, \exists B \in \mathcal{M}_5$ such that $B = Cl(A)$.
- Regarding the proof of Item 8, we have $\mathcal{M}_2 \subseteq_{cl} \mathcal{M}_5$. This means that $\forall A \in \mathcal{M}_2$, there exists $B \in \mathcal{M}_5$ such that $B = Cl(A)$. Said differently, $\forall A \in \mathcal{M}_2$, we have $Cl(A) \in \mathcal{M}_5$. Since $\mathcal{M}_3 = \circ_{cl}(\mathcal{M}_2)$, we conclude that $\mathcal{M}_3 \subseteq \mathcal{M}_5$.
- We now show that $\mathcal{M}_5 \subseteq_{incl} \mathcal{M}_7$. Let $B \in \circ_{incl}(\{\mathcal{A}\})$ and let us show that there exists a set of assertions X such that $\circ_{cl}(\{B\}) \subseteq X$ and $X \in \mathcal{M}_7$. Since $B \in \circ_{incl}(\{\mathcal{A}\})$, this means by definition that $B \subseteq \mathcal{A}$ and hence $B \subseteq \circ_{cl}(\mathcal{A})$. Now, B is consistent, this means that there exists $R \in \circ_{incl}(\circ_{cl}(\mathcal{A})) = \mathcal{M}_7$ such that $B \subseteq R$. From Lemma 5.1, R is a closed set of assertions, then this means that $Cl(B) \subseteq R$.

□

Example B.1 (Counter-examples relations given in Figure 5.3). The converse of relations given in Figure 5.3 does not holds.

1. The converse of $\mathcal{M}_2 \subseteq \mathcal{M}_1$ does not hold.

Let $\mathcal{T} = \{B \sqsubseteq C, C \sqsubseteq \neg D\}$ and $\mathcal{M} = \{\{B(a), C(a), D(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$\mathcal{M}_1 = \circ_{incl}(\mathcal{M}) = \{\{C(a), B(a)\}, \{D(a)\}\}$, and

$\mathcal{M}_2 = \circ_{card}(\mathcal{M}_1) = \{\{C(a), B(a)\}\}$.

One can check that $\mathcal{M}_1 \not\subseteq \mathcal{M}_2$.

2. The converse of $\mathcal{M}_4 \subseteq \mathcal{M}_3$ does not hold.

Let $\mathcal{T} = \{A \sqsubseteq B, B \sqsubseteq \neg C\}$ and $\mathcal{M} = \{\{A(a), C(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$\mathcal{M}_1 = \circ_{incl}(\mathcal{M}) = \{\{A(a)\}, \{C(a)\}\}$,

$\mathcal{M}_2 = \circ_{card}(\mathcal{M}_1) = \{\{A(a)\}, \{C(a)\}\}$,

$\mathcal{M}_3 = \circ_{cl}(\mathcal{M}_2) = \{\{A(a), B(a)\}, \{C(a)\}\}$, and

$\mathcal{M}_4 = \circ_{card}(\mathcal{M}_3) = \{\{A(a), B(a)\}\}$.

One can check that $\mathcal{M}_3 \not\subseteq \mathcal{M}_4$.

3. The converse of $\mathcal{M}_6 \subseteq \mathcal{M}_5$ does not hold.

Let $\mathcal{T} = \{B \sqsubseteq C, C \sqsubseteq \neg D\}$ and $\mathcal{M} = \{\{B(a), D(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$\mathcal{M}_1 = \circ_{incl}(\mathcal{M}) = \{\{B(a)\}, \{D(a)\}\}$,

$\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1) = \{\{C(a), B(a)\}, \{D(a)\}\}$, and

$\mathcal{M}_6 = \circ_{card}(\mathcal{M}_5) = \{\{C(a), B(a)\}\}$

One can check that $\mathcal{M}_5 \not\subseteq \mathcal{M}_6$.

4. The converse of $\mathcal{M}_8 \subseteq \mathcal{M}_7$ does not hold.

Let $\mathcal{T} = \{A \sqsubseteq B, B \sqsubseteq \neg D\}$ and $\mathcal{M} = \{\{A(a), D(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$\circ_{cl}(\mathcal{M}) = \{\{A(a), B(a), D(a)\}\}$,

$\mathcal{M}_7 = \circ_{incl}(\circ_{cl}(\mathcal{M})) = \{\{A(a), B(a)\}, \{D(a)\}\}$, and

$\mathcal{M}_8 = \circ_{card}(\mathcal{M}_7) = \{\{A(a), B(a)\}\}$

One can check that $\mathcal{M}_7 \not\subseteq \mathcal{M}_8$.

5. The converse of $\mathcal{M}_2 \subseteq_{cl} \mathcal{M}_5$ does not hold.

Let $\mathcal{T} = \{A \sqsubseteq B, B \sqsubseteq C, C \sqsubseteq \neg D\}$ and $\mathcal{M} = \{\{A(a), B(a), D(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$\mathcal{M}_1 = \circ_{incl}(\mathcal{M}) = \{\{A(a), B(a)\}, \{D(a)\}\}$,

$\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1) = \{\{A(a), B(a), C(a)\}, \{D(a)\}\}$, and

$\mathcal{M}_2 = \circ_{card}(\mathcal{M}_1) = \{\{A(a), B(a)\}\}$.

One can check that $\mathcal{M}_5 \not\subseteq_{cl} \mathcal{M}_2$.

6. The converse of $\mathcal{M}_3 \subseteq \mathcal{M}_5$ does not hold.

Let $\mathcal{T} = \{A \sqsubseteq B, B \sqsubseteq C, C \sqsubseteq \neg D\}$ and $\mathcal{M} = \{\{A(a), B(a), D(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$$\begin{aligned}\mathcal{M}_1 &= \circ_{incl}(\mathcal{M}) = \{\{A(a), B(a)\}, \{D(a)\}\}, \\ \mathcal{M}_5 &= \circ_{cl}(\mathcal{M}_1) = \{\{A(a), B(a), C(a)\}, \{D(a)\}\}, \\ \mathcal{M}_2 &= \circ_{card}(\mathcal{M}_1) = \{\{A(a), B(a)\}\}, \text{ and} \\ \mathcal{M}_3 &= \circ_{cl}(\mathcal{M}_2) = \{\{A(a), B(a), C(a)\}\}.\end{aligned}$$

One can check that $\mathcal{M}_5 \not\subseteq \mathcal{M}_3$.

7. The converse of $\mathcal{M}_5 \subseteq_{incl} \mathcal{M}_7$ does not hold.

$$\text{Let } \mathcal{T} = \{A \sqsubseteq \neg B, B \sqsubseteq D\} \text{ and } \mathcal{M} = \{\{A(a), B(a)\}\}.$$

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$$\begin{aligned}\circ_{cl}(\mathcal{M}) &= \{\{A(a), B(a), D(a)\}\}, \\ \mathcal{M}_7 &= \circ_{incl}(\circ_{cl}(\mathcal{M})) = \{\{A(a), D(a)\}, \{B(a), D(a)\}\}, \\ \mathcal{M}_1 &= \circ_{incl}(\mathcal{M}) = \{\{A(a)\}, \{B(a)\}\}, \text{ and} \\ \mathcal{M}_5 &= \circ_{cl}(\mathcal{M}_1) = \{\{A(a)\}, \{B(a), D(a)\}\},\end{aligned}$$

One can check that $\mathcal{M}_7 \not\subseteq_{incl} \mathcal{M}_5$.

Corollary B.1. *Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} = \{A\} \rangle$ be an inconsistent DL-Lite knowledge base. Let $\{\mathcal{M}_1, \dots, \mathcal{M}_8\}$ be the eight composite modifiers summarized in Table 5.1. Then:*

1. $\forall \mathcal{A}_i \in \mathcal{M}_3, \exists \mathcal{A}_j \in \mathcal{M}_1$ such that $\mathcal{A}_i = Cl(\mathcal{A}_j)$.
2. $\forall \mathcal{A}_i \in \mathcal{M}_4, \exists \mathcal{A}_j \in \mathcal{M}_1$ such that $\mathcal{A}_i = Cl(\mathcal{A}_j)$.
3. $\forall \mathcal{A}_i \in \mathcal{M}_6, \exists \mathcal{A}_j \in \mathcal{M}_1$ such that $\mathcal{A}_i = Cl(\mathcal{A}_j)$.
4. $\forall \mathcal{A}_i \in \mathcal{M}_1, \exists \mathcal{A}_j \in \mathcal{M}_7$ such that $\mathcal{A}_i \subseteq \mathcal{A}_j$.
5. $\forall \mathcal{A}_i \in \mathcal{M}_1, \exists \mathcal{A}_j \in \mathcal{M}_8$ such that $\mathcal{A}_i \subseteq \mathcal{A}_j$.
6. $\forall \mathcal{A}_i \in \mathcal{M}_4, \exists \mathcal{A}_j \in \mathcal{M}_2$ such that $\mathcal{A}_i = Cl(\mathcal{A}_j)$.
7. $\forall \mathcal{A}_i \in \mathcal{M}_2, \exists \mathcal{A}_j \in \mathcal{M}_7$ such that $\mathcal{A}_i \subseteq \mathcal{A}_j$.
8. $\forall \mathcal{A}_i \in \mathcal{M}_3, \exists \mathcal{A}_j \in \mathcal{M}_7$ such that $\mathcal{A}_i \subseteq \mathcal{A}_j$.
9. $\forall \mathcal{A}_i \in \mathcal{M}_4, \exists \mathcal{A}_j \in \mathcal{M}_7$ such that $\mathcal{A}_i \subseteq \mathcal{A}_j$.
10. $\forall \mathcal{A}_i \in \mathcal{M}_5, \exists \mathcal{A}_j \in \mathcal{M}_8$ such that $\mathcal{A}_i \subseteq \mathcal{A}_j$.

Proposition B.2 (Figure 5.3). *Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} = \{A\} \rangle$ be an inconsistent DL-Lite knowledge base. Let $\{\mathcal{M}_1, \dots, \mathcal{M}_8\}$ be the eight composite modifiers summarized in Table 5.1. Then:*

1. \mathcal{M}_6 and \mathcal{M}_8 are incomparable.
2. \mathcal{M}_2 and \mathcal{M}_6 are incomparable.
3. \mathcal{M}_3 and \mathcal{M}_6 are incomparable.
4. \mathcal{M}_4 and \mathcal{M}_6 are incomparable.
5. \mathcal{M}_2 and \mathcal{M}_8 are incomparable.
6. \mathcal{M}_3 and \mathcal{M}_8 are incomparable.

7. \mathcal{M}_4 and \mathcal{M}_8 are incomparable.

Example B.2 (Examples of Proposition B.2). The following examples show incomparability between MBox's

1. The MBox \mathcal{M}_6 and \mathcal{M}_8 are incomparable.

Let $\mathcal{T}=\{B \sqsubseteq \neg C, B \sqsubseteq A, C \sqsubseteq A, A \sqsubseteq \neg D, D \sqsubseteq E, E \sqsubseteq F\}$ and $\mathcal{M}=\{\{A(a), B(a), C(a), D(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$\mathcal{M}_1 = \circ_{incl}(\mathcal{M}) = \{\{A(a), C(a)\}, \{A(a), B(a)\}, \{D(a)\}\}$, and

$\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1) = \{\{A(a), C(a)\}, \{A(a), B(a)\}, \{D(a), E(a), F(a)\}\}$, and

$\mathcal{M}_6 = \circ_{card}(\mathcal{M}_5) = \{\{D(a), E(a), F(a)\}\}$,

$\circ_{cl}(\mathcal{M}) = \{\{A(a), B(a), C(a), D(a), E(a), F(a)\}\}$,

$\mathcal{M}_7 = \circ_{incl}(\circ_{cl}(\mathcal{M})) = \{\{A(a), C(a), E(a), F(a)\}, \{A(a), B(a), E(a), F(a)\}, \{D(a), E(a), F(a)\}\}$,

and

$\mathcal{M}_8 = \circ_{incl}(\mathcal{M}_7) = \{\{A(a), C(a), E(a), F(a)\}, \{A(a), B(a), E(a), F(a)\}\}$

One can check that \mathcal{M}_6 and \mathcal{M}_8 are incomparable.

2. The MBox \mathcal{M}_2 , \mathcal{M}_3 and \mathcal{M}_4 are incomparable with \mathcal{M}_6 .

Let $\mathcal{T}=\{A \sqsubseteq \neg B, C \sqsubseteq A, B \sqsubseteq D, D \sqsubseteq F\}$ and $\mathcal{M}=\{\{A(a), C(a), B(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$\mathcal{M}_1 = \circ_{incl}(\mathcal{M}) = \{\{A(a), C(a)\}, \{B(a)\}\}$,

$\mathcal{M}_2 = \circ_{card}(\mathcal{M}_1) = \{\{A(a), C(a)\}\}$,

$\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1) = \{\{A(a), C(a)\}, \{B(a), D(a), F(a)\}\}$,

$\mathcal{M}_6 = \circ_{card}(\mathcal{M}_5) = \{\{B(a), D(a), F(a)\}\}$,

One can check that \mathcal{M}_2 is incomparable with \mathcal{M}_6 .

We have also $\mathcal{M}_2 = \mathcal{M}_3 = \mathcal{M}_4 = \{\{A(a), C(a)\}\}$, So, we conclude that \mathcal{M}_3 and \mathcal{M}_4 are incomparable with \mathcal{M}_6 .

3. The MBox \mathcal{M}_2 , \mathcal{M}_3 and \mathcal{M}_4 are incomparable with \mathcal{M}_8 .

Let $\mathcal{T}=\{B \sqsubseteq A, C \sqsubseteq A, A \sqsubseteq \neg D, E \sqsubseteq D, D \sqsubseteq F\}$ and $\mathcal{M}=\{\{A(a), D(a), E(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$\mathcal{M}_1 = \circ_{incl}(\mathcal{M}) = \{\{A(a)\}, \{D(a), E(a)\}\}$, and

$\mathcal{M}_2 = \circ_{card}(\mathcal{M}_1) = \{D(a), E(a)\}$, and

$\mathcal{M}_3 = \mathcal{M}_4 = \{D(a), E(a), F(a)\}$,

$\circ_{cl}(\mathcal{M}) = \{\{A(a), B(a), C(a), D(a), E(a), F(a)\}\}$,

$\mathcal{M}_7 = \circ_{incl}(\circ_{cl}(\mathcal{M})) = \{\{A(a), C(a), B(a), F(a)\}, \{D(a), E(a), F(a)\}\}$, and

$\mathcal{M}_8 = \circ_{incl}(\mathcal{M}_7) = \{\{A(a), C(a), B(a), F(a)\}\}$,

One can check that \mathcal{M}_2 , \mathcal{M}_3 and \mathcal{M}_4 are incomparable with \mathcal{M}_8 .

Comparative studies

Composite modifiers + universal inference

Proposition B.3 (Figure 5.4). Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} = \{A\} \rangle$ be an inconsistent DL-Lite knowledge base. Let $\mathcal{M}_1, \dots, \mathcal{M}_8$ be the eight MBox's given in Table 5.1. Let q be a boolean query. Then:

1. q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_1 \rangle$ iff q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$.

2. q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_2 \rangle$ iff q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_3 \rangle$.

Proof. Item 1 and 2 follow from item 1 of Lemma 5.3 and the facts that $\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1)$ and $\mathcal{M}_3 = \circ_{cl}(\mathcal{M}_2)$. \square

Proposition B.4 (Figure 5.4). *Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} = \{\mathcal{A}\} \rangle$ be an inconsistent DL-Lite knowledge base. Let $\mathcal{M}_1, \dots, \mathcal{M}_8$ be the eight MBox's given in Table 5.1. Let q be a boolean query. Then:*

1. *If q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_1 \rangle$ (or $\langle \mathcal{T}, \mathcal{M}_5 \rangle$) then q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_2 \rangle$.*
2. *If q is universal conclusion of $\langle \mathcal{T}, \mathcal{M}_3 \rangle$ (or $\langle \mathcal{T}, \mathcal{M}_2 \rangle$) then q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_4 \rangle$.*
3. *If q is universal conclusion of $\langle \mathcal{T}, \mathcal{M}_1 \rangle$ (or $\langle \mathcal{T}, \mathcal{M}_5 \rangle$) then q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_6 \rangle$.*
4. *If q is universal conclusion of $\langle \mathcal{T}, \mathcal{M}_7 \rangle$ then q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_8 \rangle$.*
5. *If q is universal conclusion of $\langle \mathcal{T}, \mathcal{M}_1 \rangle$ (or $\langle \mathcal{T}, \mathcal{M}_5 \rangle$) then q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_7 \rangle$.*

Proof. For Items 1, 2, 3 and 4, we have $\mathcal{M}_2 \subseteq \mathcal{M}_1$, $\mathcal{M}_4 \subseteq \mathcal{M}_3$, $\mathcal{M}_6 \subseteq \mathcal{M}_5$ and $\mathcal{M}_8 \subseteq \mathcal{M}_7$. Then following Item 1 of Lemma 5.2, we have if $\langle \mathcal{T}, \mathcal{M}_1 \rangle \models_{\forall} q$ then $\langle \mathcal{T}, \mathcal{M}_2 \rangle \models_{\forall} q$. Similarly for $\mathcal{M}_4 \subseteq \mathcal{M}_3$, $\mathcal{M}_6 \subseteq \mathcal{M}_5$ and $\mathcal{M}_8 \subseteq \mathcal{M}_7$.

Finally, for item 5 recall first that $\langle \mathcal{M}_5, \forall \rangle \equiv \langle \mathcal{M}_1, \forall \rangle$ and $\forall A \in \mathcal{M}_5, \exists B \in \mathcal{M}_7$ such that $A \subseteq B$. Now let us show that $\forall B \in \mathcal{M}_7, \exists A \in \mathcal{M}_5$ such that $A \subseteq B$. Let $B \in \mathcal{M}_7 = \circ_{incl}(\circ_{cl}(\mathcal{M}))$. This means that $B \subseteq \circ_{cl}(\mathcal{M})$ and B is a maximally consistent subset. Let $C \in \circ_{incl}(\mathcal{M})$. This means that $C \subseteq \mathcal{M} \subseteq \circ_{cl}(\mathcal{M})$. Since C is also a maximally consistent subset then $C \subseteq B$. Now, recall that B is a closed set of assertion, then $A = Cl(C) \subseteq B$. Therefore we conclude that if a conclusion holds from \mathcal{M}_5 , then it holds from \mathcal{M}_7 . \square

Example B.3 (Counter-examples of Proposition B.4 of Figure 5.4). The following counter-examples illustrate the difference between inference relations.

1. There exists a DL-Lite knowledge base, and a boolean query q such that q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_2 \rangle$, but q is not a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_1 \rangle$:
 Let us consider $\mathcal{T} = \{A \sqsubseteq B, B \sqsubseteq \neg C\}$ and $\mathcal{M} = \{\{A(a), B(a), C(a)\}\}$.
 It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:
 $\mathcal{M}_1 = \circ_{incl}(\mathcal{M}) = \{\{A(a), B(a)\}, \{C(a)\}\}$, and
 $\mathcal{M}_2 = \circ_{card}(\mathcal{M}_1) = \{\{A(a), B(a)\}\}$.
 Let $q \leftarrow A(a)$ be a query. One can check that:
 $\langle \mathcal{M}_2, \forall \rangle \models q$ but
 $\langle \mathcal{M}_1, \forall \rangle \not\models q$, since $\langle \mathcal{T}, \{C(a)\} \rangle \not\models q$.
2. There exists a DL-Lite knowledge base, and a boolean query q such that q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_4 \rangle$, but q is not a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_2 \rangle$:
 Let us consider $\mathcal{T} = \{A \sqsubseteq \neg B, A \sqsubseteq F\}$ and $\mathcal{M} = \{\{A(a), B(a)\}\}$.
 It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:
 $\mathcal{M}_1 = \mathcal{M}_2 = \{\{A(a)\}, \{B(a)\}\}$,
 $\mathcal{M}_3 = \{\{A(a), F(a)\}, \{B(a)\}\}$, and
 $\mathcal{M}_4 = \{\{A(a), F(a)\}\}$.
 Let $q \leftarrow F(a)$ be a query. One can check that:
 $\langle \mathcal{M}_4, \forall \rangle \models q$ but
 $\langle \mathcal{M}_2, \forall \rangle \not\models q$, since $\langle \mathcal{T}, \{B(a)\} \rangle \not\models q$.

3. There exists a *DL-Lite* knowledge base, and a boolean query q such that q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_6 \rangle$, but q is not a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$:

Let us consider $\mathcal{T} = \{B \sqsubseteq C, C \sqsubseteq \neg D\}$ and $\mathcal{M} = \{\{B(a), D(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$$\mathcal{M}_1 = \{\{B(a)\}, \{D(a)\}\},$$

$$\mathcal{M}_5 = \{\{B(a), C(a)\}, \{D(a)\}\}, \text{ and}$$

$$\mathcal{M}_6 = \{\{B(a), C(a)\}\}.$$

Let $q \leftarrow C(a)$ be a query. One can check that:

$$\langle \mathcal{M}_6, \forall \rangle \models q \text{ but}$$

$$\langle \mathcal{M}_5, \forall \rangle \not\models q, \text{ since } \langle \mathcal{T}, \{D(a)\} \rangle \not\models q$$

4. There exists a *DL-Lite* knowledge base, and a boolean query q such that q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_8 \rangle$, but q is not a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_7 \rangle$:

Let us consider $\mathcal{T} = \{A \sqsubseteq B, B \sqsubseteq \neg C, C \sqsubseteq D, D \sqsubseteq F\}$ and $\mathcal{M} = \{\{A(a), C(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$$\circ_{cl}(\mathcal{M}) = \{A(a), C(a), B(a), D(a), F(a)\},$$

$$\mathcal{M}_7 = \{\{A(a), B(a), D(a), F(a)\}, \{C(a), D(a), F(a)\}\}, \text{ and}$$

$$\mathcal{M}_8 = \{\{A(a), B(a), D(a), F(a)\}\}.$$

Let $q \leftarrow A(a)$ be a query. One can check that:

$$\langle \mathcal{M}_8, \forall \rangle \models q, \text{ but}$$

$$\langle \mathcal{M}_7, \forall \rangle \not\models q, \text{ since } \langle \mathcal{T}, \{C(a), D(a), F(a)\} \rangle \not\models q.$$

5. There exists a *DL-Lite* knowledge base, and a boolean query q such that q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_7 \rangle$, but q is not a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$:

Let $\mathcal{T} = \{A \sqsubseteq \neg B, B \sqsubseteq D\}$ and $\mathcal{M} = \{\{A(a), B(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$$\mathcal{M}_1 = \{\{A(a)\}, \{B(a)\}\},$$

$$\circ_{cl}(\mathcal{M}) = \{\{A(a), B(a), D(a)\}\}, \text{ and}$$

$$\mathcal{M}_7 = \{\{A(a), D(a)\}, \{B(a), D(a)\}\}.$$

Let $q \leftarrow D(a)$ be a query. One can check that:

$$\langle \mathcal{M}_7, \forall \rangle \models q \text{ but}$$

$$\langle \mathcal{M}_1, \forall \rangle \not\models q, \text{ since } \langle \mathcal{T}, \{A(a)\} \rangle.$$

Proposition B.5 (Figure 5.4). *Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} = \{A\} \rangle$ be an inconsistent *DL-Lite* knowledge base. Let $\mathcal{M}_1, \dots, \mathcal{M}_8$ be the eight *MBox*'s given in Table 5.1. Then:*

1. *The universal inference from \mathcal{M}_6 is incomparable with the one obtained from \mathcal{M}_7 .*
2. *The universal inference from \mathcal{M}_6 is incomparable with the one obtained from \mathcal{M}_8 .*
3. *The universal inference from \mathcal{M}_2 (resp. $\mathcal{M}_3, \mathcal{M}_4$) is incomparable with the one obtained from \mathcal{M}_6 .*
4. *The universal inference from \mathcal{M}_2 (resp. $\mathcal{M}_3, \mathcal{M}_4$) is incomparable with the one obtained from \mathcal{M}_7 .*
5. *The universal inference from \mathcal{M}_2 (resp. $\mathcal{M}_3, \mathcal{M}_4$) is incomparable with the one obtained from \mathcal{M}_8 .*

Example B.4 (Examples of Figure 5.4). The following examples show incomparability between universal inference.

-
1. The universal inference from \mathcal{M}_6 is incomparable with the one obtained from \mathcal{M}_7 .
Let $\mathcal{T}=\{C \sqsubseteq F, F \sqsubseteq A, A \sqsubseteq \neg B, B \sqsubseteq D\}$ and $\mathcal{M}=\{\{C(a), B(a)\}\}$.
It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:
 $\mathcal{M}_1 = \circ_{incl}(\mathcal{M})=\{\{C(a)\}, \{B(a)\}\}$, and
 $\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1)=\{\{A(a), C(a), F(a)\}, \{B(a), D(a)\}\}$, and
 $\mathcal{M}_6 = \circ_{card}(\mathcal{M}_5)=\{\{A(a), C(a), F(a)\}\}$,
 $\circ_{cl}(\mathcal{M})=\{\{A(a), C(a), F(a), B(a), D(a)\}\}$,
 $\mathcal{M}_7 = \circ_{incl}(\circ_{cl}(\mathcal{M}))=\{\{A(a), C(a), F(a), D(a)\}, \{D(a), B(a)\}\}$,
Let $q_1 \leftarrow F(a)$ and $q_2 \leftarrow D(a)$ be two queries. One can check that:
 $\langle \mathcal{M}_7, \forall \rangle \models q_2$ but $\langle \mathcal{M}_6, \forall \rangle \not\models q_2$ while $\langle \mathcal{M}_6, \forall \rangle \models q_1$ but $\langle \mathcal{M}_7, \forall \rangle \not\models q_1$.
2. The universal inference from \mathcal{M}_6 is incomparable with the one obtained from \mathcal{M}_8 .
Let $\mathcal{T}=\{B \sqsubseteq \neg C, B \sqsubseteq A, C \sqsubseteq A, A \sqsubseteq \neg D, D \sqsubseteq E, E \sqsubseteq F\}$ and $\mathcal{M}=\{\{A(a), B(a), C(a), D(a)\}\}$.
It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:
 $\mathcal{M}_1 = \circ_{incl}(\mathcal{M})=\{\{A(a), C(a)\}, \{A(a), B(a)\}, \{D(a)\}\}$, and
 $\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1)=\{\{A(a), C(a)\}, \{A(a), B(a)\}, \{D(a), E(a), F(a)\}\}$, and
 $\mathcal{M}_6 = \circ_{card}(\mathcal{M}_5)=\{\{D(a), E(a), F(a)\}\}$,
 $\circ_{cl}(\mathcal{M})=\{\{A(a), B(a), C(a), D(a), E(a), F(a)\}\}$,
 $\mathcal{M}_7 = \circ_{incl}(\circ_{cl}(\mathcal{M}))=\{\{A(a), C(a), E(a), F(a)\}, \{A(a), B(a), E(a), F(a)\}, \{D(a), E(a), F(a)\}\}$,
and
 $\mathcal{M}_8 = \circ_{card}(\mathcal{M}_7)=\{\{A(a), C(a), E(a), F(a)\}, \{A(a), B(a), E(a), F(a)\}\}$
Let $q_1 \leftarrow D(a)$ and $q_2 \leftarrow A(a)$ be two queries. One can check that:
 $\langle \mathcal{M}_8, \forall \rangle \models q_2$ but $\langle \mathcal{M}_6, \forall \rangle \not\models q_2$ while $\langle \mathcal{M}_6, \forall \rangle \models q_1$ but $\langle \mathcal{M}_8, \forall \rangle \not\models q_1$.
3. The universal inference from \mathcal{M}_2 (resp. \mathcal{M}_3 and \mathcal{M}_4) is incomparable with the one obtained from \mathcal{M}_6 .
Let $\mathcal{T}=\{A \sqsubseteq \neg B, C \sqsubseteq A, B \sqsubseteq D, D \sqsubseteq F\}$ and $\mathcal{M}=\{\{A(a), C(a), B(a)\}\}$.
It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:
 $\mathcal{M}_1 = \circ_{incl}(\mathcal{M})=\{\{A(a), C(a)\}, \{B(a)\}\}$,
 $\mathcal{M}_2 = \circ_{card}(\mathcal{M}_1)=\{\{A(a), C(a)\}\}$,
 $\mathcal{M}_4 = \{\{A(a), C(a)\}\}$,
 $\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1)=\{\{A(a), C(a)\}, \{B(a), D(a), F(a)\}\}$,
 $\mathcal{M}_6 = \circ_{card}(\mathcal{M}_5)=\{\{B(a), D(a), F(a)\}\}$,
Let $q_1 \leftarrow A(a)$ and $q_2 \leftarrow B(a)$ be two queries. One can check that:
 $\langle \mathcal{M}_2, \forall \rangle \models q_1$ but $\langle \mathcal{M}_6, \forall \rangle \not\models q_1$ while $\langle \mathcal{M}_6, \forall \rangle \models q_2$ but $\langle \mathcal{M}_2, \forall \rangle \not\models q_2$. Similarly for \mathcal{M}_4
4. The universal inference from \mathcal{M}_2 (resp. \mathcal{M}_3 and \mathcal{M}_4) is incomparable with the one obtained from \mathcal{M}_7 .
Let $\mathcal{T}=\{A \sqsubseteq \neg B, C \sqsubseteq A, B \sqsubseteq D, D \sqsubseteq F\}$ and $\mathcal{M}=\{\{A(a), C(a), B(a)\}\}$.
It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:
 $\mathcal{M}_1 = \circ_{incl}(\mathcal{M})=\{\{A(a), C(a)\}, \{B(a)\}\}$,
 $\mathcal{M}_2 = \circ_{card}(\mathcal{M}_1)=\{\{A(a), C(a)\}\}$,
 $\mathcal{M}_4 = \{\{A(a), C(a)\}\}$,
 $\circ_{cl}(\mathcal{M})=\{\{A(a), C(a), B(a), D(a), F(a)\}\}$,
 $\mathcal{M}_7 = \circ_{incl}(\circ_{cl}(\mathcal{M}))=\{\{A(a), C(a), D(a), F(a)\}, \{B(a), D(a), F(a)\}\}$,
Let $q_1 \leftarrow A(a)$ and $q_2 \leftarrow D(a)$ be two queries. One can check that:
 $\langle \mathcal{M}_2, \forall \rangle \models q_1$ but $\langle \mathcal{M}_7, \forall \rangle \not\models q_1$ while $\langle \mathcal{M}_7, \forall \rangle \models q_2$ but $\langle \mathcal{M}_2, \forall \rangle \not\models q_2$. Similarly for \mathcal{M}_4 .

5. The universal inference from \mathcal{M}_2 (*resp.* \mathcal{M}_3 and \mathcal{M}_4) is incomparable with the one obtained from \mathcal{M}_8 .

Let $\mathcal{T} = \{B \sqsubseteq A, C \sqsubseteq A, A \sqsubseteq \neg D, E \sqsubseteq D, D \sqsubseteq F\}$ and $\mathcal{M} = \{\{A(a), D(a), E(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$\mathcal{M}_1 = \circ_{incl}(\mathcal{M}) = \{\{A(a)\}, \{D(a), E(a)\}\}$, and

$\mathcal{M}_2 = \circ_{carad}(\mathcal{M}_1) = \{D(a), E(a)\}$, and

$\mathcal{M}_4 = \{\{D(a), E(a), F(a)\}\}$,

$\circ_{cl}(\mathcal{M}) = \{\{A(a), B(a), C(a), D(a), E(a), F(a)\}\}$,

$\mathcal{M}_7 = \circ_{incl}(\circ_{cl}(\mathcal{M})) = \{\{A(a), C(a), B(a), F(a)\}, \{D(a), E(a), F(a)\}\}$, and

$\mathcal{M}_8 = \circ_{incl}(\mathcal{M}_7) = \{\{A(a), C(a), B(a), F(a)\}\}$,

Let $q_1 \leftarrow D(a)$ and $q_2 \leftarrow A(a)$ be two queries. One can check that:

$\langle \mathcal{M}_2, \forall \rangle \models q_1$ but $\langle \mathcal{M}_8, \forall \rangle \not\models q_1$ while $\langle \mathcal{M}_8, \forall \rangle \models q_2$ but $\langle \mathcal{M}_2, \forall \rangle \not\models q_2$. Similarly for \mathcal{M}_4

Composite modifiers+safe inference

Proposition B.6 (Figure 5.5). *Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} = \{A\} \rangle$ be an inconsistent DL-Lite knowledge base. Let $\mathcal{M}_1, \dots, \mathcal{M}_8$ be the eight MBox's given in Table 5.1. Let q be a boolean query. Then:*

1. *If q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_1 \rangle$ then q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_2 \rangle$.*
2. *If q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_1 \rangle$ then q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$.*
3. *If q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_2 \rangle$ then q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_3 \rangle$.*
4. *If q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_3 \rangle$ then q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_4 \rangle$.*
5. *If q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$ then q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_3 \rangle$.*
6. *if q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$ then q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_6 \rangle$.*
7. *If q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$ then q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_7 \rangle$.*
8. *If q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_7 \rangle$ then q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_8 \rangle$.*

Proof. The proof is as follows:

1. For items 1, we have $\mathcal{M}_2 \subseteq \mathcal{M}_1$, then following Item 2 of Lemma 5.2, if $\langle \mathcal{M}_1, \cap \rangle$ implies a query q then $\langle \mathcal{M}_2, \cap \rangle$ implies it also. The proof follow similarly for Items 4,5, 6 and 8 since $\mathcal{M}_4 \subseteq \mathcal{M}_3$, $\mathcal{M}_3 \subseteq \mathcal{M}_5$, $\mathcal{M}_6 \subseteq \mathcal{M}_5$, and $\mathcal{M}_8 \subseteq \mathcal{M}_7$.
2. For items 2 and 3, we have $\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1)$ and $\mathcal{M}_3 = \circ_{cl}(\mathcal{M}_2)$. Then following Item 3 of Lemma 5.3, if a query holds in $\langle \mathcal{M}, \cap \rangle$ then it also holds in $\langle \circ_{cl}(\mathcal{M}), \cap \rangle$.
3. For item 7, we have $\forall A \in \mathcal{M}_5, \exists B \in \mathcal{M}_7$ such that $A \subseteq B$. Let $A(a) \in \bigcap_{A_i \in \mathcal{M}_5} A_i$. Then one can check that there is no conflict \mathcal{C} in $\langle \mathcal{T}, Cl(\mathcal{M}) \rangle$ such that $A(a) \in \mathcal{C}$. Indeed, assume that such conflict exists. Then this means that there exists $B(a) \in Cl(\mathcal{M})$ where $\langle \mathcal{T}, \{(A(a), B(a))\} \rangle$ is conflicting. Two options:
 - i) $B(a) \in \mathcal{M}$. This means that there exists a maximally consistent subset X of \mathcal{M} with $B(a) \in X$. Since $B(a)$ is conflicting with $A(a)$, with respect to \mathcal{T} . Then $A(a)$ neither belongs to X nor to $Cl(X)$. This contradict the fact that $A(a) \in \bigcap_{A_i \in \mathcal{M}_5} A_i$.
 - ii) $B(a) \notin \mathcal{M}$. Let $Y \subseteq \mathcal{M}$ such that $\langle \mathcal{T}, Y \rangle \models B(a)$. Then clearly $\langle \mathcal{T}, Y \cup \{A(a)\} \rangle$ is inconsistent. Hence, there exists $D(a) \in \mathcal{M}$ such that $\langle \mathcal{T}, \{(D(a), A(a))\} \rangle$ is conflicting and

$D(a) \in Y$. This comes down to item (i). Now, since there is no conflict in $Cl(\mathcal{M})$ containing $A(a)$, then $A(a)$ belong to all maximally consistent subsets of $Cl(\mathcal{M})$, hence $A(a)$ belongs to $\bigcap_{\mathcal{A}_j \in \mathcal{M}_7} \mathcal{A}_j$. Therefore if a q holds in $\langle \mathcal{M}_5, \cap \rangle$, then it holds that $\langle \mathcal{M}_5, \cap \rangle$.

□

Example B.5 (Counter-examples of Figure 5.5). The following counter-examples illustrate the difference between inference relations given in Figure 5.5.

1. There exists a *DL-Lite* knowledge base, and a boolean query q such that q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_2 \rangle$, but q is not a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_1 \rangle$:
 Let us consider $\mathcal{T} = \{A \sqsubseteq B, B \sqsubseteq \neg C\}$ and $\mathcal{M} = \{\{C(a), A(a), B(a)\}\}$.
 It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:
 $\mathcal{M}_1 = \{\{C(a)\}, \{A(a), B(a)\}\}$, and
 $\mathcal{M}_2 = \{\{A(a), B(a)\}\}$.
 Let $q \leftarrow A(a)$ be a query. One can check that:
 $\mathcal{M}_2 \models_{\cap} q$, since $\bigcap_{\mathcal{A}_i \in \mathcal{M}_2} \mathcal{A}_i = \{A(a), B(a)\}$ but
 $\mathcal{M}_1 \not\models_{\cap} q$.
2. There exists a *DL-Lite* knowledge base, and a boolean query q such that q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$, but q is not a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_1 \rangle$:
 Let us consider $\mathcal{T} = \{B \sqsubseteq D, B \sqsubseteq \neg C, C \sqsubseteq D\}$ and $\mathcal{M} = \{\{C(a), B(a)\}\}$.
 It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:
 $\mathcal{M}_1 = \{\{C(a)\}, \{B(a)\}\}$, and
 $\mathcal{M}_5 = \{\{B(a), D(a)\}, \{C(a), D(a)\}\}$.
 Let $q \leftarrow D(a)$ be a query. One can check that :
 $\mathcal{M}_5 \models_{\cap} q$ since $\bigcap_{\mathcal{A}_i \in \mathcal{M}_5} \mathcal{A}_i = \{D(a)\}$, but
 $\mathcal{M}_1 \not\models_{\cap} q$.
3. There exists a *DL-Lite* knowledge base, and a boolean query q such that q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_3 \rangle$, but q is not a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_2 \rangle$:
 Let us consider $\mathcal{T} = \{B \sqsubseteq \neg C, C \sqsubseteq A, B \sqsubseteq A\}$ and $\mathcal{M} = \{\{C(a), B(a)\}\}$.
 It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:
 $\mathcal{M}_1 = \mathcal{M}_2 = \{\{C(a)\}, \{B(a)\}\}$, and
 $\mathcal{M}_3 = \{\{C(a), A(a)\}, \{B(a), A(a)\}\}$.
 Let $q \leftarrow A(a)$ be a query. One can check that:
 $\mathcal{M}_3 \models_{\cap} q$, but
 $\mathcal{M}_2 \not\models_{\cap} q$.
4. There exists a *DL-Lite* knowledge base, and a boolean query q such that q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_4 \rangle$, but q is not a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_3 \rangle$:
 Let us consider $\mathcal{T} = \{A \sqsubseteq B, B \sqsubseteq \neg D\}$ and $\mathcal{M} = \{\{A(a), D(a)\}\}$.
 It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:
 $\mathcal{M}_1 = \mathcal{M}_2 = \{\{A(a)\}, \{D(a)\}\}$,
 $\mathcal{M}_3 = \{\{A(a), B(a)\}, \{D(a)\}\}$, and
 $\mathcal{M}_4 = \{\{A(a), B(a)\}\}$.
 Let $q \leftarrow A(a)$ be a query. One can check that
 $\mathcal{M}_4 \models_{\cap} q$ but

$\mathcal{M}_3 \not\models_{\cap} q$.

5. There exists a *DL-Lite* knowledge base, and a boolean query q such that q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_3 \rangle$, but q is not a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$:

Let us consider $\mathcal{T} = \{A \sqsubseteq B, B \sqsubseteq \neg D\}$ and $\mathcal{M} = \{\{A(a), D(a), B(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$\mathcal{M}_1 = \{\{A(a), B(a)\}, \{D(a)\}\}$,

$\mathcal{M}_2 = \mathcal{M}_3 = \{\{A(a), B(a)\}\}$, and

$\mathcal{M}_5 = \{\{A(a), B(a)\}, \{D(a)\}\}$.

Let $q \leftarrow A(a)$ be a query. One can check that

$\mathcal{M}_3 \models_{\cap} q$ but

$\mathcal{M}_5 \not\models_{\cap} q$.

6. There exists a *DL-Lite* knowledge base, and a boolean query q such that q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_6 \rangle$, but q is not a safe conclusion of \mathcal{M}_5 :

Let us consider $\mathcal{T} = \{B \sqsubseteq C, C \sqsubseteq \neg D\}$ and $\mathcal{M} = \{\{B(a), D(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$\mathcal{M}_1 = \{\{B(a)\}, \{D(a)\}\}$,

$\mathcal{M}_5 = \{\{B(a), C(a)\}, \{D(a)\}\}$, and

$\mathcal{M}_6 = \{\{B(a), C(a)\}\}$.

Let $q \leftarrow B(a)$ be a query. One can check that

$\mathcal{M}_6 \models_{\cap} q$ but

$\mathcal{M}_5 \not\models_{\cap} q$.

7. There exists a *DL-Lite* knowledge base, and a boolean query q such that q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_7 \rangle$, but q is not a safe conclusion of \mathcal{M}_5 :

Let $\mathcal{T} = \{A \sqsubseteq \neg B, B \sqsubseteq D\}$ and $\mathcal{M} = \{\{A(a), B(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$\mathcal{M}_1 = \{\{A(a)\}, \{B(a)\}\}$,

$\mathcal{M}_5 = \{\{A(a)\}, \{B(a), D(a)\}\}$,

$\circ_{cl}(\mathcal{M}) = \{\{A(a), B(a), D(a)\}\}$, and

$\mathcal{M}_7 = \{\{A(a), D(a)\}, \{B(a), D(a)\}\}$.

Let $q \leftarrow D(a)$ be a query. One can deduce that:

$\mathcal{M}_7 \models_{\cap} q$ but

$\mathcal{M}_5 \not\models_{\cap} q$.

8. There exists a *DL-Lite* knowledge base, and a boolean query q such that q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_8 \rangle$, but q is not a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_7 \rangle$:

Let us consider $\mathcal{T} = \{A \sqsubseteq B, B \sqsubseteq \neg C, C \sqsubseteq D\}$ and $\mathcal{M} = \{\{A(a), C(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$\circ_{cl}(\mathcal{M}) = \{A(a), C(a), B(a), D(a)\}$,

$\mathcal{M}_7 = \{\{A(a), B(a), D(a)\}, \{C(a), D(a)\}\}$, and

$\mathcal{M}_8 = \{\{A(a), B(a), D(a)\}\}$.

Let $q \leftarrow A(a)$ be a boolean query. One can deduce that:

$\mathcal{M}_8 \models_{\cap} q$, but

$\mathcal{M}_7 \not\models_{\cap} q$.

Proposition B.7 (Figure 5.5). Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} = \{A\} \rangle$ be an inconsistent DL-Lite knowledge base. Let $\mathcal{M}_1, \dots, \mathcal{M}_8$ be the eight MBox's given in Table 5.1. Let q be a boolean query. Then:

1. The safe inference from \mathcal{M}_2 is incomparable with the one obtained from \mathcal{M}_6 .
2. The safe inference from \mathcal{M}_3 is incomparable with the one obtained from \mathcal{M}_6 .
3. The safe inference from \mathcal{M}_6 is incomparable with the one obtained from \mathcal{M}_7 .
4. The safe inference from \mathcal{M}_6 is incomparable with the one obtained from \mathcal{M}_8 .
5. The safe inference from \mathcal{M}_2 is incomparable with the one obtained from \mathcal{M}_5 .

Example B.6 (Examples of Figure 5.5). The following examples show incomparability between MBox's

1. The safe inference from \mathcal{M}_6 is incomparable with the one obtained from \mathcal{M}_7 .

Let $\mathcal{T} = \{C \sqsubseteq F, F \sqsubseteq A, A \sqsubseteq \neg B, B \sqsubseteq D\}$ and $\mathcal{M} = \{\{C(a), B(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$\mathcal{M}_1 = \circ_{incl}(\mathcal{M}) = \{\{C(a)\}, \{B(a)\}\}$, and

$\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1) = \{\{A(a), C(a), F(a)\}, \{B(a), D(a)\}\}$, and

$\mathcal{M}_6 = \circ_{card}(\mathcal{M}_5) = \{\{A(a), C(a), F(a)\}\}$,

$\circ_{cl}(\mathcal{M}) = \{\{A(a), C(a), F(a), B(a), D(a)\}\}$,

$\mathcal{M}_7 = \circ_{incl}(\circ_{cl}(\mathcal{M})) = \{\{A(a), C(a), F(a), D(a)\}, \{D(a), B(a)\}\}$,

Let $q_1 \leftarrow F(a)$ and $q_2 \leftarrow D(a)$ be two queries. One can check that:

$\langle \mathcal{M}_7, \cap \rangle \models q_2$ but $\langle \mathcal{M}_6, \cap \rangle \not\models q_2$ while $\langle \mathcal{M}_6, \cap \rangle \models q_1$ but $\langle \mathcal{M}_7, \cap \rangle \not\models q_1$.

2. The safe inference from \mathcal{M}_6 is incomparable with the one obtained from \mathcal{M}_8 .

Let $\mathcal{T} = \{B \sqsubseteq \neg C, B \sqsubseteq A, C \sqsubseteq A, A \sqsubseteq \neg D, D \sqsubseteq E, E \sqsubseteq F\}$ and $\mathcal{M} = \{\{A(a), B(a), C(a), D(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$\mathcal{M}_1 = \circ_{incl}(\mathcal{M}) = \{\{A(a), C(a)\}, \{A(a), B(a)\}, \{D(a)\}\}$,

$\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1) = \{\{A(a), C(a)\}, \{A(a), B(a)\}, \{D(a), E(a), F(a)\}\}$,

$\mathcal{M}_6 = \circ_{card}(\mathcal{M}_5) = \{\{D(a), E(a), F(a)\}\}$,

$\circ_{cl}(\mathcal{M}) = \{\{A(a), B(a), C(a), D(a), E(a), F(a)\}\}$,

$\mathcal{M}_7 = \circ_{incl}(\circ_{cl}(\mathcal{M})) = \{\{A(a), C(a), E(a), F(a)\}, \{A(a), B(a), E(a), F(a)\}, \{D(a), E(a), F(a)\}\}$,

and

$\mathcal{M}_8 = \circ_{incl}(\mathcal{M}_7) = \{\{A(a), C(a), E(a), F(a)\}, \{A(a), B(a), E(a), F(a)\}\}$

Let $q_1 \leftarrow D(a)$ and $q_2 \leftarrow A(a)$ be two queries. One can check that:

$\langle \mathcal{M}_8, \cap \rangle \models q_2$ but $\langle \mathcal{M}_6, \cap \rangle \not\models q_2$ while $\langle \mathcal{M}_6, \cap \rangle \models q_1$ but $\langle \mathcal{M}_8, \cap \rangle \not\models q_1$.

3. The safe inference from \mathcal{M}_2 is incomparable with the one obtained from \mathcal{M}_6 .

Let $\mathcal{T} = \{A \sqsubseteq \neg B, C \sqsubseteq A, B \sqsubseteq D, D \sqsubseteq F\}$ and $\mathcal{M} = \{\{A(a), C(a), B(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$\mathcal{M}_1 = \circ_{incl}(\mathcal{M}) = \{\{A(a), C(a)\}, \{B(a)\}\}$,

$\mathcal{M}_2 = \circ_{card}(\mathcal{M}_1) = \{\{A(a), C(a)\}\}$,

$\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1) = \{\{A(a), C(a)\}, \{B(a), D(a), F(a)\}\}$,

$\mathcal{M}_6 = \circ_{card}(\mathcal{M}_5) = \{\{B(a), D(a), F(a)\}\}$,

Let $q_1 \leftarrow A(a)$ and $q_2 \leftarrow B(a)$ be two queries. One can check that:

$\langle \mathcal{M}_2, \cap \rangle \models q_1$ but $\langle \mathcal{M}_6, \cap \rangle \not\models q_1$ while $\langle \mathcal{M}_6, \cap \rangle \models q_2$ but $\langle \mathcal{M}_2, \cap \rangle \not\models q_2$.

4. The safe inference from \mathcal{M}_2 is incomparable with the one obtained from \mathcal{M}_5 .

Let $\mathcal{T} = \{A \sqsubseteq B, C \sqsubseteq B, A \sqsubseteq \neg C, D \sqsubseteq C\}$ and $\mathcal{M} = \{\{A(a), C(a), D(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$\mathcal{M}_1 = \circ_{incl}(\mathcal{M}) = \{\{A(a)\}, \{C(a), D(a)\}\}$,

$\mathcal{M}_2 = \circ_{carad}(\mathcal{M}_1) = \{\{C(a), D(a)\}\}$, and

$\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1) = \{\{A(a), B(a)\}, \{B(a), D(a), C(a)\}\}$,

Let $q_1 \leftarrow D(a)$ and $q_2 \leftarrow B(a)$ be two queries. One can check that:

$\langle \mathcal{M}_2, \cap \rangle \models q_1$ but $\langle \mathcal{M}_5, \cap \rangle \not\models q_1$ while $\langle \mathcal{M}_5, \cap \rangle \models q_2$ but $\langle \mathcal{M}_2, \cap \rangle \not\models q_2$.

Composite modifiers+majority-based inference

Proposition B.8 (Figure 5.6). *Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} = \{A\} \rangle$ be an inconsistent DL-Lite knowledge base. Let $\mathcal{M}_1, \dots, \mathcal{M}_8$ be the eight MBox's given in Table 5.1. Let q be a boolean query. Then:*

- $\langle \mathcal{T}, \mathcal{M}_1 \rangle \models_{maj} q$ iff $\langle \mathcal{T}, \mathcal{M}_5 \rangle \models_{maj} q$.
- $\langle \mathcal{T}, \mathcal{M}_2 \rangle \models_{maj} q$ iff $\langle \mathcal{T}, \mathcal{M}_3 \rangle \models_{maj} q$.
- If $\langle \mathcal{T}, \mathcal{M}_5 \rangle \models_{maj} q$ then $\langle \mathcal{T}, \mathcal{M}_7 \rangle \models_{maj} q$.

Proof. The proof of items 1 and 2 follow immediately from the proof of item 2 of Lemma 5.3, since $\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1)$ and $\mathcal{M}_2 = \circ_{cl}(\mathcal{M}_3)$. For Item 3, we have $\forall \mathcal{A}_i \in \mathcal{M}_5, \exists \mathcal{A}_j \in \mathcal{M}_7$ such that $\mathcal{A}_i \subseteq \mathcal{A}_j$. From proof of item 5 of proposition B.4, we have $\forall \mathcal{A}_j \in \mathcal{M}_7, \exists \mathcal{A}_i \in \mathcal{M}_5$ such that $\mathcal{A}_i \subseteq \mathcal{A}_j$. We conclude that if a majority-based conclusion holds from \mathcal{M}_5 , it holds also from \mathcal{M}_7 . The converse does not hold. \square

Example B.7 (Counter-examples of Figure 5.6). The following counter-examples illustrate the difference between inference relations given in Figure 5.6.

1. There exists a DL-Lite knowledge base, and a query q such that q is a majority-based conclusion of $\langle \mathcal{T}, \mathcal{M}_7 \rangle$, but q is not a majority-based conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$:

Let $\mathcal{T} = \{A \sqsubseteq \neg B, B \sqsubseteq D\}$ and $\mathcal{M} = \{\{A(a), B(a)\}\}$.

It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:

$\circ_{cl}(\mathcal{M}) = \{\{A(a), B(a), D(a)\}\}$,

$\mathcal{M}_7 = \circ_{incl}(\circ_{cl}(\mathcal{M})) = \{\{A(a), D(a)\}, \{B(a), D(a)\}\}$,

$\mathcal{M}_1 = \circ_{incl}(\mathcal{M}) = \{\{A(a)\}, \{B(a)\}\}$, and

$\mathcal{M}_5 = \circ_{cl}(\mathcal{M}_1) = \{\{A(a)\}, \{B(a), D(a)\}\}$,

Let $q \leftarrow D(a)$ be a query. One can check that:

$\langle \mathcal{M}_7, maj \rangle \models q$ but $\langle \mathcal{M}_5, maj \rangle \not\models q$

Proposition B.9 (Figure 5.6). *Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} = \{A\} \rangle$ be an inconsistent DL-Lite knowledge base. Let $\mathcal{M}_1, \dots, \mathcal{M}_8$ be the eight MBox's given in Table 5.1. Let q be a boolean query. Then:*

- The majority-based inference from $\langle \mathcal{T}, \mathcal{M}_1 \rangle$ is incomparable with the one obtained from $\langle \mathcal{T}, \mathcal{M}_2 \rangle$.
- The majority-based inference from $\langle \mathcal{T}, \mathcal{M}_3 \rangle$ is incomparable with the one obtained from $\langle \mathcal{T}, \mathcal{M}_4 \rangle$.
- The majority-based inference from $\langle \mathcal{T}, \mathcal{M}_5 \rangle$ is incomparable with the one obtained from $\langle \mathcal{T}, \mathcal{M}_6 \rangle$.
- The majority-based inference from $\langle \mathcal{T}, \mathcal{M}_7 \rangle$ is incomparable with the one obtained from $\langle \mathcal{T}, \mathcal{M}_8 \rangle$.

Example B.8 (Examples of Figure 5.6). The following examples show incomparability between majority-based inferences.

1. The majority-based inference from $\langle \mathcal{T}, \mathcal{M}_1 \rangle$ is incomparable with the one obtained from $\langle \mathcal{T}, \mathcal{M}_2 \rangle$.
 Let $\mathcal{T} = \{B \sqsubseteq \neg C, B \sqsubseteq A, C \sqsubseteq A, A \sqsubseteq \neg D, D \sqsubseteq E, E \sqsubseteq F\}$ and $\mathcal{M} = \{\{A(a), B(a), C(a), D(a), E(a), F(a)\}\}$.
 It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:
 $\mathcal{M}_1 = \circ_{incl}(\mathcal{M}) = \{\{A(a), C(a)\}, \{A(a), B(a)\}, \{D(a), E(a), F(a)\}\}$, and
 $\mathcal{M}_2 = \circ_{card}(\mathcal{M}_1) = \{\{D(a), E(a), F(a)\}\}$
 Let $q_1 \leftarrow D(a)$ and $q_2 \leftarrow A(a)$ be two queries. One can check that:
 $\langle \mathcal{M}_1, maj \rangle \models q_2$ but $\langle \mathcal{M}_2, maj \rangle \not\models q_2$ while $\langle \mathcal{M}_2, maj \rangle \models q_1$ but $\langle \mathcal{M}_1, maj \rangle \not\models q_1$.
2. The majority-based inference from $\langle \mathcal{T}, \mathcal{M}_3 \rangle$ is incomparable with the one obtained from $\langle \mathcal{T}, \mathcal{M}_4 \rangle$.
 Let $\mathcal{T} = \{B \sqsubseteq \neg C, B \sqsubseteq A, C \sqsubseteq A, A \sqsubseteq \neg D, F \sqsubseteq D, D \sqsubseteq E\}$ and $\mathcal{M} = \{\{A(a), B(a), C(a), F(a), D(a)\}\}$.
 It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:
 $\mathcal{M}_1 = \mathcal{M}_2 = \{\{A(a), C(a)\}, \{A(a), B(a)\}, \{D(a), F(a)\}\}$,
 $\mathcal{M}_3 = \{\{A(a), C(a)\}, \{A(a), B(a)\}, \{D(a), F(a), E(a)\}\}$,
 $\mathcal{M}_4 = \{\{D(a), E(a), F(a)\}\}$
 Let $q_1 \leftarrow D(a)$ and $q_2 \leftarrow A(a)$ be two queries. One can check that:
 $\langle \mathcal{M}_3, maj \rangle \models q_2$ but $\langle \mathcal{M}_4, maj \rangle \not\models q_2$ while $\langle \mathcal{M}_4, maj \rangle \models q_1$ but $\langle \mathcal{M}_3, maj \rangle \not\models q_1$.
3. The majority-based inference from $\langle \mathcal{T}, \mathcal{M}_5 \rangle$ is incomparable with the one obtained from $\langle \mathcal{T}, \mathcal{M}_6 \rangle$.
 Let $\mathcal{T} = \{B \sqsubseteq \neg C, B \sqsubseteq A, C \sqsubseteq A, A \sqsubseteq \neg D, F \sqsubseteq D, D \sqsubseteq E\}$ and $\mathcal{M} = \{\{A(a), B(a), C(a), F(a), D(a), E(a)\}\}$.
 It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:
 $\mathcal{M}_1 = \mathcal{M}_5 = \{\{A(a), C(a)\}, \{A(a), B(a)\}, \{D(a), F(a), E(a)\}\}$,
 $\mathcal{M}_6 = \{\{D(a), E(a), F(a)\}\}$
 Let $q_1 \leftarrow D(a)$ and $q_2 \leftarrow A(a)$ be two queries. One can check that:
 $\langle \mathcal{M}_5, maj \rangle \models q_2$ but $\langle \mathcal{M}_6, maj \rangle \not\models q_2$ while $\langle \mathcal{M}_6, maj \rangle \models q_1$ but $\langle \mathcal{M}_5, maj \rangle \not\models q_1$.
4. The majority-based inference from $\langle \mathcal{T}, \mathcal{M}_7 \rangle$ is incomparable with the one obtained from $\langle \mathcal{T}, \mathcal{M}_8 \rangle$.
 Let $\mathcal{T} = \{B \sqsubseteq \neg C, B \sqsubseteq A, C \sqsubseteq A, A \sqsubseteq \neg D, F \sqsubseteq D, E \sqsubseteq D\}$ and $\mathcal{M} = \{\{A(a), F(a), E(a), B(a), C(a)\}\}$.
 It is easy to check that $\langle \mathcal{T}, \mathcal{M} \rangle$ is inconsistent. We have:
 $\circ_{cl}(\mathcal{M}) = \{\{A(a), C(a), B(a), D(a), F(a), E(a)\}\}$,
 $\mathcal{M}_7 = \{\{D(a), E(a), F(a)\}, \{A(a), B(a)\}, \{A(a), C(a)\}\}$, and
 $\mathcal{M}_8 = \{\{D(a), E(a), F(a)\}\}$, and
 Let $q_1 \leftarrow D(a)$ and $q_2 \leftarrow A(a)$ be two queries. One can check that:
 $\langle \mathcal{M}_7, maj \rangle \models q_2$ but $\langle \mathcal{M}_8, maj \rangle \not\models q_2$ while $\langle \mathcal{M}_8, maj \rangle \models q_1$ but $\langle \mathcal{M}_7, maj \rangle \not\models q_1$.

Global schema

Proposition B.10 (Figure 5.7). Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} = \{A\} \rangle$ be an inconsistent DL-Lite knowledge base. Let $\mathcal{M}_1, \dots, \mathcal{M}_8$ be the eight MBox's given in Table 5.1. Let q be boolean query. Then $\forall \mathcal{M}_i \in \{\mathcal{M}_1, \dots, \mathcal{M}_8\}$:

1. if q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_i \rangle$ then q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_i \rangle$.
2. if q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_i \rangle$ then q is a majority-based conclusion of $\langle \mathcal{T}, \mathcal{M}_i \rangle$.

Proof. The proof of item 1 (resp. item 2) follows immediately from the proof of Item 1 (resp. item 2) of Proposition 5.1. \square

Proposition B.11 (Figure 5.7). Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} = \{\mathcal{A}\} \rangle$ be an inconsistent DL-Lite knowledge base. Let $\mathcal{M}_1, \dots, \mathcal{M}_8$ be the eight MBox's given in Table 5.1. Let q be boolean query. Then $\forall \mathcal{M}_i \in \{\mathcal{M}_1, \dots, \mathcal{M}_8\}$:

1. if q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_3 \rangle$ then q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_4 \rangle$.
2. if q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_2 \rangle$ then q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_3 \rangle$.
3. if q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_1 \rangle$ then q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_2 \rangle$.
4. if q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$ then q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_1 \rangle$.
5. if q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$ then q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_6 \rangle$.
6. if q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_7 \rangle$ then q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_8 \rangle$.

Proof. The proof of item 1 follows from the facts that i) each safe conclusion of \mathcal{M}_3 is a universal conclusion of \mathcal{M}_3 and ii) $\mathcal{M}_4 \subseteq \mathcal{M}_3$. The proof of items 3, 5 and 6 follow similarly. Regarding item 2 (resp. Item 4), it holds due the fact that $(\mathcal{M}_2, \forall) \equiv (\mathcal{M}_3, \forall)$ (resp. $(\mathcal{M}_1, \forall) \equiv (\mathcal{M}_5, \forall)$). So if a conclusion holds from (\mathcal{M}_1, \cap) (resp. (\mathcal{M}_2, \cap)), then it holds from (\mathcal{M}_5, \forall) (resp. (\mathcal{M}_3, \forall)). \square

Proposition B.12 (Figure 5.7). Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} = \{\mathcal{A}\} \rangle$ be an inconsistent DL-Lite knowledge base. Let $\mathcal{M}_1, \dots, \mathcal{M}_8$ be the eight MBox's given in Table 5.1. Let q be boolean query. Then:

1. if q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_3 \rangle$ then q is a majority-based conclusion of $\langle \mathcal{T}, \mathcal{M}_4 \rangle$.
2. if q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_3 \rangle$ then q is a majority-based conclusion of $\langle \mathcal{T}, \mathcal{M}_2 \rangle$.
3. if q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_2 \rangle$ then q is a majority-based conclusion of $\langle \mathcal{T}, \mathcal{M}_3 \rangle$.
4. if q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_1 \rangle$ then q is a majority-based conclusion of $\langle \mathcal{T}, \mathcal{M}_2 \rangle$.
5. if q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_1 \rangle$ then q is a majority-based conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$.
6. if q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$ then q is a majority-based conclusion of $\langle \mathcal{T}, \mathcal{M}_1 \rangle$.
7. if q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$ then q is a majority-based conclusion of $\langle \mathcal{T}, \mathcal{M}_6 \rangle$.
8. if q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$ then q is a majority-based conclusion of $\langle \mathcal{T}, \mathcal{M}_7 \rangle$.
9. if q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_7 \rangle$ then q is a majority-based conclusion of $\langle \mathcal{T}, \mathcal{M}_8 \rangle$.

Proof. Item 1 holds due to the facts that i) the universal inference from \mathcal{M}_3 considers as valid a conclusion q iff q follows from each $\mathcal{A}_i \in \mathcal{M}_3$ and ii) since $\mathcal{M}_4 \subseteq \mathcal{M}_3$, then q is a majority-based conclusion of \mathcal{M}_4 . Item 4, 7 and 9 follow similarly. Items 2 and 3 follow immediately from the facts that $\mathcal{M}_3 = \circ_{cl}(\mathcal{M}_2)$ and $(\mathcal{M}_2, \forall) \equiv (\mathcal{M}_3, \forall)$. Then if a conclusion q is a universal conclusion of \mathcal{M}_2 (resp. \mathcal{M}_3) then it is a majority-based conclusion of \mathcal{M}_3 (resp. \mathcal{M}_2). Items 5 and 6 follow similarly. Item 8 holds from the fact that a universal conclusion of \mathcal{M}_5 is a universal conclusion of \mathcal{M}_7 , and thus, it is a majority-based conclusion of \mathcal{M}_7 . \square

Proposition B.13 (Figure 5.7). Let $\mathcal{K}_{\mathcal{M}} = \langle \mathcal{T}, \mathcal{M} = \{\mathcal{A}\} \rangle$ be an inconsistent DL-Lite knowledge base. Let $\mathcal{M}_1, \dots, \mathcal{M}_8$ be the eight MBox's given in Table 5.1. Let q be instance query. Then:

1. if q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_2 \rangle$ then q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_3 \rangle$.
2. if q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_1 \rangle$ then q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$.

-
3. q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$ iff q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$.
 4. q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_6 \rangle$ iff q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_6 \rangle$.
 5. q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_7 \rangle$ iff q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_7 \rangle$.
 6. q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_8 \rangle$ iff q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_8 \rangle$.
 7. if q is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$ iff q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_7 \rangle$.

Proof. The proof of Item 1 follows from the facts: i) $\mathcal{M}_2 \subseteq_{cl} \mathcal{M}_3$, ii) the set of universal conclusions of \mathcal{M}_2 $UC_{\mathcal{M}_2} = \{B(a) : a \in D_I, B \in D_C \text{ and } \forall \mathcal{A}_i \in \mathcal{M}_2, \langle \mathcal{T}, \mathcal{A}_i \rangle \models B(a)\} \cup \{R(a, b) : a \in D_I, b \in D_I, R \in D_R \text{ and } \forall \mathcal{A}_i \in \mathcal{M}_2, \langle \mathcal{T}, \mathcal{A}_i \rangle \models R(a, b)\}$, and iii) according to the definition of deductive closure modifier $UC_{\mathcal{M}_2} = \bigcap_{\mathcal{A}_j \in \mathcal{M}_3} \mathcal{A}_j$. The proof of item 2 and 7 follow similarly.

For item 3, we have from Proposition 5.1, if q is a safe conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$ then it is a universal conclusion of $\langle \mathcal{T}, \mathcal{M}_5 \rangle$. Now, let $UC_{\mathcal{M}_5} = \{B(a) : a \in D_I, B \in D_C \text{ and } \forall \mathcal{A}_i \in \mathcal{M}_5, \langle \mathcal{T}, \mathcal{A}_i \rangle \models B(a)\} \cup \{R(a, b) : a \in D_I, b \in D_I, R \in D_R \text{ and } \forall \mathcal{A}_i \in \mathcal{M}_5, \langle \mathcal{T}, \mathcal{A}_i \rangle \models R(a, b)\}$ be the set of universal conclusions of \mathcal{M}_5 . We have $\mathcal{M}_5 = \circ_{cl}(\circ_{incl}(\mathcal{M}))$, then according to the definition of deductive closure modifier, one can check that $UC_{\mathcal{M}_5} = \bigcap_{\mathcal{A}_j \in \mathcal{M}_5} \mathcal{A}_j$. The same proof holds for items 4, 5 and 6. \square

References

BIBLIOGRAPHY

- [Abiteboul *et al.*, 1995] Serge Abiteboul, Richard Hull, and Victor Vianu, editors. *Foundations of Databases: The Logical Level*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1st edition, 1995.
- [Adjiman *et al.*, 2006] Philippe Adjiman, Philippe Chatalic, François Goasdoué, Marie-Christine Rousset, and Laurent Simon. Distributed reasoning in a peer-to-peer setting: Application to the semantic web. *J. Artif. Intell. Res. (JAIR)*, 25:269–314, 2006.
- [Alchourrón *et al.*, 1985] Carlos E. Alchourrón, Peter Gärdenfors, and David Makinson. On the logic of theory change: Partial meet contraction and revision functions. *J. Symb. Log.*, 50(2):510–530, 1985.
- [Amgoud and Cayrol, 2002] Leila Amgoud and Claudette Cayrol. Inferring from inconsistency in preference-based argumentation frameworks. *J. Autom. Reasoning*, 29(2):125–169, 2002.
- [Amgoud and Prade, 2009] Leila Amgoud and Henri Prade. Using arguments for making and explaining decisions. *Artif. Intell.*, 173(3-4):413–436, 2009.
- [Amgoud, 2005] Leila Amgoud. A unified setting for inference and decision: An argumentation-based approach. In *UAI '05, Proceedings of the 21st Conference in Uncertainty in Artificial Intelligence, Edinburgh, Scotland, July 26-29, 2005*, pages 26–33. AUAI Press, 2005.
- [Artale *et al.*, 2009] Alessandro Artale, Diego Calvanese, Roman Kontchakov, and Michael Zakharyashev. The DL-Lite family and relations. *J. Artif. Intell. Res. (JAIR)*, 2009.
- [Artale *et al.*, 2013] Alessandro Artale, Roman Kontchakov, Frank Wolter, and Michael Zakharyashev. Temporal description logic for ontology-based data access. In Rossi [2013].
- [Ayachi *et al.*, 2013] Raouia Ayachi, Nahla Ben Amor, and Salem Benferhat. A comparative study of compilation-based inference methods for min-based possibilistic networks. In van der Gaag [2013], pages 25–36.
- [Baader and Hollunder, 1995] Franz Baader and Bernhard Hollunder. Priorities on defaults with prerequisites, and their application in treating specificity in terminological default logic. *J. Autom. Reasoning*, 15(1):41–68, 1995.
- [Baader *et al.*, 2005a] Franz Baader, Sebastian Brandt, and Carsten Lutz. Pushing the envelope. In *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence, IJCAI '05*, pages 364–369, 2005.
- [Baader *et al.*, 2005b] Franz Baader, Ian Horrocks, and Ulrike Sattler. Description logics as ontology languages for the semantic web. In Dieter Hutter and Werner Stephan, editors, *Mechanizing Mathematical Reasoning*, volume 2605 of *Lecture Notes in Computer Science*, pages 228–248. Springer, 2005.
- [Baader *et al.*, 2010] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider. *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, New York, NY, USA, 2nd edition, 2010.

- [Baets *et al.*, 1999] Bernard De Baets, Elena Tsiporkova, and Radko Mesiar. Conditioning in possibility theory with strict order norms. *Fuzzy Sets and Systems*, 106(2):221 – 229, 1999.
- [Baget *et al.*, 2009] Jean-François Baget, Michel Leclère, Marie-Laure Mugnier, and Eric Salvat. Extending decidable cases for rules with existential variables. In Boutilier [2009], pages 677–682.
- [Baget *et al.*, 2011] Jean-François Baget, Michel Leclère, Marie-Laure Mugnier, and Eric Salvat. On rules with existential variables: Walking the decidability line. *Artif. Intell.*, 175(9-10):1620–1654, 2011.
- [Baral *et al.*, 1992] Chitta Baral, Sarit Kraus, Jack Minker, and V. S. Subrahmanian. Combining knowledge bases consisting of first-order analysis. *Computational Intelligence*, 8:45–71, 1992.
- [Benamor, 2002] Nahla Benamor. *Qualitative possibilistic graphical models: From independence to propagation algorithms*. PhD thesis, Institut superieur de gestion de Tunis, 2002.
- [Benferhat and Kaci, 2003] Salem Benferhat and Souhila Kaci. Fusion of possibilistic knowledge bases from a postulate point of view. *Int. J. Approx. Reasoning*, 33(3):255–285, 2003.
- [Benferhat *et al.*, 1992] Salem Benferhat, Didier Dubois, and Henri Prade. Representing default rules in possibilistic logic. In Bernhard Nebel, Charles Rich, and William R. Swartout, editors, *KR*, pages 673–684. Morgan Kaufmann, 1992.
- [Benferhat *et al.*, 1993a] Salem Benferhat, Claudette Cayrol, Didier Dubois, Jérôme Lang, and Henri Prade. Inconsistency management and prioritized syntax-based entailment. In Ruzena Bajcsy, editor, *IJCAI*, pages 640–647. Morgan Kaufmann, 1993.
- [Benferhat *et al.*, 1993b] Salem Benferhat, Didier Dubois, and Henri Prade. Argumentative inference in uncertain and inconsistent knowledge bases. In David Heckerman and E. H. Mamdani, editors, *UAI*, pages 411–419. Morgan Kaufmann, 1993.
- [Benferhat *et al.*, 1995] Salem Benferhat, Didier Dubois, and Henri Prade. How to infer from inconsistent beliefs without revising? In *IJCAI*, pages 1449–1457. Morgan Kaufmann, 1995.
- [Benferhat *et al.*, 1997a] Salem Benferhat, Didier Dubois, and Henri Prade. Some syntactic approaches to the handling of inconsistent knowledge bases: A comparative study part 1: The flat case. *Studia Logica*, 58(1):17–45, 1997.
- [Benferhat *et al.*, 1997b] Salem Benferhat, Didier Dubois, and Henri Prade. Syntactic combination of uncertain information: A possibilistic approach. In Dov M. Gabbay, Rudolf Kruse, Andreas Nonnen-gart, and Hans Jürgen Ohlbach, editors, *ECSQARU-FAPR*, volume 1244 of *Lecture Notes in Computer Science*, pages 30–42. Springer, 1997.
- [Benferhat *et al.*, 1998a] Salem Benferhat, D. Dubois, and H. Prade. *Some syntactic approaches to the handling of inconsistent knowledge bases : A comparative study. Part 2 : the prioritized case*, volume 24, pages 473–511. Physica-Verlag, Heidelberg, 1998.
- [Benferhat *et al.*, 1998b] Salem Benferhat, Didier Dubois, Jérôme Lang, Henri Prade, Alessandro Saffiotti, and Philippe Smets. A general approach for inconsistency handling and merging information in prioritized knowledge bases. In Anthony G. Cohn, Lenhart K. Schubert, and Stuart C. Shapiro, editors, *KR*, pages 466–477. Morgan Kaufmann, 1998.
- [Benferhat *et al.*, 1999] Salem Benferhat, Didier Dubois, Henri Prade, and Mary-Anne Williams. A practical approach to fusing prioritized knowledge bases. In *EPIA*, pages 223–236, 1999.

- [Benferhat *et al.*, 2000] Salem Benferhat, Souhila Kaci, Didier Dubois, and Henri Prade. A principled analysis of merging operations in possibilistic logic. In Craig Boutilier and Moisés Goldszmidt, editors, *UAI*, pages 24–31. Morgan Kaufmann, 2000.
- [Benferhat *et al.*, 2001] Salem Benferhat, Didier Dubois, Souhila Kaci, and Henri Prade. Graphical readings of possibilistic logic bases. In Jack S. Breese and Daphne Koller, editors, *UAI '01: Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence, University of Washington, Seattle, Washington, USA, August 2-5, 2001*, pages 24–31. Morgan Kaufmann, 2001.
- [Benferhat *et al.*, 2002a] Salem Benferhat, Didier Dubois, Laurent Garcia, and Henri Prade. On the transformation between possibilistic logic bases and possibilistic causal networks. *Int. J. Approx. Reasoning*, 29(2):135–173, 2002.
- [Benferhat *et al.*, 2002b] Salem Benferhat, Didier Dubois, Sylvain Lagrue, and Odile Papini. Making revision reversible: an approach based on polynomials. *Fundam. Inform.*, 53(3-4):251–280, 2002.
- [Benferhat *et al.*, 2002c] Salem Benferhat, Didier Dubois, Henri Prade, and Mary-Anne Williams. A practical approach to revising prioritized knowledge bases. *Studia Logica*, 70(1):105–130, 2002.
- [Benferhat *et al.*, 2007] Salem Benferhat, Sylvain Lagrue, and Julien Rossit. An egalitarian fusion of incommensurable ranked belief bases under constraints. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22-26, 2007, Vancouver, British Columbia, Canada [2007]*, pages 367–372.
- [Benferhat *et al.*, 2010a] Salem Benferhat, Jonathan Ben-Naim, Odile Papini, and Eric Würbel. An answer set programming encoding of prioritized removed sets revision: application to gis. *Appl. Intell.*, 32(1):60–87, 2010.
- [Benferhat *et al.*, 2010b] Salem Benferhat, Didier Dubois, Henri Prade, and Mary-Anne Williams. A framework for iterated belief revision using possibilistic counterparts to jeffrey’s rule. *Fundam. Inform.*, 99(2):147–168, 2010.
- [Benferhat *et al.*, 2011] Salem Benferhat, Karim Tabia, and Karima Sedki. Jeffrey’s rule of conditioning in a possibilistic framework. *Annals of Mathematics and Artificial Intelligence*, 61(3):185–202, March 2011.
- [Bertossi *et al.*, 2005a] Leopoldo E. Bertossi, Anthony Hunter, and Torsten Schaub, editors. *Inconsistency Tolerance [result from a Dagstuhl seminar]*, volume 3300 of *Lecture Notes in Computer Science*. Springer, 2005.
- [Bertossi *et al.*, 2005b] Leopoldo E. Bertossi, Anthony Hunter, and Torsten Schaub. Introduction to inconsistency tolerance. In *Inconsistency Tolerance [result from a Dagstuhl seminar] [2005a]*, pages 1–14.
- [Bertossi, 2011] Leopoldo E. Bertossi. *Database Repairing and Consistent Query Answering*. Synthesis Lectures on Data Management. Morgan & Claypool Publishers, 2011.
- [Besnard and Hunter, 1995] Philippe Besnard and Anthony Hunter. Quasi-classical logic: Non-trivializable classical reasoning from inconsistent information. In Christine Froidevaux and Jürg Kohlas, editors, *Symbolic and Quantitative Approaches to Reasoning and Uncertainty, European Conference, ECSQARU'95, Fribourg, Switzerland, July 3-5, 1995, Proceedings*, volume 946 of *Lecture Notes in Computer Science*, pages 44–51. Springer, 1995.

- [Besnard and Hunter, 2008] Philippe Besnard and Anthony Hunter. *Elements of Argumentation*. MIT Press, 2008.
- [Bienvenu and Rosati, 2013] Meghyn Bienvenu and Riccardo Rosati. Tractable approximations of consistent query answering for robust ontology-based data access. In Rossi [2013].
- [Bienvenu *et al.*, 2014] Meghyn Bienvenu, Camille Bourgaux, and François Goasdoué. Querying inconsistent description logic knowledge bases under preferred repair semantics. In Brodley and Stone [2014], pages 996–1002.
- [Bienvenu, 2012] Meghyn Bienvenu. On the complexity of consistent query answering in the presence of simple ontologies. In Hoffmann and Selman [2012].
- [Bloch *et al.*, 2001] Isabelle Bloch, Anthony Hunter, Alain Appriou, André Ayoun, Salem Benferhat, Philippe Besnard, Laurence Cholvy, Roger M. Cooke, Frédéric Cuppens, Didier Dubois, Hélène Fargier, Michel Grabisch, Rudolf Kruse, Jérôme Lang, Serafín Moral, Henri Prade, Alessandro Saffiotti, Philippe Smets, and Claudio Sossai. Fusion: General concepts and characteristics. *Int. J. Intell. Syst.*, 16(10):1107–1134, 2001.
- [Bobillo and Straccia, 2007] Fernando Bobillo and Umberto Straccia. A fuzzy description logic with product t-norm. In *FUZZ-IEEE 2007, IEEE International Conference on Fuzzy Systems, Imperial College, London, UK, 23-26 July, 2007, Proceedings*, pages 1–6. IEEE, 2007.
- [Bobillo and Straccia, 2012] Fernando Bobillo and Umberto Straccia. Generalized fuzzy rough description logics. *Information Sciences*, 189:43–62, 2012.
- [Bobillo *et al.*, 2012] Fernando Bobillo, Miguel Delgado, and Juan Gómez-Romero. Delorean: A reasoner for fuzzy OWL 2. *Expert Systems with Applications*, 39(1):258–272, 2012.
- [Bobillo *et al.*, 2013] Fernando Bobillo, Miguel Delgado, and Juan Gómez-Romero. Reasoning in fuzzy OWL 2 with delorean. In Fernando Bobillo, Paulo Cesar G. da Costa, Claudia d’Amato, Nicola Fanizzi, Kathryn B. Laskey, Kenneth J. Laskey, Thomas Lukasiewicz, Matthias Nickles, and Michael Pool, editors, *Uncertainty Reasoning for the Semantic Web II*, volume 7123 of *Lecture Notes in Computer Science*, pages 119–138. Springer, 2013.
- [Borgida, 1985] Alexander Borgida. Language features for flexible handling of exceptions in information systems. *ACM Trans. Database Syst.*, 10(4):565–603, December 1985.
- [Bouchon-Meunier *et al.*, 2002] Bernadette Bouchon-Meunier, Giulianella Coletti, and Christophe Marsala. Independence and possibilistic conditioning. *Annals of Mathematics and Artificial Intelligence*, 35(1-4):107–123, 2002.
- [Boutilier, 1993] Craig Boutilier. Revision sequences and nested conditionals. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence - Volume 1, IJCAI’93*, pages 519–525, San Francisco, CA, USA, 1993. Morgan Kaufmann Publishers Inc.
- [Boutilier, 2009] Craig Boutilier, editor. *IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, California, USA, July 11-17, 2009*, 2009.
- [Brachman and Levesque, 1985] Ronald J. Brachman and Hector J. Levesque, editors. *Readings in Knowledge Representation*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1985.
- [Brewka, 1989] Gerhard Brewka. Preferred subtheories: An extended logical framework for default reasoning. In N. S. Sridharan, editor, *Proceedings of the 11th International Joint Conference on Artificial Intelligence. Detroit, MI, USA, August 1989*, pages 1043–1048. Morgan Kaufmann, 1989.

- [Brodley and Stone, 2014] Carla E. Brodley and Peter Stone, editors. *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada*. AAAI Press, 2014.
- [Calì *et al.*, 2012] Andrea Calì, Georg Gottlob, and Thomas Lukasiewicz. A general datalog-based framework for tractable query answering over ontologies. *J. Web Sem.*, 14:57–83, 2012.
- [Calvanese *et al.*, 2005] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, and R. Rosati. DL-Lite: Tractable description logics for ontologies. In *Proceedings, The Twentieth National Conference on Artificial Intelligence and the Seventeenth Innovative Applications of Artificial Intelligence Conference, AAAI '05*, pages 602–607. AAAI Press / The MIT Press, 2005.
- [Calvanese *et al.*, 2006] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, and R. Rosati. Data complexity of query answering in description logics. In Doherty *et al.* [2006], pages 260–270.
- [Calvanese *et al.*, 2007a] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, and R. Rosati. Tractable reasoning and efficient query answering in description logics: The *dl-lite* family. *J. Autom. Reasoning*, 39(3):385–429, 2007.
- [Calvanese *et al.*, 2007b] Diego Calvanese, Enrico Franconi, Volker Haarslev, Domenico Lembo, Boris Motik, Anni-Yasmin Turhan, and Sergio Tessaris, editors. *Proceedings of the 2007 International Workshop on Description Logics (DL2007), Brixen-Bressanone, near Bozen-Bolzano, Italy, 8-10 June, 2007*, volume 250 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2007.
- [Calvanese *et al.*, 2010] Diego Calvanese, Evgeny Kharlamov, Werner Nutt, and Dmitriy Zheleznyakov. Evolution of DL-Lite knowledge bases. In *International Semantic Web Conference (1)*, pages 112–128, 2010.
- [Chein and Mugnier, 2009] Michel Chein and Marie-Laure Mugnier. *Graph-based Knowledge Representation - Computational Foundations of Conceptual Graphs*. Advanced Information and Knowledge Processing. Springer, 2009.
- [Chein and Mugnier, 2014] Michel Chein and Marie-Laure Mugnier. Conceptual graphs are also graphs. In Nathalie Hernandez, Robert Jäschke, and Madalina Croitoru, editors, *Graph-Based Representation and Reasoning - 21st International Conference on Conceptual Structures, ICCS 2014, Iași, Romania, July 27-30, 2014, Proceedings*, volume 8577 of *Lecture Notes in Computer Science*, pages 1–18. Springer, 2014.
- [Chomicki, 2007] Jan Chomicki. Consistent query answering: Five easy pieces. In Thomas Schwentick and Dan Suciu, editors, *Database Theory - ICDT 2007, 11th International Conference, Barcelona, Spain, January 10-12, 2007, Proceedings*, volume 4353 of *Lecture Notes in Computer Science*, pages 1–17. Springer, 2007.
- [Cook, 1971] Stephen A. Cook. The complexity of theorem-proving procedures. In Michael A. Harrison, Ranan B. Banerji, and Jeffrey D. Ullman, editors, *Proceedings of the 3rd Annual ACM Symposium on Theory of Computing, May 3-5, 1971, Shaker Heights, Ohio, USA*, pages 151–158. ACM, 1971.
- [Corcho *et al.*, 2009] Óscar Corcho, Catherine Roussey, Luis Manuel Vilches Blázquez, and Iván Pérez. Pattern-based OWL ontology debugging guidelines. In Eva Blomqvist, Kurt Sandkuhl, François Scharffe, and Vojtech Svátek, editors, *Proceedings of the Workshop on Ontology Patterns (WOP 2009), collocated with the 8th International Semantic Web Conference (ISWC-2009), Washington D.C., USA, 25 October, 2009.*, volume 516 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2009.

- [Couchariere *et al.*, 2008a] Olivier Couchariere, Marie-Jeanne Lesot, and Bernadette Bouchon-Meunier. Consistency checking for extended description logics. In *Proceedings of the 21st International Workshop on Description Logics DL2008*, volume 9 of *Description Logics*, pages 602–607. CEUR-WS.org / CEUR Workshop Proceedings, 2008.
- [Couchariere *et al.*, 2008b] Olivier Couchariere, Marie-Jeanne Lesot, Bernadette Bouchon-Meunier, and J.-L. Rogier. Inconsistency degree computation for possibilistic description logic: an extension of the tableau algorithm. In *Fuzzy Information Processing Society, 2008. NAFIPS 2008. Annual Meeting of the North American*, pages 1–6, May 2008.
- [Croitoru and Vesic, 2013] Madalina Croitoru and Srdjan Vesic. What can argumentation do for inconsistent ontology query answering? In Weiru Liu, V. S. Subrahmanian, and Jef Wijsen, editors, *Scalable Uncertainty Management - 7th International Conference, SUM 2013, Washington, DC, USA, September 16-18, 2013. Proceedings*, volume 8078 of *Lecture Notes in Computer Science*, pages 15–29. Springer, 2013.
- [Dalal, 1988] Mukesh Dalal. Investigations into a theory of knowledge base revision. In Howard E. Shrobe, Tom M. Mitchell, and Reid G. Smith, editors, *Proceedings of the 7th National Conference on Artificial Intelligence. St. Paul, MN, August 21-26, 1988.*, pages 475–479. AAAI Press / The MIT Press, 1988.
- [Darwiche and Pearl, 1997] Adnan Darwiche and Judea Pearl. On the logic of iterated belief revision. *Artificial Intelligence*, 89(1–2):1 – 29, 1997.
- [DBL, 2006] *Proceedings, The Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference, July 16-20, 2006, Boston, Massachusetts, USA*. AAAI Press, 2006.
- [DBL, 2007] *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22-26, 2007, Vancouver, British Columbia, Canada*. AAAI Press, 2007.
- [Deagustini *et al.*, 2014] Cristhian Ariel David Deagustini, Maria Vanina Martinez, Marcelo A. Falappa, and Guillermo Ricardo Simari. Improving inconsistency resolution by considering global conflicts. In Umberto Straccia and Andrea Cali, editors, *Scalable Uncertainty Management - 8th International Conference, SUM 2014, Oxford, UK, September 15-17, 2014. Proceedings*, volume 8720 of *Lecture Notes in Computer Science*, pages 120–133. Springer, 2014.
- [Delgrande *et al.*, 2006] James P. Delgrande, Didier Dubois, and Jérôme Lang. Iterated revision as prioritized merging. In Doherty *et al.* [2006], pages 210–220.
- [Doherty *et al.*, 2006] Patrick Doherty, John Mylopoulos, and Christopher A. Welty, editors. *Proceedings, Tenth International Conference on Principles of Knowledge Representation and Reasoning, Lake District of the United Kingdom, June 2-5, 2006*. AAAI Press, 2006.
- [Dou *et al.*, 2005] Dejing Dou, Drew V. McDermott, and Peishen Qi. Ontology translation on the semantic web. *J. Data Semantics*, 2:35–57, 2005.
- [Dou *et al.*, 2011] Dejing Dou, Han Qin, and Haishan Liu. Semantic translation for rule-based knowledge in data mining. In Abdelkader Hameurlain, Stephen W. Liddle, Klaus-Dieter Schewe, and Xiaofang Zhou, editors, *Database and Expert Systems Applications - 22nd International Conference, DEXA 2011, Toulouse, France, August 29 - September 2, 2011, Proceedings, Part II*, volume 6861 of *Lecture Notes in Computer Science*, pages 74–89. Springer, 2011.

-
- [Du *et al.*, 2013] Jianfeng Du, Guilin Qi, and Yi-Dong Shen. Weight-based consistent query answering over inconsistent SHIQ knowledge bases. *Knowl. Inf. Syst.*, 34(2):335–371, 2013.
- [Dubois and Prade, 1988a] Didier Dubois and Henri Prade. Conditioning in possibility and evidence theories - A logical viewpoint. In Bernadette Bouchon-Meunier, Lorenza Saitta, and Ronald R. Yager, editors, *Uncertainty and Intelligent Systems, 2nd International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, IPMU '88, Urbino, Italy, July 4-7, 1988, Proceedings*, volume 313 of *Lecture Notes in Computer Science*, pages 401–408. Springer, 1988.
- [Dubois and Prade, 1988b] Didier Dubois and Henri Prade. Possibility theory. *Plenum Press, New-York*, 1988.
- [Dubois and Prade, 1990] Didier Dubois and Henri Prade. The logical view of conditioning and its application to possibility and evidence theories. *Int. J. Approx. Reasoning*, 4(1):23–46, 1990.
- [Dubois and Prade, 1991a] Didier Dubois and Henri Prade. Epistemic entrenchment and possibilistic logic. *Artif. Intell.*, 50(2):223–239, 1991.
- [Dubois and Prade, 1991b] Didier Dubois and Henri Prade. Inference in possibilistic hypergraphs. In Bernadette Bouchon-Meunier, Ronald R. Yager, and Lotfi A. Zadeh, editors, *Uncertainty in Knowledge Bases*, volume 521 of *Lecture Notes in Computer Science*, pages 249–259. Springer Berlin Heidelberg, 1991.
- [Dubois and Prade, 1992] Didier Dubois and Henri Prade. When upper probabilities are possibility measures. *Fuzzy Sets and Systems*, 49(1):65 – 74, 1992.
- [Dubois and Prade, 1997] Didier Dubois and Henri Prade. A synthetic view of belief revision with uncertain inputs in the framework of possibility theory. *International Journal of Approximate Reasoning*, 17(2–3):295 – 324, 1997.
- [Dubois and Prade, 1998] Didier Dubois and Henri Prade. Possibility theory: Qualitative and quantitative aspects. In Philippe Smets, editor, *Quantified Representation of Uncertainty and Imprecision*, volume 1 of *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, pages 169–226. Springer Netherlands, 1998.
- [Dubois and Prade, 2010] Didier Dubois and Henri Prade. *Formal Representations of Uncertainty*. ISTE, 2010.
- [Dubois and Prade, 2011] Didier Dubois and Henri Prade. Possibility theory and its applications: Where do we stand?, 2011.
- [Dubois *et al.*, 1992] Didier Dubois, Jérôme Lang, and Henri Prade. Dealing with multi-source information in possibilistic logic. In *ECAI*, pages 38–42, 1992.
- [Dubois *et al.*, 1994] Didier Dubois, Jérôme Lang, and Henri Prade. Possibilistic logic. volume 3 of *In Handbook on Logic in Artificial Intelligence and Logic Programming*, pages 439–513. Oxford University press, 1994.
- [Dubois *et al.*, 1997] Didier Dubois, Serafin Moral, and Henri Prade. A semantics for possibility theory based on likelihoods. *Journal of Mathematical Analysis and Applications*, 205(2):359 – 380, 1997.
- [Dubois *et al.*, 2003] Didier Dubois, Sébastien Konieczny, and Henri Prade. Quasi-possibilistic logic and its measures of information and conflict. *Fundam. Inform.*, 57(2-4):101–125, 2003.

- [Dubois *et al.*, 2006] Didier Dubois, Jérôme Mengin, and Henri Prade. Possibilistic uncertainty and fuzzy features in description logic. a preliminary discussion. In Elie Sanchez, editor, *Fuzzy Logic and the Semantic Web*, volume 1 of *Capturing Intelligence*, pages 101 – 113. Elsevier, 2006.
- [Euzenat and Shvaiko, 2013] Jérôme Euzenat and Pavel Shvaiko. *Ontology Matching, Second Edition*. Springer, 2013.
- [Everaere *et al.*, 2010] Patricia Everaere, Sébastien Konieczny, and Pierre Marquis. Disjunctive merging: Quota and gmin merging operators. *Artif. Intell.*, 174(12-13):824–849, 2010.
- [Everaere *et al.*, 2012] Patricia Everaere, Sébastien Konieczny, and Pierre Marquis. Compositional belief merging. In Gerhard Brewka, Thomas Eiter, and Sheila A. McIlraith, editors, *KR*. AAAI Press, 2012.
- [Fermé and Hansson, 1999] Eduardo L. Fermé and Sven Ove Hansson. Selective revision. *Studia Logica*, 63(3):331–342, 1999.
- [Fitting, 1990] Melvin Fitting. *First-Order Logic and Automated Theorem Proving*. Texts and Monographs in Computer Science. Springer, 1990.
- [Flesca *et al.*, 2002] Sergio Flesca, Sergio Greco, Nicola Leone, and Giovambattista Ianni, editors. *Logics in Artificial Intelligence, European Conference, JELIA 2002, Cosenza, Italy, September, 23-26, Proceedings*, volume 2424 of *Lecture Notes in Computer Science*. Springer, 2002.
- [Flouris *et al.*, 2004] Giorgos Flouris, Dimitris Plexousakis, and Grigoris Antoniou. Generalizing the agm postulates: preliminary results and applications. In James P. Delgrande and Torsten Schaub, editors, *NMR*, pages 171–179, 2004.
- [Flouris *et al.*, 2005] Giorgos Flouris, Dimitris Plexousakis, and Grigoris Antoniou. On applying the agm theory to dls and owl. In *ISWC*, pages 216–231, 2005.
- [Flouris *et al.*, 2006a] Giorgos Flouris, Zhisheng Huang, Jeff Z. Pan, Dimitris Plexousakis, and Holger Wache. Inconsistencies, negations and changes in ontologies. In *AAAI [2006]*, pages 1295–1300.
- [Flouris *et al.*, 2006b] Giorgos Flouris, Dimitris Plexousakis, and Grigoris Antoniou. On generalizing the agm postulates. In *STAIRS*, pages 132–143, 2006.
- [Fonck, 1994] P. Fonck. *Réseaux d'inférence pour le raisonnement possibiliste*. PhD thesis, Université de Liège, Faculté des Sciences, 1994.
- [Fonck, 1997] Pascale Fonck. A comparative study of possibilistic conditional independence and lack of interaction. *International Journal of Approximate Reasoning*, 16(2):149 – 171, 1997.
- [Fuhrmann and Hansson, 1994] André Fuhrmann and Sven Ove Hansson. A survey of multiple contractions. *Journal of Logic, Language and Information*, 3(1):39–75, 1994.
- [Fuhrmann, 1997] A. Fuhrmann. *An essay on contraction*. CSLI Publications, Stanford. California, 1997.
- [Gao *et al.*, 2012] Sibe Gao, Guilin Qi, and Haofen Wang. A new operator for abox revision in DL-Lite. In *AAAI*, 2012.
- [Garey and Johnson, 1979] M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.

- [Garriga, 2013] Gemma C. Garriga. *Formal Methods for Mining Structured Objects*, volume 475 of *Studies in Computational Intelligence*. Springer, 2013.
- [Gebhardt and Kruse, 1996] Jörg Gebhardt and Rudolf Kruse. Learning possibilistic networks from data. In Doug Fisher and Hans-J. Lenz, editors, *Learning from Data*, volume 112 of *Lecture Notes in Statistics*, pages 143–153. Springer New York, 1996.
- [Giacomo *et al.*, 2007] Giuseppe De Giacomo, Maurizio Lenzerini, Antonella Poggi, and Riccardo Rosati. On the approximation of instance level update and erasure in description logics. In *AAAI [2007]*, pages 403–408.
- [Giugno and Lukasiewicz, 2002] Rosalba Giugno and Thomas Lukasiewicz. P-SHOQ(D): A probabilistic extension of SHOQ(D) for probabilistic ontologies in the semantic web. In Flesca *et al.* [2002], pages 86–97.
- [Grant and Hunter, 2011] John Grant and Anthony Hunter. Measuring the good and the bad in inconsistent information. In Toby Walsh, editor, *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011*, pages 2632–2637. IJCAI/AAAI, 2011.
- [Grant and Hunter, 2013] John Grant and Anthony Hunter. Distance-based measures of inconsistency. In van der Gaag [2013], pages 230–241.
- [Grau *et al.*, 2008] Bernardo Cuenca Grau, Ian Horrocks, Yevgeny Kazakov, and Ulrike Sattler. Modular reuse of ontologies: Theory and practice. *J. Artif. Intell. Res. (JAIR)*, 31:273–318, 2008.
- [Grau *et al.*, 2009] Bernardo Cuenca Grau, Ian Horrocks, Yevgeny Kazakov, and Ulrike Sattler. Extracting modules from ontologies: A logic-based approach. In Heiner Stuckenschmidt, Christine Parent, and Stefano Spaccapietra, editors, *Modular Ontologies: Concepts, Theories and Techniques for Knowledge Modularization*, volume 5445 of *Lecture Notes in Computer Science*, pages 159–186. Springer, 2009.
- [Grosz *et al.*, 2003] Benjamin N. Grosz, Ian Horrocks, Raphael Volz, and Stefan Decker. Description logic programs: combining logic programs with description logic. In Gusztáv Hencsey, Bebo White, Yih-Farn Robin Chen, László Kovács, and Steve Lawrence, editors, *Proceedings of the Twelfth International World Wide Web Conference, WWW 2003, Budapest, Hungary, May 20-24, 2003*, pages 48–57. ACM, 2003.
- [Grove, 1988] Adam Grove. Two modellings for theory change. *Journal of Philosophical Logic*, 17(2):157–170, 1988.
- [Gruber, 1993] Thomas R. Gruber. A translation approach to portable ontology specifications. *Knowl. Acquis.*, 5(2):199–220, June 1993.
- [Gruber, 1995] Thomas R. Gruber. Toward principles for the design of ontologies used for knowledge sharing? *Int. J. Hum.-Comput. Stud.*, 43(5-6):907–928, 1995.
- [Haase *et al.*, 2005] Peter Haase, Frank van Harmelen, Zhisheng Huang, Heiner Stuckenschmidt, and York Sure. A framework for handling inconsistency in changing ontologies. In Yolanda Gil, Enrico Motta, V. Richard Benjamins, and Mark A. Musen, editors, *The Semantic Web - ISWC 2005, 4th International Semantic Web Conference, ISWC 2005, Galway, Ireland, November 6-10, 2005, Proceedings*, volume 3729 of *Lecture Notes in Computer Science*, pages 353–367. Springer, 2005.

- [Halaschek-wiener *et al.*, 2006] Christian Halaschek-wiener, Yarden Katz, and Bijan Parsia. Belief base revision for expressive description logics. In *In Proc. of Workshop on OWL Experiences and Directions*, 2006.
- [Hansson, 1992] Sven Ove Hansson. A dyadic representation of belief. pages 89–121, 1992.
- [Hansson, 1994] Sven Ove Hansson. Kernel contraction. *Journal of Symbolic Logic*, 59:845–859, 1994.
- [Hansson, 1997] Sven Ove Hansson. Semi-revision. *Journal of Applied Non-classical Logic*, 7:151–175, 1997.
- [Hansson, 1998] Sven Ove Hansson. Revision of belief sets and belief bases. *Handbook of Defeasible Reasoning and Uncertainty Management systems*, 3:17–75, 1998.
- [Hansson, 2008] Sven Ove Hansson. Specified meet contraction. *Erkenntnis*, 69(1):31–54, 2008.
- [Hisdal, 1978] Ellen Hisdal. Conditional possibilities independence and noninteraction. *Fuzzy Sets and Systems*, 1(4):283 – 297, 1978.
- [Hitzler *et al.*, 2009] Pascal Hitzler, Markus Krötzsch, and Sebastian Rudolph. *Foundations of Semantic Web Technologies*. Chapman & Hall/CRC, 2009.
- [Hoffmann and Selman, 2012] Jörg Hoffmann and Bart Selman, editors. *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada*. AAAI Press, 2012.
- [Hollunder, 1995] B. Hollunder. An alternative proof method for possibilistic logic and its application to terminological logics. *International Journal of Approximate Reasoning*, 12(2):85–109, 1995.
- [Horrocks and Patel-Schneider, 2004] Ian Horrocks and Peter F. Patel-Schneider. Reducing OWL entailment to description logic satisfiability. *J. Web Sem.*, 1(4):345–357, 2004.
- [Hou *et al.*, 2005] Chih-Sheng Johnson Hou, Mark A. Musen, and Natalya Fridman Noy. EZPAL: environment for composing constraint axioms by instantiating templates. *Int. J. Hum.-Comput. Stud.*, 62(5):578–596, 2005.
- [Hué *et al.*, 2008] J. Hué, E. Würbel, and O. Papini. Removed sets fusion: Performing off the shelf. In *Proc. of ECAI’08 (FIAI 178)*, pages 94–98, 2008.
- [Hunter and Konieczny, 2005] Anthony Hunter and Sébastien Konieczny. Approaches to measuring inconsistent information. In Bertossi *et al.* [2005a], pages 191–236.
- [Hunter and Konieczny, 2010] Anthony Hunter and Sébastien Konieczny. On the measure of conflicts: Shapley inconsistency values. *Artif. Intell.*, 174(14):1007–1026, 2010.
- [Hunter, 1998] Anthony Hunter. *Handbook of defeasible reasoning and uncertainty management systems*. chapter Paraconsistent Logics, pages 11–36. Kluwer Academic Publishers, Norwell, MA, USA, 1998.
- [Huth and Ryan, 2004] Michael Huth and Mark Dermot Ryan. *Logic in computer science - modelling and reasoning about systems (2. ed.)*. Cambridge University Press, 2004.
- [Jeffrey, 1965] J Jeffrey. *The logic of decision*. McGraw-Hill Press, 1965.
- [Kalyanpur *et al.*, 2005] Aditya Kalyanpur, Bijan Parsia, Evren Sirin, and James A. Hendler. Debugging unsatisfiable classes in OWL ontologies. *J. Web Sem.*, 3(4):268–293, 2005.

- [Katsuno and Mendelzon, 1991] Hirofumi Katsuno and Alberto O. Mendelzon. Propositional knowledge base revision and minimal change. *Artificial Intelligence*, 52(3):263–294, 1991.
- [Kharlamov and Zheleznyakov, 2011] Evgeny Kharlamov and Dmitriy Zheleznyakov. Understanding inexpressibility of model-based abox evolution in DL-Lite. In P. Barceló and V. Tannen, editors, *AMW*, volume 749 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2011.
- [Kharlamov *et al.*, 2013] Evgeny Kharlamov, Dmitriy Zheleznyakov, and Diego Calvanese. Capturing model-based ontology evolution at the instance level: The case of DL-Lite. *J. Comput. Syst. Sci.*, 79(6):835–872, 2013.
- [Kifer and Lausen, 1989] Michael Kifer and Georg Lausen. F-logic: A higher-order language for reasoning about objects, inheritance, and scheme. In James Clifford, Bruce G. Lindsay, and David Maier, editors, *Proceedings of the 1989 ACM SIGMOD International Conference on Management of Data, Portland, Oregon, May 31 - June 2, 1989.*, pages 134–146. ACM Press, 1989.
- [Knuth, 1998] Donald E. Knuth. *The Art of Computer Programming*, volume 2: Seminumerical Algorithms, pages 694–695. Addison Wesley, 1998.
- [Konieczny and Marquis, 2002] Sébastien Konieczny and Pierre Marquis. Three-valued logics for inconsistency handling. In Flesca *et al.* [2002], pages 332–344.
- [Konieczny and Pino Pérez, 2002] Sébastien Konieczny and Ramón Pino Pérez. Merging information under constraints: A logical framework. *J. Log. Comput.*, 12(5):773–808, 2002.
- [Konieczny *et al.*, 2005] Sébastien Konieczny, Jérôme Lang, and Pierre Marquis. Reasoning under inconsistency: the forgotten connective. In Leslie Pack Kaelbling and Alessandro Saffiotti, editors, *IJCAI-05, Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence, Edinburgh, Scotland, UK, July 30-August 5, 2005*, pages 484–489. Professional Book Center, 2005.
- [Konieczny *et al.*, 2008] Sébastien Konieczny, Pierre Marquis, and Philippe Besnard. Bipolarity in bi-lattice logics. *Int. J. Intell. Syst.*, 23(10):1046–1061, 2008.
- [Kotis *et al.*, 2006] Konstantinos Kotis, George A. Vouros, and Konstantinos Stergiou. Towards automatic merging of domain ontologies: The hcone-merge approach. *J. Web Sem.*, 4(1):60–79, 2006.
- [Lam *et al.*, 2008] Joey Sik Chun Lam, Derek H. Sleeman, Jeff Z. Pan, and Wamberto Weber Vasconcelos. A fine-grained approach to resolving unsatisfiable ontologies. In *J. Data Semantics* [2008], pages 62–95.
- [Lang and Marquis, 2002] Jérôme Lang and Pierre Marquis. Resolving inconsistencies by variable forgetting. In Dieter Fensel, Fausto Giunchiglia, Deborah L. McGuinness, and Mary-Anne Williams, editors, *Proceedings of the Eight International Conference on Principles and Knowledge Representation and Reasoning (KR-02), Toulouse, France, April 22-25, 2002*, pages 239–250. Morgan Kaufmann, 2002.
- [Lang and Marquis, 2010] Jérôme Lang and Pierre Marquis. Reasoning under inconsistency: A forgetting-based approach. *Artificial Intelligence*, 174(12–13):799 – 823, 2010.
- [Lehmann, 1995] Daniel Lehmann. Another perspective on default reasoning. *Annals of Mathematics and Artificial Intelligence*, 15(1):61–82, 1995.
- [Lembo and Ruzzi, 2007] Domenico Lembo and Marco Ruzzi. Consistent query answering over description logic ontologies. In Calvanese *et al.* [2007b].

- [Lembo *et al.*, 2010] Domenico Lembo, Maurizio Lenzerini, Riccardo Rosati, Marco Ruzzi, and Domenico Fabio Savo. Inconsistency-tolerant semantics for description logics. In Pascal Hitzler and Thomas Lukasiewicz, editors, *RR*, volume 6333 of *Lecture Notes in Computer Science*, pages 103–117. Springer, 2010.
- [Liu *et al.*, 2006] Hongkai Liu, Carsten Lutz, Maja Milicic, and Frank Wolter. Updating description logic aboxes. In Doherty *et al.* [2006], pages 46–56.
- [Ludwig and Peñaloza, 2014] Michel Ludwig and Rafael Peñaloza. Error-tolerant reasoning in the description logic $\mathcal{E}\{\text{kern-.1em}\}L$. In Eduardo Fermé and João Leite, editors, *Logics in Artificial Intelligence - 14th European Conference, JELIA 2014, Funchal, Madeira, Portugal, September 24-26, 2014. Proceedings*, volume 8761 of *Lecture Notes in Computer Science*, pages 107–121. Springer, 2014.
- [Lukasiewicz and Straccia, 2008] Thomas Lukasiewicz and Umberto Straccia. Managing uncertainty and vagueness in description logics for the semantic web. *J. Web Sem.*, 6(4):291–308, 2008.
- [Lukasiewicz and Straccia, 2009] Thomas Lukasiewicz and Umberto Straccia. Description logic programs under probabilistic uncertainty and fuzzy vagueness. *International Journal of Approximate Reasoning*, 50(6):837–853, 2009.
- [Lukasiewicz *et al.*, 2012a] Thomas Lukasiewicz, Maria Vanina Martinez, Giorgio Orsi, and Gerardo I. Simari. Heuristic ranking in tightly coupled probabilistic description logics. In Nando de Freitas and Kevin P. Murphy, editors, *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence, Catalina Island, CA, USA, August 14-18, 2012*, pages 554–563. AUAI Press, 2012.
- [Lukasiewicz *et al.*, 2012b] Thomas Lukasiewicz, Maria Vanina Martinez, and Gerardo I. Simari. Inconsistency handling in datalog \pm ontologies. In Luc De Raedt, Christian Bessiere, Didier Dubois, Patrick Doherty, Paolo Frasconi, Fredrik Heintz, and Peter J. F. Lucas, editors, *ECAI 2012 - 20th European Conference on Artificial Intelligence, 2012*, volume 242 of *Frontiers in Artificial Intelligence and Applications*, pages 558–563. IOS Press, 2012.
- [Lukasiewicz *et al.*, 2013] Thomas Lukasiewicz, Maria Vanina Martinez, and Gerardo I. Simari. Complexity of inconsistency-tolerant query answering in datalog \pm . In *Informal Proceedings of the 26th International Workshop on Description Logics, Ulm, Germany, July 23 - 26, 2013*, pages 791–803, 2013.
- [Lukasiewicz, 1999] Thomas Lukasiewicz. Probabilistic deduction with conditional constraints over basic events. *J. Artif. Intell. Res. (JAIR)*, 10:199–241, 1999.
- [Lukasiewicz, 2002] Thomas Lukasiewicz. Probabilistic default reasoning with conditional constraints. *Ann. Math. Artif. Intell.*, 34(1-3):35–88, 2002.
- [Lutz *et al.*, 2013] Carsten Lutz, Inanç Seylan, David Toman, and Frank Wolter. The combined approach to OBDA: taming role hierarchies using filters. In Harith Alani, Lalana Kagal, Achille Fokoue, Paul T. Groth, Chris Biemann, Josiane Xavier Parreira, Lora Aroyo, Natasha F. Noy, Chris Welty, and Krzysztof Janowicz, editors, *The Semantic Web - ISWC 2013 - 12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part I*, volume 8218 of *Lecture Notes in Computer Science*, pages 314–330. Springer, 2013.
- [Ma and Hitzler, 2010] Yue Ma and Pascal Hitzler. Distance-based measures of inconsistency and incoherency for description logics. In Volker Haarslev, David Toman, and Grant E. Weddell, editors, *Proceedings of the 23rd International Workshop on Description Logics (DL 2010), Waterloo, Ontario, Canada, May 4-7, 2010*, volume 573 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2010.

- [Ma *et al.*, 2011] Yue Ma, Guilin Qi, and Pascal Hitzler. Computing inconsistency measure based on paraconsistent semantics. *J. Log. Comput.*, 21(6):1257–1281, 2011.
- [Makinson and Gärdenfors, 1989] David Makinson and Peter Gärdenfors. Relations between the logic of theory change and nonmonotonic logic. In André Fuhrmann and Michael Morreau, editors, *The Logic of Theory Change*, volume 465 of *Lecture Notes in Computer Science*, pages 185–205. Springer, 1989.
- [Martinez *et al.*, 2008] Maria Vanina Martinez, Francesco Parisi, Andrea Pugliese, Gerardo I. Simari, and V. S. Subrahmanian. Inconsistency management policies. In Gerhard Brewka and Jérôme Lang, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the Eleventh International Conference, KR 2008, Sydney, Australia, September 16-19, 2008*, pages 367–377. AAAI Press, 2008.
- [Meyer *et al.*, 2005] Thomas Andreas Meyer, Kevin Lee, and Richard Booth. Knowledge integration for description logics. In Manuela M. Veloso and Subbarao Kambhampati, editors, *Proceedings, The Twentieth National Conference on Artificial Intelligence and the Seventeenth Innovative Applications of Artificial Intelligence Conference, July 9-13, 2005, Pittsburgh, Pennsylvania, USA*, pages 645–650. AAAI Press / The MIT Press, 2005.
- [Mika and Akkermans, 2004] Peter Mika and Hans Akkermans. Towards a new synthesis of ontology technology and knowledge management. *Knowledge Eng. Review*, 19(4):317–345, 2004.
- [Mika *et al.*, 2004] Peter Mika, Victor Iosif, York Sure, and Hans Akkermans. Ontology-based content management in a virtual organization. In Steffen Staab and Rudi Studer, editors, *Handbook on Ontologies*, International Handbooks on Information Systems, pages 455–476. Springer, 2004.
- [Moguillansky and Falappa, 2007] Martín O. Moguillansky and Marcelo A. Falappa. A non-monotonic description logics model for merging terminologies. *Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial*, 11(35):77–88, 2007.
- [Molinaro *et al.*, 2009] Cristian Molinaro, Jan Chomicki, and Jerzy Marcinkowski. Disjunctive databases for representing repairs. *Annals of Mathematics and Artificial Intelligence*, 57(2):103–124, 2009.
- [Mugnier and Thomazo, 2014] Marie-Laure Mugnier and Michaël Thomazo. An introduction to ontology-based query answering with existential rules. In Manolis Koubarakis, Giorgos B. Stamou, Giorgos Stoilos, Ian Horrocks, Phokion G. Kolaitis, Georg Lausen, and Gerhard Weikum, editors, *Reasoning Web. Reasoning on the Web in the Big Data Era - 10th International Summer School 2014, Athens, Greece, September 8-13, 2014. Proceedings*, volume 8714 of *Lecture Notes in Computer Science*, pages 245–278. Springer, 2014.
- [Nebel, 1994] Bernhard Nebel. Base revision operations and schemes: Semantics, representation and complexity. In *ECAI*, pages 341–345, 1994.
- [Nonfjall and Larsen, 1992] H. Nonfjall and H. L. Larsen. Detection of potential inconsistencies in knowledge bases. *International Journal of Intelligent Systems*, 7(2):81–96, 1992.
- [Noy and Musen, 2000] Natalya Fridman Noy and Mark A. Musen. PROMPT: algorithm and tool for automated ontology merging and alignment. In Henry A. Kautz and Bruce W. Porter, editors, *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence, July 30 - August 3, 2000, Austin, Texas, USA.*, pages 450–455. AAAI Press / The MIT Press, 2000.

- [Noy *et al.*, 2006] Natalya Fridman Noy, Abhita Chugh, William Liu, and Mark A. Musen. A framework for ontology evolution in collaborative environments. In Isabel F. Cruz, Stefan Decker, Dean Allemang, Chris Preist, Daniel Schwabe, Peter Mika, Michael Uschold, and Lora Aroyo, editors, *The Semantic Web - ISWC 2006, 5th International Semantic Web Conference, ISWC 2006, Athens, GA, USA, November 5-9, 2006, Proceedings*, volume 4273 of *Lecture Notes in Computer Science*, pages 544–558. Springer, 2006.
- [Pan *et al.*, 2007] Jeff Z. Pan, Giorgos B. Stamou, Giorgos Stoilos, and Edward Thomas. Expressive querying over fuzzy DL-Lite ontologies. In Calvanese *et al.* [2007b].
- [Papini, 1992] O. Papini. A complete revision function in propositional calculus. In *ECAI*, pages 339–343, 1992.
- [Parsons and Hunter, 1998] Simon Parsons and Anthony Hunter. A review of uncertainty handling formalisms. In Anthony Hunter and Simon Parsons, editors, *Applications of Uncertainty Formalisms*, volume 1455 of *Lecture Notes in Computer Science*, pages 8–37. Springer, 1998.
- [Peñaloza and Sertkaya, 2010] Rafael Peñaloza and Baris Sertkaya. On the complexity of axiom pinpointing in the EL family of description logics. In Fangzhen Lin, Ulrike Sattler, and Miroslaw Truszczyński, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the Twelfth International Conference, KR 2010, Toronto, Ontario, Canada, May 9-13, 2010*. AAAI Press, 2010.
- [Pinto *et al.*, 2012] Floriana Di Pinto, Giuseppe De Giacomo, Maurizio Lenzerini, and Riccardo Rosati. Ontology-based data access with dynamic tboxes in DL-Lite. In Hoffmann and Selman [2012].
- [Pivert and Prade, 2010] Olivier Pivert and Henri Prade. Handling dirty databases: From user warning to data cleaning - towards an interactive approach. In Amol Deshpande and Anthony Hunter, editors, *Scalable Uncertainty Management - 4th International Conference, SUM 2010. Proceedings*, volume 6379 of *Lecture Notes in Computer Science*, pages 292–305. Springer, 2010.
- [Plessers *et al.*, 2007] Peter Plessers, Olga De Troyer, and Sven Casteleyn. Understanding ontology evolution: A change detection approach. *J. Web Sem.*, 5(1):39–49, 2007.
- [Poggi *et al.*, 2008] Antonella Poggi, Domenico Lembo, Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, and Riccardo Rosati. Linking data to ontologies. In *J. Data Semantics* [2008], pages 133–173.
- [Qi and Du, 2009] Guilin Qi and Jianfeng Du. Model-based revision operators for terminologies in description logics. In Boutilier [2009], pages 891–897.
- [Qi and Hunter, 2007] Guilin Qi and Anthony Hunter. Measuring incoherence in description logic-based ontologies. In Karl Aberer, Key-Sun Choi, Natasha Fridman Noy, Dean Allemang, Kyung-Il Lee, Lyndon J. B. Nixon, Jennifer Golbeck, Peter Mika, Diana Maynard, Riichiro Mizoguchi, Guus Schreiber, and Philippe Cudré-Mauroux, editors, *The Semantic Web, 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference, ISWC 2007 + ASWC 2007, Busan, Korea, November 11-15, 2007.*, volume 4825 of *Lecture Notes in Computer Science*, pages 381–394. Springer, 2007.
- [Qi and Pan, 2007] Guilin Qi and J Pan. A stratification-based approach for inconsistency handling in description logics. In *IWOD'07*, page 83, 2007.
- [Qi and Yang, 2008] Guilin Qi and Fangkai Yang. A survey of revision approaches in description logics. In Franz Baader, Carsten Lutz, and Boris Motik, editors, *Description Logics*, volume 353 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2008.

- [Qi *et al.*, 2006a] G. Qi, W. Liu, and D. A. Bell. A revision-based approach to handling inconsistency in description logics. *Artif. Intell. Rev.*, 26(1-2):115–128, 2006.
- [Qi *et al.*, 2006b] Guilin Qi, Weiru Liu, and David A. Bell. Knowledge base revision in description logics. In *Proc. of JELIA 2006*, volume 4160 of *Lecture Notes in Computer Science*, pages 386–398. Springer, 2006.
- [Qi *et al.*, 2006c] Guilin Qi, Weiru Liu, and David A. Bell. Merging stratified knowledge bases under constraints. In *AAAI [2006]*, pages 281–286.
- [Qi *et al.*, 2007a] Guilin Qi, Jeff Z. Pan, and Q. Ji. A possibilistic extension of description logics. In *Proceedings of the 2007 International Workshop on Description Logics (DL2007)*, volume 4724 of *Description Logics*, pages 602–607. CEUR-WS.org / CEUR Workshop Proceedings, 2007.
- [Qi *et al.*, 2007b] Guilin Qi, Jeff Z. Pan, Jeff Z. Pan, and Q. Ji. Extending description logics with uncertainty reasoning in possibilistic logic. In *Symbolic and Quantitative Approaches to Reasoning with Uncertainty, 9th European Conference (ECSQARU 2007)*, volume 4724 of *Lecture Notes in Computer Science*, pages 828–839. Springer, 2007.
- [Qi *et al.*, 2008a] Guilin Qi, Peter Haase, Zhisheng Huang, Qiu Ji, Jeff Z. Pan, and Johanna Völker. A kernel revision operator for terminologies - algorithms and evaluation. In Amit P. Sheth, Steffen Staab, Mike Dean, Massimo Paolucci, Diana Maynard, Timothy W. Finin, and Krishnaprasad Thirunarayan, editors, *International Semantic Web Conference*, volume 5318 of *Lecture Notes in Computer Science*, pages 419–434. Springer, 2008.
- [Qi *et al.*, 2008b] Guilin Qi, Jeff Z. Pan, and Jeff Z. Pan. A tableau algorithm for possibilistic description logic. In *The Semantic Web, 3rd Asian Semantic Web Conference, ASWC 2008*, volume 5367 of *Lecture Notes in Computer Science*, pages 61–75. Springer, 2008.
- [Qi *et al.*, 2010a] Guilin Qi, Qiu Ji, Jeff Z. Pan, and Jianfeng Du. Possdl - A possibilistic DL reasoner for uncertainty reasoning and inconsistency handling. In Lora Aroyo, Grigoris Antoniou, Eero Hyvönen, Annette ten Teije, Heiner Stuckenschmidt, Liliana Cabral, and Tania Tudorache, editors, *The Semantic Web: Research and Applications, 7th Extended Semantic Web Conference, ESWC 2010, Heraklion, Crete, Greece, May 30 - June 3, 2010, Proceedings, Part II*, volume 6089 of *Lecture Notes in Computer Science*, pages 416–420. Springer, 2010.
- [Qi *et al.*, 2010b] Guilin Qi, Weiru Liu, and David A. Bell. A comparison of merging operators in possibilistic logic. In Yaxin Bi and Mary-Anne Williams, editors, *KSEM*, volume 6291 of *Lecture Notes in Computer Science*, pages 39–50. Springer, 2010.
- [Qi *et al.*, 2011] Guilin Qi, Jeff Z. Pan, Q. Ji, Jeff Z. Pan, and J. Du. Extending description logics with uncertainty reasoning in possibilistic logic. *Int. J. Intell. Syst.*, 26(4):353–381, 2011.
- [Reiter, 1987] R. Reiter. A theory of diagnosis from first principles. *Artif. Intell.*, 32(1):57–95, 1987.
- [Rescher and Manor, 1970] Nicholas Rescher and Ruth Manor. On inference from inconsistent premisses. *Theory and Decision*, 1(2):179–217, 1970.
- [Ribeiro and Wassermann, 2007] M. Ribeiro and R. Wassermann. Base revision in description logics — preliminary results. In *Int. Workshop on Ontology Dynamics (IWOD-07)*, Innsbruck, Austria, 2007.
- [Rosati, 2014] Riccardo Rosati. Tractable approaches to consistent query answering in ontology-based-data access. In Laura Giordano, Valentina Gliozzi, and Gian Luca Pozzato, editors, *Proceedings of the 29th Italian Conference on Computational Logic, Torino, Italy, June 16-18, 2014.*, volume 1195 of *CEUR Workshop Proceedings*, page 12. CEUR-WS.org, 2014.

- [Rossi, 2013] Francesca Rossi, editor. *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, August 3-9, 2013*. IJCAI/AAAI, 2013.
- [Satoh, 1988] Ken Satoh. Nonmonotonic reasoning by minimal belief revision. In *FGCS*, pages 455–462, 1988.
- [Schlobach and Cornet, 2003] Stefan Schlobach and Ronald Cornet. Non-standard reasoning services for the debugging of description logic terminologies. In Georg Gottlob and Toby Walsh, editors, *IJCAI-03, Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, Acapulco, Mexico, August 9-15, 2003*, pages 355–362. Morgan Kaufmann, 2003.
- [Shadbolt *et al.*, 2006] Nigel Shadbolt, Tim Berners-Lee, and Wendy Hall. The semantic web revisited. *IEEE Intelligent Systems*, 21(3):96–101, 2006.
- [Shafer, 1976] Glenn Shafer, editor. *A Mathematical Theory of Evidence*. Princeton University Press, 1976.
- [Shvaiko and Euzenat, 2013] Pavel Shvaiko and Jérôme Euzenat. Ontology matching: State of the art and future challenges. *IEEE Trans. Knowl. Data Eng.*, 25(1):158–176, 2013.
- [Sowa, 1987] John F. Sowa. Semantic networks. <http://www.jfsowa.com/pubs/semnet.htm>, 1987. Encyclopedia of Artificial Intelligence.
- [Spaccapietra, 2008] Stefano Spaccapietra, editor. *Journal on Data Semantics X*, volume 4900 of *Lecture Notes in Computer Science*. Springer, 2008.
- [Spohn, 1988] Wolfgang Spohn. A general non-probabilistic theory of inductive reasoning. In Ross D. Shachter, Tod S. Levitt, Laveen N. Kanal, and John F. Lemmer, editors, *UAI '88: Proceedings of the Fourth Annual Conference on Uncertainty in Artificial Intelligence, Minneapolis, MN, USA, July 10-12, 1988*, pages 149–158. North-Holland, 1988.
- [Staworko *et al.*, 2012] Slawek Staworko, Jan Chomicki, and Jerzy Marcinkowski. Prioritized repairing and consistent query answering in relational databases. *Ann. Math. Artif. Intell.*, 64(2-3):209–246, 2012.
- [Straccia, 1998] Umberto Straccia. A fuzzy description logic. In Jack Mostow and Chuck Rich, editors, *Proceedings of the Fifteenth National Conference on Artificial Intelligence and Tenth Innovative Applications of Artificial Intelligence Conference, AAAI 98, IAAI 98, July 26-30, 1998, Madison, Wisconsin, USA.*, pages 594–599. AAAI Press / The MIT Press, 1998.
- [Straccia, 2001] Umberto Straccia. Reasoning within fuzzy description logics. *J. Artif. Intell. Res. (JAIR)*, 14:137–166, 2001.
- [Straccia, 2006a] Umberto Straccia. Answering vague queries in fuzzy DL-Lite. In *Proceedings of the 11th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, (IPMU-06)*, pages 2238–2245, 2006.
- [Straccia, 2006b] Umberto Straccia. Towards top-k query answering in description logics: The case of DL-Lite. In M. Fisher, W. van der Hoek, B. Konev, and Alexei Lisitsa, editors, *Logics in Artificial Intelligence, 10th European Conference, JELIA 2006, Liverpool, UK, September 13-15, 2006, Proceedings*, volume 4160 of *LNCS*, pages 439–451. Springer, 2006.
- [Straccia, 2012] Umberto Straccia. Top-k retrieval for ontology mediated access to relational databases. *Inf. Sci.*, 198:1–23, 2012.

- [Straccia, 2013] Umberto Straccia. *Foundations of Fuzzy Logic and Semantic Web Languages*. CRC Studies in Informatics Series. Chapman & Hall, 2013.
- [Torniai *et al.*, 2008] Carlo Torniai, Jelena Jovanovic, Dragan Gasevic, Scott Bateman, and Marek Hatala. E-learning meets the social semantic web. In *The 8th IEEE International Conference on Advanced Learning Technologies, ICALT 2008, July 1st- July 5th, 2008, Santander, Cantabria, Spain*, pages 389–393. IEEE, 2008.
- [Valiant, 1979] Leslie G. Valiant. The complexity of computing the permanent. *Theor. Comput. Sci.*, 8:189–201, 1979.
- [van der Gaag, 2013] Linda C. van der Gaag, editor. *Symbolic and Quantitative Approaches to Reasoning with Uncertainty - 12th European Conference, ECSQARU 2013, Utrecht, The Netherlands, July 8-10, 2013. Proceedings*, volume 7958 of *Lecture Notes in Computer Science*. Springer, 2013.
- [W3C, 2008] W3C. Owl 2 web ontology language document overview (second edition). <http://www.w3.org/2012/pdf/REC-owl2-overview-20121211.pdf>, 2008. [Online; W3C Recommendation 11 December 2012].
- [Wang *et al.*, 2010] Zhe Wang, Kewen Wang, and Rodney W. Topor. A new approach to knowledge base revision in DL-Lite. In Maria Fox and David Poole, editors, *AAAI*. AAAI Press, 2010.
- [Wilkerson *et al.*, 1989] R. W. Wilkerson, R. Greiner, and B. A. Smith. A correction to the algorithm in reiter’s theory of diagnosis. *Artificial Intelligence*, 41:79–88, 1989.
- [Würbel *et al.*, 2000] Eric Würbel, Robert Jeansoulin, and Odile Papini. Revision: an application in the framework of GIS. In A. G. Cohn, F. Giunchiglia, and B. Selman, editors, *KR*, pages 505–515. Morgan Kaufmann, 2000.
- [Yager, 1992] Ronald R. Yager. On the specificity of a possibility distribution. *Fuzzy Sets and Systems*, 50(3):279 – 292, 1992.
- [Zadeh, 1965] Lotfi A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, 1965.
- [Zadeh, 1978] L.A Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, 1(1):3–28, 1978.
- [Zadeh, 1988] Lotfi A. Zadeh. Fuzzy logic. *IEEE Computer*, 21(4):83–93, 1988.
- [Zadeh, 1999] L.A. Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, 100, Supplement 1(0):9 – 34, 1999.
- [Zhang and Lin, 2013] Xiaowang Zhang and Zuoquan Lin. An argumentation framework for description logic ontology reasoning and management. *Journal of Intelligent Information Systems*, 40(3):375–403, 2013.
- [Zhou *et al.*, 2009] Liping Zhou, Houkuan Huang, Guilin Qi, Yue Ma, Zhisheng Huang, and Youli Qu. Measuring inconsistency in DL-Lite ontologies. In *Web Intelligence*, pages 349–356. IEEE, 2009.
- [Zhou *et al.*, 2012] Liping Zhou, Houkuan Huang, Guilin Qi, Yue Ma, Zhisheng Huang, and Youli Qu. Paraconsistent query answering over DL-Lite ontologies. *Web Intelligence and Agent Systems*, 10(1):19–31, 2012.

- [Zhu *et al.*, 2013] Jinfan Zhu, Guilin Qi, and Boontawee Suntisrivaraporn. Tableaux algorithms for expressive possibilistic description logics. In *2013 IEEE/WIC/ACM International Conferences on Web Intelligence, WI 2013, Atlanta, GA, USA, November 17-20, 2013*, pages 227–232. IEEE Computer Society, 2013.
- [Zhuang *et al.*, 2014] Zhiqiang Zhuang, Zhe Wang, Kewen Wang, and Guilin Qi. Contraction and revision over DL-Lite tboxes. In Brodley and Stone [2014], pages 1149–1156.