COMPUTING SPACES OF INDEPENDENT EXPLANATIONS VIA ABDUCTIVE REASONING

A THESIS SUBMITTED TO THE UNIVERSITY OF MANCHESTER
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
IN THE FACULTY OF SCIENCE AND ENGINEERING

2022

Warren Del-Pinto

School of Computer Science

Contents

Word Count: 69066

List of Tables

List of Figures

Abstract

COMPUTING SPACES OF INDEPENDENT EXPLANATIONS VIA ABDUCTIVE REASONING
Warren Del-Pinto
A thesis submitted to the University of Manchester
for the degree of Doctor of Philosophy, 2022

This thesis investigates methods for abductive reasoning in large knowledge bases. Abduction refers to the process of explaining new observations using prior knowledge, which enables tasks that require the generation of new hypotheses including scientific discovery, belief expansion, diagnostics, language interpretation and inductive learning. This thesis focuses on knowledge represented using Description Logics (DLs), which are commonly used to model information in domains such as bioinformatics, healthcare, robotics and natural language processing. A variety of research has been conducted on abduction in DLs, though it remains a hard problem.

In this work, the aim is to produce hypotheses that take the form of a set of explanations and are semantically minimal. Producing a set of explanations, rather than a single one, provides a way to examine multiple avenues of explaining the observation, providing further insight into both the observation and the available knowledge. Semantic minimality limits hypotheses to those that assume no more than is necessary to explain the observation given the existing knowledge. For the general application of abduction, it is natural to first seek explanations that are likely, but limit the strength of initial assumptions until further evidence is available. This provides a useful mechanism for ordering hypotheses, seeking the least assumptive (weakest) ones first, which can then be refined through further investigation.

However, semantic minimality is problematic in the presence of disjunction as it permits any number of redundant explanations (disjuncts). Therefore, disjunction has previously been excluded from solutions when considering semantic minimality. In this work, this problem must be addressed as the hypothesis takes the form of a set of possible explanations, represented as a disjunction. Therefore, a new DL abduction

problem is defined. The proposed problem introduces a notion of independence, where explanations must not contradict existing knowledge and must not express information that is contained within the other explanations. This problem is the first to consider both semantic minimality and independence of explanations together in the DL setting. The issue of permitting language extensions in the hypotheses is also discussed and motivated, as this makes the abduction problem considered here significantly different to prior work. An example of this is disjunctions of DL axioms used to represent the hypotheses.

To solve the problem, novel methods that utilise the connection between forgetting and abduction are proposed. The need for further investigation into this connection in the DL setting is addressed, including investigation of the characteristics of forgetting solutions in relation to the proposed problem and the development of efficient methods for eliminating redundant explanations. Extensions to existing forgetting tools required for expressive abduction are also proposed and implemented. The forgetting-based abduction approaches developed are evaluated over corpora containing real ontologies. The results indicate that in the majority of cases, the approaches can efficiently compute spaces of explanations over $\mathcal{ALC}$ knowledge bases with tens of thousands of axioms. The use of the disjunctive hypotheses produced by forgetting-based abduction approaches is also explored with respect to the problems of hypothesis refinement, induction and concept learning in ontologies, to provide suggestions on how the characteristics of these hypotheses may be utilised in practice.

Declaration

No portion of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

Copyright

Acknowledgements

I would like to thank my supervisor Renate A. Schmidt for her constant help and guidance throughout the course of this PhD. The numerous lengthy discussions and encouragement have made the journey towards and completion of this thesis not only possible, but also enjoyable.

I would also like to thank my office colleagues, collaborators, university staff and others with whom I have had numerous pleasant, interesting and insightful discussions over the years. The conversations we have engaged in kept me sane during even the most stressful periods of my research. While there are too many to list, among these are Patrick Koopmann, Yizheng Zhao and all my colleagues in the Formal Methods group who have provided me with many insightful debates and discussions regarding topics in our field of research and academic life in general.

My thanks are also extended to the many friends I have made both during and before embarking on this PhD, including Sam, Zico, Toby, Matt, Jordan, Steph, Pauline, Ruben and many others who have managed to make life outside work entertaining when needed especially during the last year.

For their endless encouragement, advice and patience over the last few years, I would like to thank my partner Jennifer Goossens, and my parents Carlino and Bridget Del-Pinto. I am eternally grateful for all the times you have been there for me and this thesis would not be possible without you. To my grandmother Bridget Moriarty, and my late grandfather Michael Moriarty, I extend my thanks for creating an incredible family for whom I am very grateful. To all of those in the Moriarty, Del-Pinto and Goossens families who have supported me over the years, I cannot thank you enough.

Chapter 1

Introduction

Abductive, inductive and deductive reasoning are fundamental forms of reasoning, each of which encapsulates a process distinct from the others. The characterisation of these three forms of reasoning, and particularly the identification of abduction as a third form of reasoning alongside deduction and induction, was a key element in the work of Charles Sanders Peirce [Pei78]. Notions related to abductive reasoning date back further, with methods of argument such as the Greek apagoge in the works of Aristotle.

While deduction and induction are widely known, the role of abduction is less widely understood. Often, abductive reasoning is not explicitly identified when utilised alongside other forms of reasoning, where its role is instead assumed as an implicit part of the process of inductive reasoning [Pop14]. The two forms of reasoning do share similarities. Both abduction and induction are ampliative: they provide mechanisms by which new knowledge can be obtained that goes beyond what is already known. As a result, another similarity between abduction and induction is that they are not necessarily truth preserving: a false conclusion can be inferred from entirely true premises. The results of abduction and induction are intended to be likely under the given circumstances, but are not guaranteed to be correct. It is in these aspects that abduction and induction differ from deduction. Deductive reasoning enables the process of inferring implicit knowledge that can be derived from pre-existing, explicit facts, without going beyond the existing knowledge. It is by nature truth-preserving: given that the premises of an inference are true, the conclusion must also be true.

However, despite these similarities, abduction and induction emphasise two distinct notions. Abduction concerns *explanation* of a specific set of observations, producing candidate explanations for the given observations with respect to some existing knowledge. Meanwhile, inductive reasoning is instead related to the notion of *generalisation*. Given a set of observations, the aim of induction is to produce a general rule that covers the entire population from which the observations were drawn. In this sense, abduction can be seen as a form of hypothesis generation, while induction can be viewed as hypothesis evaluation [FK00b].

Just as deduction and induction are utilised as core mechanisms in Artificial Intelligence (AI), in areas such as Automated Reasoning and Machine Learning, abduction too plays a key role in a variety of tasks which require the generation of explanations for given observations. Consequently, abductive reasoning has become a topic of recurring interest in AI. The identification and investigation of notions of best or preferred hypotheses is core to the study and application of abductive reasoning in practice, and abduction is often referred to as inference to the best explanation as a result. The development of methods to compute abductive hypotheses became the subject of a wide variety of work in AI, spanning work in areas such as theorem proving [Pop73] and abductive logic programming [KKT92]. Across AI, applications of abductive reasoning are varied and include tasks such as natural language interpretation [HSAM93, Sti91] and inference [RNM05], automating the process of scientific inquiry [KWJ$^+$04, Ray07] and the extension and repair of large knowledge bases [EKS06]. A variety of work has also been performed on the integration of abduction and induction [FK00a] in areas such as machine learning [Moo00], abductive logic programming [Ray09] and statistical relational AI [RM10].

1.1 Motivation

As discussed, the importance of abductive reasoning has been recognised in many areas of AI research. This is also true in the area of Knowledge Representation and Reasoning, a major branch of which makes use of ontologies.

The term ontology has roots in philosophy, where it broadly encompasses the study of notions of existence. In computer science and AI, an ontology can generally be seen as a specification of the collection of entities and relationships between entities related to a given domain [Gru95]. Often, ontologies are referred to as knowledge bases[1] and generalise notions such as Knowledge Graphs to include both an upper-level schema of general entities and more specific, instance level data.

Description Logics (DLs) provide expressive languages which can be used to represent knowledge contained within an ontology. These DL languages enable the formal basis to capture the semantics of knowledge in a way that enables expressive representations of complex relationships while maintaining the ability to efficiently make inferences on existing knowledge. These capabilities have led to a widespread use of DLs in representing ontologies in a variety of fields, prominent examples of which include the SNOMED-CT clinical terminology [SPSW01] used internationally in healthcare, the Gene Ontology [Con04] in bioinformatics which aims to represent genetic information across a wide variety of species for use and the IEEE Standard Ontology for Robotics and Automation [SPM$^+$12].

The majority of reasoning services currently provided for DL ontologies are deductive in nature. That is, they provide mechanisms by which implicit relationships can be inferred from background knowledge. However, many tasks require a mechanism for the generation of new knowledge that goes beyond what is already explicitly or implicitly known. Such tasks include hypothesis generation, diagnostics and belief expansion. To provide support for these problems, it is necessary to look to alternative

[1]In some domains the terms ontology and knowledge base differ in their consideration of what is assumed to be present. For example, in some cases ontology may refer to an upper level terminology consisting of universally quantified statements, without instance-level data, i.e., ground assertions. In this thesis, the terms KB and ontology are used interchangeably.

reasoning mechanisms such as abduction. The need for such a mechanism to address corresponding problems in the domain of DL ontologies has been identified [EKS06]. This has led to the application of abduction in DLs to problems such as matchmaking [CDNDS^{+}05, DNDSD07], negative query answering [COSS13] and ontology repair [LWKDI13, WKDL14].

Despite the longstanding interest in abductive reasoning in the wider field of AI, abductive reasoning in DLs is a relatively new problem. Providing abductive reasoning capability in the context of DL ontologies is the focus of this thesis.

1.2 Challenges

There exist many definitions of the abduction problem, each differing in the notion of what constitutes an acceptable or "best" hypothesis. The decision as to how to formulate an appropriate abduction problem therefore depends upon a multitude of factors, including the domain of application and the intended use case, each of which may require a different set of constraints to be applied to the hypotheses produced. There is also the issue of complexity: abductive reasoning has been identified as a difficult problem throughout its history as a topic of interest in AI. This also holds true for logic-based abduction. In propositional logic, complexity studies have been conducted that indicate that the task of abduction is more difficult, in terms of computational cost, than deduction [EG95]. This difficulty naturally applies also to the domain of DL ontologies, where further complexity analyses have been performed, focusing particularly on lightweight DLs such as the $\mathcal{EL}$ family [Bie08]. In general, the complexity of abductive reasoning is highly dependent on the constraints imposed upon the hypotheses produced: certain preference relations between hypotheses increase the complexity more significantly than others.

The difficulty of the problem also depends heavily on the size of the input. While DLs are designed to be decidable, ontologies are often large. As a result, the difficulty of abductive reasoning is amplified in the setting of DL ontologies, since the space of possible explanations can become intractably large, possibly infinite, as the size of the background knowledge increases. This is problematic considering that efficiency and tractability are core considerations when utilising DL ontologies. Therefore, any useful abductive reasoning system developed for this domain should also attempt to satisfy these requirements as far as possible. Further, in many cases, for a hypothesis to be useful in this domain it must also be interpretable by a user, otherwise any insight that could be derived from the explanations produced would be lost. As a result, different considerations are required when attempting to produce a solution to abductive problems than in settings which assume that the background theory is small, or those in which redundant explanations are less problematic. Narrowing the hypothesis space to avoid computing a potentially infinite number of solutions, while simultaneously prioritising useful hypotheses, is a difficult task.

Given this difficulty it becomes less surprising that, despite the clear and identified need for abductive reasoning capabilities in the area, there is a lack of practical systems for performing abductive reasoning in large ontologies. So far, several works have presented methods for addressing different instantiations of the problems of TBox abduction [WKDL14, HBK14, DWM17] and ABox abduction [KES11, HB12, DWS14, PH17]. Despite this, existing approaches tend to be purely theoretical in nature [KES11], scale only to small, example ontologies [PH18, MPH18] or address the problem in specific Horn fragments of DLs [DWS14].

Few existing works on abduction consider the problem of computing a hypothesis that makes the least assumptions necessary to entail the given observation under the available knowledge. This notion is quite natural considering the role of abduction as providing initial, "likely" explanations, and has been of interest in several applications including natural language interpretation [Sti91] and diagnostics [PH17]. In the DL literature, the notion of a least assumptive hypothesis is referred to as *semantic minimality* [KES11, HB12, HBK14]. However, despite this interest practical solutions to computing semantically minimal hypotheses in DLs have yet to manifest. The lack of work on this form of abduction is likely due to the computational complexity of finding these solutions: it is not possible to simply compute all possible hypotheses then perform entailment checks between them. Such generate-and-test approaches would be intractable in practice, particularly in more expressive DLs such as $\mathcal{ALC}$ where the complexity of entailment checking is at least exponential in the size of the input. In addition, existing works on semantically minimal abduction restrict the problem to exclude disjunction in the hypotheses produced [KES11, HB12, HBK14]. This is partly due to further increasing the difficulty, but is also due to an insufficiency in the definition of semantic minimality when considering logics that permit disjunction. Effectively, the set of acceptable hypotheses can contain hypotheses that consist of the expected disjuncts plus any arbitrary number of redundant disjuncts joined to the rest of the hypothesis. Eliminating these redundancies is in itself a challenging problem, but in the presence of large ontologies it can quickly become impossible to identify the redundancies contained within a hypothesis, reducing the interpretability of the solution returned.

A common feature of abduction problems in fields such as abductive logic programming [KKT92] is the ability to specify a set of allowed symbols, called abducibles, which restricts hypotheses to a subset of the vocabulary used to express a knowledge base. In contrast, in the DL setting many approaches do not enable the specification of abducibles. However, abducibles are a useful way to constrain the search space for hypotheses, while providing a user the ability to seek hypotheses that are of interest based on a subset of the domain spanned by a given knowledge base.

From this, it is clear that further work is needed on abductive reasoning in DLs, particularly when computing least assumptive hypotheses. A promising avenue of investigation lies in the connection between abductive reasoning and another non-standard reasoning task: *forgetting*. The application of forgetting to abductive reasoning has been proposed in a general setting [DLS01], but has remained undeveloped.

1.3 Contributions

In this thesis, the aim is to investigate the problem of abduction and develop practical hypothesis generation capabilities for different forms of DL abduction. To this end, the use of forgetting methods for abductive reasoning is investigated in the setting of DL ontologies, and the connection between abductive reasoning and forgetting in this domain is developed. To evaluate the effectiveness of these approaches in practice, existing state-of-the-art forgetting tools in DLs were investigated and extended.

The choice to investigate forgetting for abduction is based on the connection between forgetting solutions, strongest necessary and weakest sufficient conditions. By utilising forgetting together with contraposition, the weakest sufficient condition for a given abduction problem can be obtained, which corresponds to a notion of semantic minimality. Additionally, efficient forgetting approaches have been developed for DLs in recent years, which handle expressive DL languages and perform well over large, real world ontologies. Considering these factors, a forgetting based approach to abductive reasoning appears promising.

Previous work has identified the connection between forgetting and abduction in the context of small theories in classical logics [DLS01, GSS08], but has not addressed the problem of redundant explanations. Since a hypothesis consisting of mostly redundant explanations is of little use in practice, the notion of redundancy between disjunctive explanations must be addressed. Solving this problem is an essential step to ensuring that the abduction approach yields hypotheses that are interpretable and useful in practice in the setting of DL ontologies, since the number of redundancies in the hypotheses produced tends to be very large in the presence of a large amount of background knowledge. Similarly, the notion of semantic minimality as established so far in the work on abduction in DLs does not address abduction settings that permit disjunction in the hypothesis.

The main contributions of this thesis are as follows:

- A new abduction problem is proposed for the setting of DL ontologies, which extends the condition of semantic minimality to account for disjunction in the hypothesis by viewing each disjunct as a single explanation for the observation and considering independence between each of these explanations. Therefore, in this perspective the aim is not to compute a single explanation for the given observation, as is common in prior work on abduction, but is instead to compute a hypothesis that is a representation of the entire space of possible explanations for the observation.

- Methods for producing the desired hypotheses are presented, developing the connection between forgetting and abduction for the problem of ABox abduction. To this end, several state-of-the-art forgetting approaches for DLs are investigated and compared for use in abduction systems. The problem of eliminating redundant explanations from the hypotheses is identified and solved via two proposed approaches: a low-cost approximation-based approach and an extensive approach that eliminates all redundancies by combining the approximation-based approach with entailment checking using an external reasoner. The result is the first practical approach for producing hypotheses that take the form of a space of independent explanations for a given observation in the DL setting.

- Extensions to existing forgetting methods that are necessary for their application to abductive reasoning are identified for both ABox and TBox abduction. These are then addressed and solved by extending the forgetting calculus. By extending the abduction problem definition and notions of redundancy, the resulting calculus is then utilised to solve the problem of *Knowledge Base Abduction*, which generalises both ABox and TBox abduction to explain observations and produce hypotheses that can contain both universally quantified and ground statements. Knowledge Base abduction is the most general form of abduction identified for DLs [EKS06]. The result is the first approach to address and completely solve the Knowledge Base Abduction problem, which produces hypotheses as a set of independent knowledge bases that explain the new observation.

- The presented abduction approaches are all evaluated via a set of experiments performed over several corpora consisting of real ontologies in active use, primarily in the biomedical domain. The experimental evaluations presented in this thesis constitute one of the few experimental evaluations of abduction in the domain of DL ontologies. To the authors' knowledge, this is the first extensive experimental evaluation of an abduction system for DL ontologies that aims to compute semantically minimal, i.e., least assumptive hypotheses, particularly in the presence of disjunction. The results support the practicality of the approaches presented in this thesis for all three of the forms of DL abduction investigated and highlight the importance of eliminating redundant explanations when utilising approaches of this kind.

- The use of the hypotheses produced by the abduction approaches in this work is discussed with respect to several reasoning tasks in the area of DLs. Since hypotheses as spaces of independent explanations have yet to be investigated in DLs, promising directions for utilising the hypotheses produced by the abduction approaches in this thesis are proposed. Directions include the problems of hypothesis refinement, inductive reasoning and ontology learning.

This work provides the first investigation into developing practical signature-based approaches to ABox abduction, TBox abduction and Knowledge Base abduction in DLs, resulting in the first solution that computes semantically minimal hypotheses in the presence of disjunction.

1.4 Chapter Overview and Published Results

Chapter 2 covers the basic notation and definitions relevant to Description Logic ontologies, which is the setting of this work. Chapter 3 focuses on summarising relevant background on abductive reasoning, with a focus on relevant definitions on abductive reasoning in DLs and the task of forgetting, which are core to the abduction approaches presented in this work.

Chapter 4 focuses on the abduction problem that is identified and solved in this thesis. The key constraints applied to the abductive solutions are discussed and motivation is provided for the choices made in tackling the defined problem. Therefore, Chapter 4 provides context and a generalised basis for the work in subsequent chapters, which aim to solve the problem.

Chapter 5 presents an approach to performing ABox abduction in large description logic ontologies expressed in the language of $\mathcal{ALC}$. The focus is on developing a practical method for computing hypotheses as semantically minimal spaces of independent explanations, including efficient methods for solving the problem of redundancy in the hypothesis. Proofs of soundness and completeness and experimental evaluations are provided. A short version of this material is published at the Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19) [DS19a]. Preliminary ideas for this approach were presented at the First Workshop on Second-Order Quantifier Elimination and Related Topics [DS17].

Chapter 6 investigates the ABox abduction problem in a more expressive setting, with the aim of extending the space of solutions that can be reached for the ABox abduction problem in $\mathcal{ALC}$. Specifically, the focus is on investigating the integration of a different form of forgetting to the abduction approach, namely semantic forgetting, and comparing the hypotheses produced to those obtained using the approach in Chapter 5. This investigation was first described in a paper that was published in the proceedings of the International Symposium on Frontiers of Combining Systems (FroCoS 2019) [DS19b].

Chapter 7 extends the scope of the abduction problem and the abduction approach presented in Chapter 5. Extensions that must be made to both the proposed abduction problem and the forgetting calculus used are identified and developed, with the aim of solving a more expressive form of ABox abduction and the problem of TBox abduction. By lifting these results, this is then followed by a description of a system that solves the Knowledge Base (KB) abduction problem, which generalises the problems of ABox and TBox abduction. This includes the extended forgetting calculus, a new approach to applying the inferences in the calculus to avoid redundant explanations and proofs of the main properties of the approach including soundness and completeness. Discussions of and solutions for the problem of eliminating redundant explanations in this more expressive setting are presented, together with an experimental results, making use of a newly developed framework for performing experiments for abductive reasoning in DLs. The work in Chapter 7, particularly Sections 7.3, 7.4 and the Appendix A.1, was the result of a collaboration and originally presented in a paper published at the Seventeenth International Conference on Principles of Knowledge Representation and Reasoning (KR2020) [KDTS20]. New compared to [KDTS20] are additional discussions and proposals for combining the filtering approaches in earlier chapters with the approach in Chapter 7 in Section 7.5, while the results of the experimental evaluation in Section 7.6 and the approach used to obtain them are new to this thesis and hence different from those presented in the paper.

Chapter 8 provides perspectives and an initial investigation into the use of the signature-based abduction approaches developed in this work for applications related to inductive reasoning and learning in DL ontologies. The connection between KB abduction and problems such as explanatory induction are discussed, and examples are provided to motivate the use of the hypotheses produced by the methods in this thesis in inductive problems. The problem of selecting an appropriate forgetting signature and hypothesis refinement are also discussed. The work in this chapter builds on ideas presented at the Automated Reasoning Workshop [DS18] and work presented at the Deduction Beyond Satisfiability seminar (19371) as part of the Dagstuhl Seminar series [Del19].

Chapter 2

Basics of Description Logics

This thesis focuses primarily on abductive reasoning in a particular setting: large knowledge bases, often referred to as ontologies, expressed in formal languages called Description Logics (DLs). To provide context to this domain, this Chapter gives an overview of the basics of the description logic languages considered in this work. For the purpose of this chapter, some knowledge of first-order logic and set theory is assumed. Description logics are decidable fragments of first-order logic that are widely used to represent domain knowledge in the form of ontologies. This knowledge is represented by the use of three sets of symbols: *concepts*, *roles* and *individuals*. These three can be described as follows: concepts group sets of individuals under a common entity, roles are relations between two individuals, and the individuals themselves are constants, i.e., a single instance. Recently, description logics have been used to provide the basis of the current version of the Web Ontology Language, OWL 2, of the World Wide Web Consortium (W3C). As a result, constructors in OWL are closely related to the concept, role and individual symbols discussed above, where concept and role symbols are instead referred to as *classes* and *properties* respectively. In this thesis, the DL naming conventions will be used. The exact relationship between OWL and DLs is not covered here, since the syntax and semantics of DLs is sufficient to discuss the topics of this thesis; further details on OWL can be found in [HKP$^+$09].

2.1 Description Logics

Description Logics provide expressive languages with which complex specialist knowledge can be represented, while retaining the possibility of developing efficient reasoning procedures to derive further insight from the available knowledge. As such, one of the most common formalisms used to express ontologies are DLs, which provide the semantic underpinning behind the Web Ontology Language (OWL), a family of languages which are widely used in creating ontologies for knowledge representation. An ontology expressed in the OWL-DL sublanguage directly corresponds to a DL knowledge base [Hor05], and the formalism underlying the successor to OWL, OWL2 [GHM$^+$08], is the DL language $\mathcal{SROIQ}(D)$ [HKS06]. DL ontologies are split into two main parts. The first is a TBox containing information regarding general entities called *concepts* and relations between these entities called *roles*. The second is called the ABox, which contains assertions regarding specific instances of these concepts, called *individuals*, and the relationships that exist between them. The primary emphasis behind the use of DLs is practicality: common reasoning problems such as satisfiability and subsumption are decidable and it is possible to develop systems that solve these problems efficiently in practice [Hor05].

Given these benefits, DL ontologies are utilised in a wide variety of fields, where they address the need to represent domain knowledge in a way that is machine-readable, clear and consistent while providing the ability to reason about entities and relations present within a given domain. Examples of areas in which ontologies have seen widespread application include the Semantic Web, AI, computational-linguistics, bioinformatics, medical informatics and robotics. The benefit of using DL ontologies is particularly pronounced when there is a large amount of knowledge to be modelled and the complexity of the given knowledge increases. For example, increased adoption of the expressivity provided by DL languages underpinning OWL has been motivated for the clinical terminology SNOMED-CT [SPSW01], which is used globally to model clinical information essential to the provision of healthcare. Key motivations for this adoption include improving the representations of clinical context, flexible handling of definitions for entities and the ability to better represent complex relations between entities in different parts of the terminology [RB08]. Other examples of ontologies in use in industry include the Financial Industry Business Ontology [Ben13] used to describe and give meaning to data in the financial sector and the IEEE Standard Ontology for Robotics and Automation [SPM$^+$12].

There are a variety of DL languages, each of which is defined according to its expressivity. The name given to a DL language, for example $\mathcal{ALC}$, refers to the operators that can be used to construct concepts in the given language. Extensions of a DL language are then named according to the additional operators added compared to the original language, for example $\mathcal{ALCI}$ extends the language $\mathcal{ALC}$ with inverse role symbols. Generally, as the expressivity of DLs increases, so does the computational complexity of reasoning in the language.

An overview of description logic languages is provided in [BCM$^+$03]. Here, the focus is on first introducing the core DL relevant to this thesis: $\mathcal{ALC}$. Following this, extensions of $\mathcal{ALC}$ that are relevant to later chapters are also introduced. The specific motivation behind the use of these extensions with respect to the abduction problem are discussed in the relevant chapters.

2.2 The Description Logic $\mathcal{ALC}$

The DL language $\mathcal{ALC}$, or "Attributive Concept descriptions with Complements" [SSS91], is a widely used DL language. It is considered to be a foundational example of an expressive DL. Here an overview of the syntax and semantics of $\mathcal{ALC}$ are defined.

Let N_C, N_R and N_I be pairwise disjoint and countably infinite sets of concept, role and individual symbols respectively. Concepts in $\mathcal{ALC}$ can take one of the forms in the left-hand column in Table 2.1.

A concept is referred to as an *atomic* concept if it is a concept name, and *complex* if it is constructed using a combination of $\mathcal{ALC}$ operators and atomic concepts. For

$\mathcal{ALC}$ **concepts**		$\mathcal{ALC}$ **TBox axioms**	
Top concept:	$\top$	Concept inclusion:	$C \sqsubseteq D$
Bottom concept:	$\bot$	Concept Equivalence:	$C \equiv D$
Atomic concept:	A		
Negation:	$\neg C$		
Conjunction:	$C \sqcap D$		
Disjunction:	$C \sqcup D$	$\mathcal{ALC}$ **ABox assertions**	
$\exists$-role restriction:	$\exists r.C$	Concept assertion:	$C(a)$
$\forall$-role restriction:	$\forall r.C$	Role assertion:	$r(a,b)$

Table 2.1: The forms taken by concepts and axioms in the description logic $\mathcal{ALC}$, where A denotes a concept name $A \in N_C$, r denotes a role name $r \in N_R$, $a,b \in N_I$ denote individual names and C and D are arbitrary $\mathcal{ALC}$ concepts.

example, the concept $Reptile$ is an atomic concept, consisting of a single concept symbol, while the concept $\exists hasFeature.ScalySkin$ is a complex concept constructed using the operator $\exists$, the role symbol $hasFeature$ and the concept symbol $ScalySkin$.

The semantics of $\mathcal{ALC}$ can be defined as follows. An interpretation $\mathcal{I}$ consists of two components: a non-empty set $\Delta^{\mathcal{I}}$ called the *domain* and a function $\cdot^{\mathcal{I}}$ which assigns to every atomic concept symbol $A \in N_C$ a subset of the domain $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, to every role symbol $r \in N_R$ a binary relation $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ and to every individual symbol $a \in N_I$ an element of the domain $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$. The semantics of the general $\mathcal{ALC}$ concepts in Table 2.1 is then defined by extending the interpretation function $\cdot^{\mathcal{I}}$ using the inductive definitions shown in Table 2.2.

$\mathcal{ALC}$ concept	Interpretation
$\bot$	$\emptyset$
$\top$	$\Delta^{\mathcal{I}}$
$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
$\exists r.C$	$\{x \in \Delta^{\mathcal{I}} \mid \exists y.(x,y) \in r^{\mathcal{I}} \land y \in C^{\mathcal{I}}\}$
$\forall r.C$	$\{x \in \Delta^{\mathcal{I}} \mid \forall y.(x,y) \in r^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}}\}$

Table 2.2: Interpretation of $\mathcal{ALC}$ concepts where C, D are arbitrary concepts, r is a role symbol, $\Delta^{\mathcal{I}}$ is the domain and $C^{\mathcal{I}}$ denotes the application of the interpretation function $\cdot^{\mathcal{I}}$ to the concept C.

An $\mathcal{ALC}$ ontology captures knowledge about a domain that can be represented using the $\mathcal{ALC}$ concepts in Table 2.1 as well as the set of individuals N_i. The knowledge described by an ontology is split into two parts: the *TBox* and the *ABox*, containing axioms in the right-hand side of Table 2.1. The TBox describes terminological knowledge regarding general entities, i.e., concepts. It consists of statements of two forms: *concept inclusion* axioms of the form $C \sqsubseteq D$ and *concept equivalence* axioms of the form $C \equiv D$, where C and D are arbitrary $\mathcal{ALC}$ concepts of one of the forms given above. An axiom of the form $C \equiv D$ can also be represented by two inclusion axioms, $C \sqsubseteq D$ and $D \sqsubseteq C$. Meanwhile, the ABox describes assertional knowledge about specific instances of concepts, i.e., individuals. It contains *concept assertions* of the form $C(a)$ and *role assertions* of the form $r(a,b)$. For simplicity, in this thesis the term *ABox axioms* will be used interchangeably to refer to *ABox assertions*. To illustrate, consider the following simple example:

Example 2.2.1. *Consider the following TBox and ABox, both expressed in $\mathcal{ALC}$:*

$$\begin{aligned}
\mathcal{T} = \{&Reptile \sqsubseteq Animal,\\
&Mammal \sqsubseteq Animal,\\
&Reptile \sqcap Mammal \sqsubseteq \bot,\\
&ScalySkin \sqsubseteq BodyFeature,\\
&\top \sqsubseteq \forall hasBodyFeature.BodyFeature,\\
&Reptile \sqsubseteq \exists hasBodyFeature.ScalySkin,\\
&Hatchling \equiv \exists hasAgeGroup.Infant \sqcap Reptile\}\\
\mathcal{A} = \{&\exists hasBodyFeature.ScalySkin(rep1),\\
&isParentOf(rep1, rep2)\}
\end{aligned}$$

Together, $\mathcal{T}$ and $\mathcal{A}$ form an $\mathcal{ALC}$ ontology $\mathcal{O}$, where $\mathcal{O} = \mathcal{T} \cup \mathcal{A}$.

In the above example, the axiom $Reptile \sqsubseteq Animal$ is a concept inclusion, while $Hatchling \equiv \exists hasAgeGroup.Infant \sqcap Reptile$ is a concept equivalence. The axiom

$Reptile \sqcap Mammal \sqsubseteq \bot$ is a specific type of concept inclusion, often referred to as a disjointness axiom, specifying that no domain element can belong to both *Reptile* and *Mammal* simultaneously. The axiom $\top \sqsubseteq \forall hasBodyFeature.BodyFeature$ is another type of concept inclusion, known as a range axiom, specifying that all successors of the *hasBodyFeature* relation must be instances of the concept *BodyFeature*. The final two axioms are examples of typical ABox axioms: a concept and role assertion respectively, where $rep1$ and $rep2$ are individuals.

An $\mathcal{ALC}$ ontology is therefore the union of a set of TBox axioms (TBox) $\mathcal{T}$ and a set of ABox axioms (ABox) $\mathcal{A}$, i.e., $\mathcal{O} = \mathcal{T} \cup \mathcal{A}$. For the remainder of this thesis, the notation $\mathcal{T}, \mathcal{A}$ will be used instead of $\mathcal{T} \cup \mathcal{A}$, and $\mathcal{O}, \beta$ instead of $\mathcal{O} \cup \beta$ where β is an axiom. This will be used for any arbitrary combination of sets of DL axioms, for example in the abduction problem the notation $\mathcal{O}, \mathcal{H}$ will be used to refer to the union $\mathcal{O} \cup \mathcal{H}$ of a background ontology $\mathcal{O}$ and a set of DL axioms as a hypothesis $\mathcal{H}$. In cases where the second argument takes the form of a single axiom the same notation will be used, where the second argument is interpreted as a singleton set. For example, for $\mathcal{O}, \mathcal{H}$ it is often the case that the hypothesis $\mathcal{H}$ will consist of a single axiom.

In the literature, the term "ontology" is often used to refer directly to the terminology defined by a TBox, i.e., the tuple $\mathcal{O} = \langle \mathcal{T} \rangle$. Meanwhile, the term "knowledge base" is used to refer to the combination of a terminological component and a data or assertional component, i.e., the tuple containing both a TBox and an ABox $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$. However, in this thesis the terms "ontology" and "knowledge base" are used interchangeably unless specified otherwise (see Chapter 7).

The semantics of TBox and ABox axioms is defined as follows. For a given axiom α and an interpretation $\mathcal{I}$, the notation $\mathcal{I} \models \alpha$ means that α is true in $\mathcal{I}$. A concept inclusion $C \sqsubseteq D$, where C and D are arbitrary $\mathcal{ALC}$ concepts, is true under an interpretation $\mathcal{I}$ iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. Similarly, a concept equivalence $C \equiv D$ is true under an interpretation $\mathcal{I}$ iff $C^{\mathcal{I}} = D^{\mathcal{I}}$. An interpretation $\mathcal{I}$ is referred to as a *model* of a TBox $\mathcal{T}$ iff every axiom contained in $\mathcal{T}$ is true under $\mathcal{I}$. If $\mathcal{I}$ is a model of $\mathcal{T}$ then the notation $\mathcal{I} \models \mathcal{T}$ is used. For ABox axioms, a concept assertion $C(a)$ is true under $\mathcal{I}$ iff $a^{\mathcal{I}} \in C^{\mathcal{I}}$

and a role assertion $r(a,b)$ is true under $\mathcal{I}$ iff $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$. The interpretation $\mathcal{I}$ is a model for an ABox $\mathcal{A}$ iff every axiom in$\mathcal{A}$ is true under $\mathcal{I}$, and as before the notation $\mathcal{I} \models \mathcal{A}$ is used in this case. For an ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$, if $\mathcal{I}$ is a model of both the TBox $\mathcal{T}$ and ABox $\mathcal{A}$, then it is a model of the ontology, i.e., $\mathcal{I} \models \mathcal{O}$.

A given ontology $\mathcal{O}$ is referred to as *inconsistent* if there is no interpretation $\mathcal{I}$ that is a model of $\mathcal{O}$. If an axiom α is true in every model of an ontology $\mathcal{O}$, then the axiom α is said to be entailed by $\mathcal{O}$, written as $\mathcal{O} \models \alpha$. If every axiom of an ontology $\mathcal{O}$ is entailed by a second ontology $\mathcal{O}'$, then $\mathcal{O}$ is said to be entailed by $\mathcal{O}'$, written $\mathcal{O}' \models \mathcal{O}$.

A given class C is said to be *unsatisfiable* under an ontology $\mathcal{O}$, i.e., $C \equiv \perp$, if there is no model of $\mathcal{O}$ for which the interpretation of C is non-empty.

In this thesis, the term *signature* will be used to refer to a set of concept and role symbols, and is denoted by $sig(X)$, where X can be any one of a DL concept, a DL axiom, a TBox, an ABox or an ontology (knowledge base). For example, given ontology $\mathcal{O}$, the signature $sig(\mathcal{O})$ is the set of concept and role symbols occurring in $\mathcal{O}$.

2.3 Extensions to $\mathcal{ALC}$

Here the extensions to $\mathcal{ALC}$ that are relevant to this thesis are defined. These include nominals, inverse roles, disjunctive assertions ($\vee$) and fixpoints (ν, μ). In each case, the semantics of the extended language is obtained by extending the interpretation function $\cdot^{\mathcal{I}}$ defined for $\mathcal{ALC}$ in the previous section.

2.3.1 Nominals

When describing $\mathcal{ALC}$, it is stated that individual symbols can only occur in the ABox of an ontology in either concept assertions $C(a)$ or role assertions r(a, b). However, in some cases it can be useful to include individuals in the TBox. For this purpose, nominals are utilised, indicated by $\mathcal{O}$ in the name of the DL language, e.g. $\mathcal{ALCO}$. A nominal is a concept that has only one instance, for example the nominal $\{a\}$ denotes the concept with the sole instance being the individual a.

For example, the following ABox axiom:

$$\exists \text{hasChild}.\top(\text{mary})$$

states that the individual *mary* has at least one child. Suppose it is also known that *mary* has two children, *jim* and *james*, where $jim \neq james$. This information could be added to the ABox in the form of two role assertions $hasChild(mary, jim)$ and $hasChild(mary, james)$. However, suppose it is specifically known that *mary* only has *jim* and *james* as children and a modeller would like to express this information directly. Nominals provide several options for this, such as:

$$\forall \text{hasChild}.(\{\text{james}\} \sqcup \{\text{jim}\})(\text{mary})$$

Alternatively, the enumeration of the children of Mary can be constructed as follows:

$$\text{ChildrenOfMary} \equiv \{\text{james}\} \sqcup \{\text{jim}\}$$

where for an individual a, the corresponding nominal is denoted as $\{a\}$. Thus, nominals are useful in expressing the enumeration of a set of individuals when combined with the disjunction operator $\sqcup$ [KSH12]. It is also worth noting that ABox assertions can be represented using nominals: a concept assertion $C(a)$ can be equivalently represented as $\{a\} \sqsubseteq C$ while a role assertion $r(a,b)$ can be represented as $\{a\} \sqsubseteq \exists r.\{b\}$. This highlights the absence of a mathematical significance behind the separation of the TBox and the ABox, which exists primarily for modelling purposes [KSH12].

For the purposes of this thesis, it is also worth noting that certain forms of axioms that would normally not be available in the considered DLs can be equivalently represented using nominals. In most formalisms ABoxes do not permit negated role assertions of the form $\neg r(a,b)$, with a few exceptions [ABHM03, LLMW06] and in the expressive DL $\mathcal{SROIQ}$ [HKS06]. However, for some approaches it can be necessary to be able to equivalently represent such a negated assertion, without the full machinery available in the most expressive DLs. For the work in this thesis, particularly Chapter 6, nominals are used to equivalently represent negated role assertions when required, where the assertion $\neg r(a,b)$ can be represented as $\{a\} \sqsubseteq \forall r.\neg\{b\}$.

The extension of $\mathcal{ALC}$ with nominals is denoted by $\mathcal{ALCO}$. The semantics of $\mathcal{ALCO}$ can be obtained from $\mathcal{ALC}$ by extending the interpretation function $\cdot^{\mathcal{I}}$ as follows: let N_O be the set of *nominal symbols*. For each $a \in N_O$, the function $\cdot^{\mathcal{I}}$ assigns the singleton set $\{a^{\mathcal{I}}\} \subseteq \Delta^{\mathcal{I}}$.

2.3.2 Inverse Roles

Inverse roles enable the representation of a symmetrical relationship between two binary roles. They provide a means by which entities can be described by the sum of their parts, and simultaneously parts can be described by the entities to which they belong [HS99]. For example, given the following axioms:

$$\text{RotorSystem} \sqsubseteq \exists\text{isPartOf.Helicopter}$$
$$\text{Helicopter} \sqsubseteq \text{FlyingVehicle} \sqcap \exists\text{hasPart.RotorSystem}$$

without inverse roles, it is not possible to model the fact that the roles *isPartOf* and *hasPart* are directly related: they are inverse relations. This can be specified as follows:

$$\text{isPartOf} \equiv \text{hasPart}^{-}$$

where r^{-} indicates the inverse of a role r. The usefulness of this notion is also evident when considering assertional information. For example, if it is known that two individuals a and b are related under the *hasParent* relation, i.e., $\mathit{hasParent}(a,b)$ then it should not be the case that $\exists \mathit{hasChild}.\bot(b)$, i.e., b has no child, since *hasChild* is the inverse of *hasParent*. To capture this notion, it is sufficient to identify that *hasChild* and *hasParent* are inverses to one another.

The semantics of $\mathcal{ALC}$ extended with inverse roles can be obtained by extending the interpretation function as follows: $(r^-)^\mathcal{I} = \{(y,x) \in \Delta^\mathcal{I} \times \Delta^\mathcal{I} | (x,y) \in r^-\}$.

2.3.3 Universal Role

In some cases, it is useful to extend the set of role symbols N_R to include the *top* or *universal* role, which is denoted by ∇. This notion is analogous to the notion of the top concept $\top$, where the universal role ∇ relates all pairs of individuals.

The semantics of $\mathcal{ALC}$ extended with the universal role (∇) can be obtained by extending the interpretation function $\cdot^\mathcal{I}$ in Table 2.2 as follows:

$$\nabla^\mathcal{I} = \Delta^\mathcal{I} \times \Delta^\mathcal{I}.$$

The use cases for the universal role in the context of this thesis are discussed alongside the proposed abduction methods.

2.4 Additional Language Features

2.4.1 Disjunctive Axioms

In most cases, the ABox of a DL ontology does not permit disjunctions over multiple individuals, permitting only disjunctions over a single individual, i.e., ABox axioms of the form $C \sqcup D(a)$. In applications requiring explicit representation of partial information [ABHM03], this is overly restrictive. Abduction is one of these scenarios, since the hypothesis produced as a result of abduction is not guaranteed to be true, it is only a possible explanation for the given observation. Often, there exists not one possible explanation, but a set of competing explanations that could be true given the information available. Representing the entire set of explanations gives rise to the ability to perform further investigation to confirm or deny the truth of a given explanation, arriving at a more specific hypothesis when more information has been obtained.

Therefore, it is useful to utilise the notion of a disjunctive ABox assertions [KS15b].

A disjunctive assertion is an ABox axiom of the following form:

$$C_1(a_1) \vee ... \vee C_n(a_n)$$

where each C_i is a concept in the language and each a_i is an individual. Cases involving disjunctions over a single individual, such as $C(a) \vee D(a)$, can also be expressed as a formula in pure $\mathcal{ALC}$ making use of the $\mathcal{ALC}$ disjunction operator $\sqcup$, e.g., $(C \sqcup D)(a)$.

An ontology expressed in $\mathcal{ALC}$ extended with disjunctive assertions is an ontology containing at least one ABox axiom of the above form, where each C_i is an arbitrary $\mathcal{ALC}$ concept. The semantics of disjunctive assertions is as follows:

$$\mathcal{O} \models C_1(a_1) \vee ... \vee C_n(a_n) \qquad \text{iff} \qquad \mathcal{O} \models C_i(a_i)$$

for some $i \in \{1, ..., n\}$, where $\mathcal{O}$ is an ontology.

The notion of disjunctive assertions will also be lifted to include disjunctions of ABoxes, i.e., disjunctions of the form $\alpha_1 \vee ... \vee \alpha_n$ where each α_i for $1 \leq i \leq n$ is a conjunction of $\mathcal{ALC}$ ABox axioms. In Chapter 7, disjunctive TBoxes and hence disjunctions of DL knowledge bases are introduced. Since this extension is driven by the proposed abduction problem and approach, it is discussed in the aforementioned chapter. In each case, the notation $(\vee)$ will be used to denote the extension of a DL language with disjunctive axioms, for example $\mathcal{ALC}(\vee)$ which extends $\mathcal{ALC}$ with disjunctive axioms. The extent of the disjunction, i.e., disjunctions of ABox axioms, TBox axioms or both, will be stated for each of the proposed problems and solutions.

2.4.2 Fixpoints

In some cases, cycles may occur in ontologies, resulting in concepts that cannot be finitely represented without additional constructs. Fixpoint operators provide a way to extend DL languages to enable finite representations of these cycles [Sch94, CDGL99].

The symbol $X \in N_v$ is used to represent a concept variable, where N_v is the set of concept variables and is disjoint from the sets N_C, N_R, N_I and N_O. Fixpoints are then concepts of the form:

$$\mu X.C[X] \qquad\qquad \text{or} \qquad\qquad \nu X.C[X]$$

which denote *least* and *greatest* fixpoints respectively [CDGL99]. Concepts of the form $C[X]$ are viewed as functions, where $C[C']$ takes the concept C' as an argument and returns the result of replacing all occurrences of X in C with C'. Here, the concept $C[X]$ represents a concept in which the concept variable X occurs only positively, i.e., under an even number of negations [Koo15]. In the above, least and greatest fixpoint concepts X is said to be bound, where μ and ν are treated as quantifiers [CDGL99]. If all variables in $C[X]$ are bound, then C is *closed*. Otherwise, $C[X]$ is said to be open. In this work, it is required that all C occurring in the axioms of an ontology are closed.

Under the Knaster-Tarski theorem [T$^+$55], all monotone functions have a least and greatest fixpoint [Sch94]. If a concept $C[X]$ contains X only positively, then $C[X]$ is monotonic with respect to the inclusion relation $\sqsubseteq$. Since $\sqsubseteq$ has a minimal and maximal element in the top $\top$ and bottom $\bot$ concepts respectively, this means that for each concept $C[X]$ there always exists a least and greatest fixpoint [Koo15].

The semantics of fixpoints in DLs can be defined as follows [CDGL99]. Let $\mathcal{I}$ be an interpretation and ρ be a valuation on $\mathcal{I}$ which maps concept variables X to subsets of the domain $\Delta^{\mathcal{I}}$. The function $\cdot^{\mathcal{I}}_{\rho}$ is an *extension function* which maps concepts to subsets of the domain $\Delta^{\mathcal{I}}$ and nary relations to subsets of $(\Delta^{\mathcal{I}})^n$. The semantics of fixpoint operators can then be provided as follows [CDGL99]:

$$(\nu X.C)^{\mathcal{I}} = \bigcup\{\mathcal{E} \subseteq \Delta^{\mathcal{I}} | \mathcal{E} \subseteq C^{\mathcal{I}}_{\rho[X/\mathcal{E}]}\}$$
$$(\mu X.C)^{\mathcal{I}} = \bigcap\{\mathcal{E} \subseteq \Delta^{\mathcal{I}} | C^{\mathcal{I}}_{\rho[X/\mathcal{E}]\sqsubseteq\mathcal{E}}\}$$

where $\rho[X/\mathcal{E}]$ is a valuation function identical to ρ aside from the following equality: $\rho[X/\mathcal{E}](X) = \mathcal{E}$. If C is a closed concept, as is assumed in this setting, then the extension is independent of the valuation [CDGL99]. Thus, the interpretation function $\cdot^{\mathcal{I}}$ provided in Table 2.2 can be extended by setting $C^{\mathcal{I}} = C^{\mathcal{I}}_{\rho}$ for each concept C.

The extension of $\mathcal{ALC}$ with greatest (least) fixpoints is denoted by $\mathcal{ALC}\nu$ ($\mathcal{ALC}\mu$). As greatest and least fixpoints are dual to each other [GSS08]:

$$\mu X.C[X] \equiv \neg \nu X.\neg C[X/\neg X]$$
$$\nu X.C[X] \equiv \neg \mu X.\neg C[X/\neg X]$$

meaning that the negation of a greatest fixpoint results in a least fixpoint.

2.5 DL Languages Used in this Work

The DL languages used in this thesis are summarised in Table 2.3. The languages are split into "input", referring to the language used for the input provided to the proposed abduction methods, and "output", which refers to the language used for the result (hypothesis) in the most expressive cases.

The thesis focuses on abduction problems for which both the background ontology $\mathcal{O}$ and the observation ψ are expressed in $\mathcal{ALC}$. Therefore, the input to the abduction methods is in $\mathcal{ALC}$ throughout. The DL languages used to represent the results of intermediate steps, for example contraposition to obtain $\mathcal{O}, \neg\psi$, are discussed alongside the proposed abduction methods. Chapter 4 defines the abduction problem for $\mathcal{ALC}$, where the hypothesis is a disjunction of explanations, necessitating the use of $\mathcal{ALC}(\vee)$ for the output. Subsequent chapters present abduction methods that produce more expressive solutions, where the output language depends upon the characteristics of the proposed algorithm, which differs in Chapters 5, 6 and 7. The rationale behind these output languages is motivated and discussed in the relevant chapters.

As a summary, the output languages used to represent hypotheses produced by the methods presented in this work are defined as follows. Concepts in the DL $\mathcal{ALC}$ can be constructed according to the following syntax rules:

$$C ::= A \mid \neg C \mid C \sqcup C \mid \exists r.C$$

Chapter	**Input Language**	**Output Language**
4	$\mathcal{ALC}$	$\mathcal{ALC}(\vee)^*$
5	$\mathcal{ALC}$	$\mathcal{ALC}\mu(\vee)^*$
6	$\mathcal{ALC}$	$\mathcal{ALCOI}(\nabla)$
7	$\mathcal{ALC}$	$\mathcal{ALCOI}\mu(\vee)^{**}$

Table 2.3: A summary of the description logic languages used throughout this thesis. * denotes the use of disjunctive ABoxes, while ** denotes the use of disjunctive knowledge bases (i.e., disjunctions of ontologies).

where $A \in N_C$ is an atomic concept, C is an arbitrary $\mathcal{ALC}$ concept and $r \in N_r$ is a role. The DL $\mathcal{ALCOI}$ extends the set N_r to include inverse roles r^- and permits additional concepts of the form $\{a\}$ where $\{a\} \in N_o$ is the nominal corresponding to the individual $a \in N_I$. The DL $\mathcal{ALCOI}(\nabla)$ then extends the set of role symbols N_r further to include the universal role ∇.

The DL $\mathcal{ALC}\mu$ ($\mathcal{ALCOI}\mu$) extends $\mathcal{ALC}$ ($\mathcal{ALCOI}$) with *least fixpoint concepts* of the form $\mu X.C$, where $X \in N_v$ is a concept variable. For least fixpoint concepts $\mu X.C$, X occurs only positively, i.e., under an even number of negations ($\neg$).

The DL $\mathcal{ALC}(\vee)$ ($\mathcal{ALC}\mu(\vee)$) extends $\mathcal{ALC}$ ($\mathcal{ALC}\mu$) with disjunctive ABox axioms of the form $C_1(a_1) \vee ... \vee C_n(a_n)$ where each $C_i \in N_C$ for $1 \leq i \leq n$ is an $\mathcal{ALC}$ ($\mathcal{ALC}\mu$) concept and each $a_i \in N_I$ is an individual.

In Chapter 7, the DL $\mathcal{ALCOI}\mu(\vee)$, extends this notion further to include disjunctions of both TBox and ABox axioms, i.e., disjunctions of DL ontologies. The term boolean knowledge base (KB) will be used to refer to this extension, which can be constructed as follows:

$$C ::= \beta \mid \neg\mathcal{K} \mid \mathcal{K} \wedge \mathcal{K} \mid \mathcal{K} \vee \mathcal{K}$$

where β is an $\mathcal{ALCOI}\mu$ axiom and $\mathcal{K}$ is a KB. Each KB $\mathcal{K}$ is defined as the conjunction of the axioms contained within it. This then extends the DL setting to include negation and disjunction of ontologies, which will be motivated in Chapter 7. The semantics for a given KB depends upon the language in which it is expressed. The semantics of the DLs considered are the same as defined in previous sections. The interpretation

$\mathcal{I}$ of KBs can be extended from these semantics as follows. For a given KB $\mathcal{K}$, an interpretation $\mathcal{I}$ which satisfies $\mathcal{K}$, denoted as $\mathcal{I} \models \mathcal{K}$, is referred to as a model of $\mathcal{K}$. For the constructions of $\mathcal{K}$ above: $\mathcal{I} \models \neg\mathcal{K}$ if $\mathcal{I} \not\models \mathcal{K}$, $\mathcal{I} \models \mathcal{K}_1 \wedge \mathcal{K}_2$ if both $\mathcal{I} \models \mathcal{K}_1$ and $\mathcal{I} \models \mathcal{K}_2$, and finally $\mathcal{I} \models \mathcal{K}_1 \vee \mathcal{K}_2$ if either $\mathcal{I} \models \mathcal{K}_1$ or $\mathcal{I} \models \mathcal{K}_2$. The notation $\mathcal{K}_i \models \mathcal{K}_j$ means that every model of $\mathcal{K}_i$ is also a model of the KB $\mathcal{K}_j$.

Chapter 3

Background on Abduction and Forgetting

This Chapter provides context behind abductive reasoning, since this is the core problem addressed in this thesis with focus on abduction in the setting of description logic ontologies. However, there exists a range of significant work on abduction across several subfields of Artificial Intelligence. Therefore, a brief introduction to the philosophical background of abductive reasoning is given, followed by an overview of studies on abduction in Artificial Intelligence, including relevant work in the area of Abductive Logic Programming (ALP). Work on abductive reasoning in DL ontologies is subsequently discussed with respect to the main problems that have been identified in this setting, common constraints applied to the hypotheses and approaches that have been developed to solving various instantiations of the abduction problem. Work in classical logics on the use of second-order quantifier elimination for abductive reasoning problems is then discussed. This is of particular relevance to this thesis, since the approaches here are based on forgetting which is closely related to the problem of second-order quantifier elimination. Details on the two forgetting approaches relevant to this thesis are then provided, focusing on the key properties that are needed to provide context and understanding to the abduction approaches proposed in this work. Finally, the connection between forgetting and abductive reasoning is discussed.

3.1 Abductive Reasoning

There exist multiple perspectives on the form taken by and role of abductive reasoning, particularly with respect to its distinction from inductive reasoning. Two ways of distinguishing and defining the interactions between deduction, induction and abduction were given by Peirce. The first is referred to as his *syllogistic theory*, where syllogisms are arguments usually represented in three lines: two propositions from which a conclusion is drawn deductively. This framing of abduction used syllogisms to illustate the difference between each form of reasoning, as shown in Figure 3.1 via an example of each case [Pei78].

From this, it can be seen that the three forms of reasoning differ in both input and output. Deduction takes a general rule and a specific case, both known to be true, and derives a consequence that must be true as a result. Induction, on the other hand, takes a specific case and a result, both known to be true, and generalises to a rule that probably covers the entire population from which the case was drawn. Finally, abductive reasoning takes a rule and a result, both known to be true, and provides a possible explanation as to why the specific result holds given the prior rule.

The second perspective taken by Peirce resulted in his *inferential theory* [Pei60, FK00a]. Rather than specifically considering syllogistic arguments, Peirce referred to

Deduction	*The beans in this bag are white* *These beans were in the bag* Conclusion: *These beans are white*	(Rule) (Case) (Result)
Induction	*These beans are selected from this bag* *These beans are white* Conclusion: *All the beans from this bag are white*	(Case) (Result) (Rule)
Abduction	*All the beans from this bag are white* *These beans are white* Conclusion: *These beans are from this bag*	(Rule) (Result) (Case)

Figure 3.1: The distinction between deduction, induction and abduction in Peirce's syllogistic theory.

the interaction between the three forms of reasoning in terms of how they are used to infer new knowledge based on prior knowledge and observations. Here, the three forms of reasoning are seen in terms of a separate stage in this process. Abduction is the process of hypothesis generation: when something new is observed, abduction is used to produce an initial hypothesis to explain the observation. In this way, abduction can be viewed as a "flash of insight" [Pei60]. Following this, deduction is the process of deriving consequences that would hold if the initial hypothesis was assumed to be true, under existing knowledge. Induction is then the process of hypothesis evaluation, or hypothesis refinement, akin to the process of performing an experiment to confirm or deny the initial explanation. These steps then form a loop which gradually accounts for new knowledge by generating, testing and refining new hypotheses. This explicitly separates the process of hypothesis generation from hypothesis evaluation, which are otherwise often assumed to be part of the same process.

3.2 Abduction in Artificial Intelligence

Abductive reasoning has been a recurring topic of interest in various subfields of Artificial Intelligence (AI), having long been recognised as an important mechanism for reasoning and problem solving [Pop73]. The role filled by abductive reasoning, i.e., accounting for new observations by producing an explanation that leverages prior knowledge, has led to a wide range of work on applying abduction to key problems in AI such as scientific hypothesis generation [KWJ+04, Ray07], diagnostics [SMvS+18] and interpretation of natural language [Sti91, HSAM93].

Abduction has also been recognised as a hard problem in AI. In terms of computational cost, abductive reasoning is difficult in comparison to many of the reasoning tasks commonly addressed via deductive approaches [EG95], requiring the use of a range of constraints designed to restrict the size of the search space for hypotheses. Nevertheless, abduction has been studied extensively and applied successfully in several subfields of AI. Among these is the area of logic programming [DK02], where the explicit separation between induction and abduction has proven to be an interesting and useful perspective. In fields such as inductive [MDR94] and abductive [KKT92] logic programming, there exists a variety of work on integrating abduction and induction leading to the development of systems that utilise abductive reasoning as a core component [MB00, RBR03, TNCKM06, Ray09, IFKN09, CRL10].

Abductive reasoning has also been studied from a variety of perspectives in classical logics. Abduction can be linked to the generation of prime implicates, which has been studied recently in the setting of first-order logic [EPT17, EPS18]. The notion of weakest sufficient conditions [Lin01] is also closely related to the notion of a least assumptive abductive hypothesis. Techniques for performing second-order quantifier elimination have been proposed as a promising direction for computing weakest sufficient conditions for the purpose of abductive reasoning [DLS01, GSS08, Wer13]. Second-order quantifier elimination techniques are related to the task of forgetting, which is discussed further in Section 3.4.

The perspective of integrating abductive and inductive reasoning in a cycle of extending and refining existing knowledge has garnered significant interest [FK00b]. This is exemplified by the fields of abductive and inductive reasoning as discussed above, where the importance of abduction continues to be emphasised as an important direction for future work [MDRP^{+}12]. The integration of these two forms of reasoning has also been investigated in other areas of AI such as machine learning [Moo00], statistical relational AI [RM10, BHD^{+}11] and natural language processing [RNM05]. The integration of induction and abductive reasoning has continued to be identified as a promising direction for recent approaches to statistical learning [DXYZ19, Zho19].

3.3 Abduction in Description Logics

The need for abductive reasoning in DL ontologies was motivated by [EKS06], who outlined a number of application scenarios and forms for the abduction problems in this context. The need for abductive reasoning from an ontology engineering perspective, particularly in the tasks of ontology alignment and quality assurance, was also advocated by [BMH08].

In the setting of DL ontologies, it is common to separate the basic abduction problem into several subproblems. The main forms of abduction that have been identified include concept abduction, ABox abduction, TBox abduction and Knowledge-Base (KB) abduction. One of the primary differences between each of these abduction problems is the forms taken by the observations and hypotheses.

Concept abduction focuses on the task of finding all subconcepts for a given concept, thereby extending the range of subsumptions in a given ontology [CDNDS+03, CDNDS+05]. The concept abduction problem can be defined as follows [CDNDS+05]:

Definition 3.3.1. *Let $\mathcal{L}$ be a DL language, S and D be concepts in $\mathcal{L}$ and $\mathcal{O}$ be a set of axioms in $\mathcal{L}$, where both S and D are satisfiable in $\mathcal{O}$. The* Concept Abduction Problem *is the task of finding a concept $H \in \mathcal{L}$ such that $\mathcal{O} \models (S \sqcap H) \sqsubseteq D$ and $S \sqcap H$ is satisfiable in $\mathcal{O}$. In this case, H is a* hypothesis *about S according to D and $\mathcal{O}$.*

Concept abduction has been applied to problems such as semantic matchmaking in electronic marketplaces [CDNDS+03] and image understanding [AHB13]. In contrast to the other forms of abduction discussed, which focus on observations and hypotheses as sets of DL statements, concept abduction focuses on observations and hypotheses in the form of individual DL concepts. As a result, the task of concept abduction is inherently different to the forms of abduction most relevant to this thesis, and the methods for concept abduction cannot be directly extended to handle more challenging abduction problems including ABox, TBox and KB abduction.

Additionally, concept abduction has also been considered as part of a framework of belief revision for ontologies, where the revision of an ontology with a new piece of information is split into two steps: contraction and expansion [CDNDS+05, RSS20]. Given a new piece of knowledge, contraction first modifies the original ontology to ensure that the resulting ontology is consistent with the new information. Following this, an expansion step is applied to add the new information to the modified ontology. In the setting of this work, the aim is instead to generate a hypothesis that is consistent with the original ontology, which will be reflected in subsequent definitions of the abduction problem. As such, contraction is not considered, while the abduction process provides a candidate hypothesis to consistently expand the original ontology.

A basic definition for the class of abduction problems most relevant to this thesis is as follows:

Definition 3.3.2. ***Abduction in DLs.*** *Let $\mathcal{O}$ be a knowledge base and ψ be a set of axioms, both expressed in a DL language $\mathcal{L}$. If $\mathcal{O}, \psi \not\models \bot$ and $\mathcal{O} \not\models \psi$, then the tuple $\langle \mathcal{O}, \psi \rangle$ is an instance of the basic abduction problem. The aim of the basic abduction problem is to compute a hypothesis $\mathcal{H}$ as a set of axioms, expressed in a DL language $\mathcal{L}'$, satisfying the following constraints:*

(i) ***Consistency:*** *$\mathcal{H}$ is said to be consistent if $\mathcal{O}, \mathcal{H} \not\models \bot$*

(ii) ***Explanation:*** *$\mathcal{H}$ is said to be an explanation if $\mathcal{O}, \mathcal{H} \models \psi$*

The two constraints considered in the basic abduction problem, *consistency* and *explanation*, are perhaps the most common constraints applied to abductive hypotheses. Consistency captures the notion of not contradicting what is already known. Meanwhile, when added to the given background knowledge, an explanatory hypothesis should lead to the entailment of the given observation.

Two subtypes of this general problem can be defined as follows:

Definition 3.3.3. ***ABox Abduction***. *Consider the abduction problem $\langle \mathcal{O}, \psi \rangle$ from Definition 3.3.2. For the ABox abduction problem, ψ takes the form of a set of ABox axioms expressed in a DL language $\mathcal{L}$. The* solution to the ABox abduction problem *is a* hypothesis *$\mathcal{H}$ as a set of ABox axioms expressed in a DL language $\mathcal{L}'$ satisfying the constraints of (i) consistency and (ii) explanation.*

Definition 3.3.4. ***TBox Abduction***. *Consider the abduction problem $\langle \mathcal{O}, \psi \rangle$ from Definition 3.3.2. For the TBox abduction problem, ψ takes the form of a set of TBox* *axioms expressed in a DL language $\mathcal{L}$. The* solution to the TBox abduction problem *is a* hypothesis $\mathcal{H}$ *as a set of TBox axioms expressed in a DL language $\mathcal{L}'$ satisfying the constraints of (i)* consistency *and (ii)* explanation.

It is worth noting that in Definitions 3.3.2–3.3.4, no specific relation is imposed between the input language $\mathcal{L}$, used to express $\mathcal{O}$ and ψ, and the language used to express the solutions $\mathcal{L}'$. In many cases for abductive reasoning, the input language $\mathcal{L}$ and the output language $\mathcal{L}'$ are the same i.e., $\mathcal{L} = \mathcal{L}'$. In other cases, it may be that the hypotheses are restricted in some way, so that the output language $\mathcal{L}'$ is less expressive than the input language $\mathcal{L}$. An example of this case is in Klarman et al [KES11], which concerns an ABox abduction problem for which the background ontology $\mathcal{O}$ may be expressed in the DL language $\mathcal{ALC}$, while the hypotheses (and observations) are restricted to a less expressive language that excludes disjunction and uses negation only over concepts ($\mathcal{ALE}$).

Another possibility is that it may be necessary to utilise a more expressive language for the solutions compared to the input, depending on the requirements placed on and form taken by abductive hypotheses. This is particularly relevant in the context of this thesis, where the proposed abduction problem and the methods for solving this problem assume that the hypothesis $\mathcal{H}$ may be expressed in an extended DL language compared to the input. A core element of the work in this thesis is that the $\mathcal{H}$ produced as a solution to the abduction problem takes the form of a set of possible, alternative explanations for the observation ψ. This leads to the main language extension that will feature in this work: disjunctive axioms, which are used to represent this notion. This is discussed in Chapter 4, while other language extensions are discussed in subsequent chapters alongside the proposed abduction methods.

It is worth noting that, since the constraints are based upon entailment relations between $\mathcal{O}$, $\mathcal{H}$ and ψ, the language of the constraints is the same as the most expressive language used to express these (be it $\mathcal{L}$ or $\mathcal{L}'$).

As evident from Definitions 3.3.3 and 3.3.4, the ABox and TBox abduction problems differ in the form taken by the observations and hypotheses. This delineation between TBox and ABox abduction reflects the treatment of universally and existentially quantified axioms in the DL setting in general: ontologies are usually split into a terminology (TBox) about general entities and a collection of assertions or data regarding specific individuals (ABox). The majority of previous work on abduction in DLs focuses on tackling an instance of either the TBox or ABox abduction problem, rather than both simultaneously. The notion captured by ABox abduction corresponds more closely with the notion of abduction most commonly adopted in the fields of ALP [KKT92] and ILP [MDR94], where abduction is often separated from induction in the fact that it concerns ground explanations for ground observations without providing a generalising effect. TBox abduction, on the other hand, lifts the abduction problem to universally quantified statements, where neither the observations nor the hypotheses are ground. However, TBox abduction still focuses on producing an explanation for the given observation rather than on generalisation; the fact that the abductive hypotheses are TBox statements is a consequence of the need to provide an explanation for a TBox observation rather than the result of a process of generalisation.

A third, but previously unsolved, task is knowledge-base (KB) abduction. For KB abduction, the restrictions on the observations and hypotheses are lifted: these can contain a mixture of TBox or ABox axioms and therefore both the observation and hypothesis can be viewed as ontologies. In this way, KB abduction can be viewed as a generalisation of both the ABox and TBox abduction problems [EKS06] as in the basic abduction problem of Definition 3.3.2. For example, as pointed out in Elsenbroich, Kutz and Sattler [EKS06], if the given observation is a set of ABox assertions, then the hypothesis produced as a solution to the ABox abduction problem could be viewed as a solution to the KB abduction problem where the observation does not contain any TBox axioms. The same perspective can be taken on the relation between the TBox and KB abduction problems.

For each of the three general abduction problems defined above, the preconditions on the observation ψ constrain the focus to abductive problems. For example, if $\mathcal{O} \models \psi$, then the background knowledge in $\mathcal{O}$ already explains the observation ψ.

Thus, the problem of "explaining" ψ would instead be a deductive one, i.e., searching for an existing proof of ψ from the axioms in $\mathcal{O}$, which falls under problems such as computing justifications for existing entailments [KPHS07]. Additionally, the background knowledge base $\mathcal{O}$ must itself be consistent, and the observation ψ must be consistent with $\mathcal{O}$. Thus, the separate problem of abductive reasoning under inconsistency [DWS15] is not considered in this work.

So far, the basic abduction problem definitions have considered two constraints: *consistency* and *explanation*. However, the types of hypotheses returned as solutions to a given abduction problem vary widely, being defined in each case by the set of abductive constraints that they must satisfy. There are a variety of constraints in the abduction literature, and though two given problems may both be identified as a form of abduction, it is not necessarily the case that the problems being solved are similar: particularly if the abductive constraints used in both cases are significantly different. In general, the purpose of abductive constraints is to limit the size of the search space of hypotheses for practical purposes, since this space can be intractably large in many cases. However, this restriction should be made while simultaneously ensuring that the remaining, valid hypotheses are those that are most useful for a particular purpose. This makes the task of identifying and selecting constraints a challenging one.

Consistency and explanation are commonly used constraints throughout work on abduction in AI, including in the setting of DL ontologies. In some cases, an additional restriction to the notion of explanation is also specified. This constraint, referred to as relevance [EKS06, KES11], ensures that the hypothesis does not directly entail the observation, i.e. $\mathcal{H} \models \psi$, without the use of the background knowledge in $\mathcal{O}$. This avoids producing somewhat trivial explanations, for example the case that the hypothesis $\mathcal{H}$ is simply the same as the given observation ψ.

Solutions to the abduction problem can also be constrained so that the hypotheses can only be expressed using a set of allowed symbols, referred to as *abducibles*. In abductive logic programming this constraint is used widely [KKT92] and has more recently been investigated as part of implicate generation approaches in the setting of first-order logic [EPT17, EPS18]. In contrast, abducibles are less widely investigated in the setting in DLs. In fact, most abduction approaches in DLs do not provide a way to specify abducibles as part of the abduction problem, both in the cases of ABox abduction [KES11, HB12, PH17, PH18] and TBox abduction [WKDL14, DWM17]. Nonetheless, abducibles are useful in restricting the set of solutions to a specific subset of the domain knowledge contained in a given background ontology. This can be useful in avoiding uninteresting explanations in a number of application scenarios. For example, in diagnostics it may be useful to restrict the hypotheses produced to those that explain the observed fault, such as a disease or a malfunction in a given system, by making use of symbols specifically associated with causes.

The basic abduction problems specified in Definitions 3.3.2–3.3.4 above can be extended with the notion of abducible symbols. The abduction problem is then the triple $\langle O, \psi, \mathcal{S}_A \rangle$, where $\mathcal{S}_A \subseteq sig(\mathcal{O}, \psi)$ specifies the signature of abducible symbols and for a hypothesis $\mathcal{H}$ to be a valid solution, it must be the case that $sig(\mathcal{H}) \subseteq \mathcal{S}_A$. In this work, the signature of abducibles contains only concept and role symbols and for the presented methods, the hypotheses can contain any individuals specified in the ontology $\mathcal{O}$ or the observation ψ.

Hypotheses can also be constrained using notions of *minimality*. Two commonly discussed minimality criteria in the DL setting are *syntactic* and *semantic minimality*.

Syntactic minimality restrains solutions to the abduction problem to hypotheses that are shortest in length [EKS06]. Syntactic minimality has also been referred to as subset minimality and can be defined as follows [HB12, HBK14]:

Definition 3.3.5. *Given an abduction problem $\langle \mathcal{O}, \psi \rangle$, a hypothesis $\mathcal{H}$ is said to be* syntactically smaller *than a second hypothesis $\mathcal{H}'$ if $\mathcal{H} \subset \mathcal{H}'$. The hypothesis $\mathcal{H}$ is said to be* syntactically minimal *if no proper subset of $\mathcal{H}$ provides an explanation for ψ under $\mathcal{O}$.*

Effectively, this criterion focuses on removing unnecessary assertions or guesses from an explanation; if the observation can be explained by a given set of assertions, then any additional assertions are deemed to be superfluous. In this sense, syntactic minimality is useful for removing unnecessary axioms from explanations.

A second minimality criterion that is often discussed is *semantic minimality*. Semantic minimality focuses on relating hypotheses via entailment, rather than length. Under this notion of minimality, a hypothesis $\mathcal{H}$ is preferred over a second hypothesis $\mathcal{H}'$ if it is the case that $\mathcal{H}$ is weaker than $\mathcal{H}'$ under the available background knowledge, i.e., $\mathcal{H}'$ implies $\mathcal{H}$ but not the reverse. Semantic minimality can be defined as follows:

Definition 3.3.6. *Given an abduction problem $\langle \mathcal{O}, \psi \rangle$, a hypothesis $\mathcal{H}$ is said to be* semantically minimal *if there is no other hypothesis $\mathcal{H}'$ that is not equivalent to $\mathcal{H}$ under $\mathcal{O}$ such that $\mathcal{O}, \mathcal{H} \models \mathcal{H}'$.*

This criterion captures the notion of making the *least assumptions* necessary to explain the observation ψ given the available background knowledge in $\mathcal{O}$. This is particularly useful in ranking hypotheses with respect to the ontology $\mathcal{O}$ [HBK14].

It is possible for a hypothesis to be syntactically minimal but not semantically minimal. To illustrate the difference between these notions, consider the following ABox abduction example.

Example 3.3.1. *Consider a simplified ontology $\mathcal{O}$ concerning a species of reptile, which contains the following axioms:*

$$Pogona \sqsubseteq \exists livesIn.Arid$$

$$PogonaMinor \sqsubseteq Pogona$$

$$Arid \sqsubseteq Habitat$$

$$AridWoodland \sqsubseteq Arid$$

and let $\psi = \exists livesIn.Arid(ind)$ be a new observation about an individual ind. Let the set of abducible symbols exclude the symbol Arid, thereby avoiding the trivial, i.e., non-relevant explanation $\mathcal{H} = \psi$. For simplicity, let the hypotheses be restricted to *those that do not contain disjunction. Consider the following set of candidate hypotheses to explain ψ under $\mathcal{O}$:*

$$\mathcal{H}_1 = Pogona(ind)$$
$$\mathcal{H}_2 = PogonaMinor(ind)$$
$$\mathcal{H}_3 = \exists livesIn.AridWoodland(ind)$$
$$\mathcal{H}_4 = Pogona \sqcap \exists livesIn.AridWoodland(ind)$$

If the syntactic minimality criterion is applied, then $\mathcal{H}_4$ is not a valid solution to the abduction problem since the set of axioms in $\mathcal{H}_1$ is a subset of those in $\mathcal{H}_4$. The same is true for $\mathcal{H}_3$ and $\mathcal{H}_4$. This leaves $\mathcal{H}_1$, $\mathcal{H}_2$ and $\mathcal{H}_3$ as possible abductive solutions. If the semantic minimality criterion is applied instead, then $\mathcal{H}_4$ is still not a valid solution, since $\mathcal{O}, \mathcal{H}_4 \models \mathcal{H}_1$ and $\mathcal{O}, \mathcal{H}_4 \models \mathcal{H}_3$ but neither of the reverse cases hold. Similarly, $\mathcal{H}_2$ is not a semantically minimal hypothesis since $\mathcal{O}, \mathcal{H}_2 \models \mathcal{H}_1$ and $\mathcal{O}, \mathcal{H}_1 \not\models \mathcal{H}_2$. This leaves $\mathcal{H}_1$ and $\mathcal{H}_3$ as possible abductive solutions.

Therefore, Example 3.3.1 provides an example of a hypothesis ($\mathcal{H}_2$) that is syntactically minimal, but not semantically minimal. The reverse is also possible. To illustrate this, a given semantically minimal hypothesis could be conjunctively extended with a consequence of the ontology $\mathcal{O}$. In Example 3.3.1, extending $\mathcal{H}_1$ to obtain $Pogona(ind) \sqcap \exists livesIn.Habitat(ind)$ results in an example of a hypothesis that is semantically minimal, but not syntactically minimal.

In this section, common abductive constraints applied to hypotheses have been outlined, but no specific abduction problem has been defined utilising a certain set of constraints. The general abduction problem that is identified and solved in this thesis is discussed in Chapter 4, which will then provide a basis for extending the problem to more expressive forms of abduction in subsequent chapters.

Existing Approaches in DLs

A variety of work exists on abduction in DLs. Alongside studies of the complexity of abduction problems in given DL languages [Bie08], this work can be divided into approaches that concern a form of the TBox abduction problem, and those that focus on a form of the ABox abduction problem. As noted, though these approaches all perform a particular form of abduction, they often differ significantly in the actual problem being addressed. This is largely due to constraints placed on the language in which the background ontology, observations and hypotheses are specified, or due to abductive constraints placed on the hypotheses produced such as those discussed previously.

In the DL setting, ABox abduction has been applied to problems such as text interpretation [PKMM08] and query explanation [COSS13]. A number of works approach the task of ABox abduction via DL tableau-based approaches. Klarman et al [KES11] present a theoretical framework for ABox abduction in $\mathcal{ALC}$ relying on two reasoning techniques: regular connection tableaux and resolution with set-of-support. The approach restricts observations and hypotheses to be in the subset of $\mathcal{ALC}$ that excludes disjunction and uses negation only over concept names, namely $\mathcal{ALE}$. The issue of non-termination of the approach is discussed with respect to relaxing the minimality requirement or restricting TBoxes to be acyclic. The approach is shown to be sound and complete for this problem, though it is theoretical in nature, without an implementation, and no experimental evaluations are provided.

The connection tableau approach in [KES11] was performed directly using DL tableau by Halland and Britz [HB12]. The method relies upon the fact that if $\mathcal{O}, \mathcal{H} \models \psi$ for some observation ψ, then $\mathcal{O}, \mathcal{H}, \neg\psi \models \bot$. Tableau approaches are used to compute all models of $\mathcal{O}, \neg\psi$, while Reiter's minimal hitting set algorithm [Rei87] is used to generate abductive solutions by combining negated assertions from each of the given models to form a hypothesis. The approach computes consistent, relevant and syntactically minimal hypotheses in $\mathcal{ALE}$, and is sound for this problem but not complete. Semantically minimal explanations are not computed, but are noted to be of interest.

Building on the work in [HB12], Pukancova and Homola [PH17] extend the approach based on minimal-hitting sets, providing an implementation that utilises the DL reasoner Pellet [SPG$^+$07] as a black-box. As a result, the DL language used is restricted only by the reasoner, which in this case is $\mathcal{SROIQ}$. The approach supports atomic and negated atomic concept and role assertions, computing syntactically minimal hypotheses starting with the shortest explanations first, and is sound and complete for this problem. An experimental evaluation is provided in [PH18] over three small ontologies ranging from 24 axioms to 291 axioms in size. Explanations of up to length three were computed, after which point the approach is limited by memory usage.

Du et al [DWS14] developed an approach to ABox abduction where the observations take the form of Boolean conjunctive queries (BCQs) [COSS13] and the background knowledge takes the form of Datalog rewritable ontologies, which restrict ontologies to be expressed in Horn fragments of DLs. The approach can be applied to ontologies expressed in the Horn fragment of the DL $\mathcal{SHIQ}$. The hypotheses produced are syntactically minimal and a class of representative explanations is introduced to reduce the size of the hypothesis space. The method is sound and complete for this problem, and experimental results are provided over three ontologies for BCQs that are atomic and for general BCQs that allow existentially quantified variables.

TBox abduction in DLs has been applied to problems such as ontology repair [LDI12, WKDL14]. In Lambrix et al [LDI12] it is assumed that a set of required repairs have been detected, taking the form of missing is-a relations. The problem of finding ways to repair this missing information is formulated as TBox abduction problem, and an algorithm is presented to solve this problem in the setting of acyclic $\mathcal{ALC}$ TBoxes, where solutions are restricted to sets of atomic concept inclusions. In Wei-Kleiner et al [WKDL14] the problem of repairing missing is-a relations is similarly solved for $\mathcal{EL}$++ ontologies, where the aim is to compute hypotheses as sets of atomic concept inclusions that conform to a notion of semantic maximality, i.e., most specific explanations. This criterion is noted to be specifically relevant to the repair problem being addressed. The hypotheses produced are also required to be subset minimal. The algorithm presented is sound and an experimental evaluation is performed over three ontologies commonly used for reasoning competitions.

In Halland et al [HBK14], a TBox abduction method for $\mathcal{ALC}$ is presented that uses an approach based on the use of DL tableau and minimal hitting sets, similar to the approaches described for ABox abduction above. A fragment of $\mathcal{ALC}$ is specified that restricts solutions to using atomic negation, limited existential and universal restrictions and excludes both conjunction and disjunction. The approach is sound and complete, and a post-processing approach based on using DL tableau to test entailment between solutions is suggested to ensure that the solutions conform to a notion of semantic minimality, though no experimental evaluation is provided for the approach.

Du et al [DWM17] developed a TBox abduction approach that uses justification patterns, which can be constructed using fresh or existing concept names, to explain observations as sets of atomic concept inclusions. The explanations produced conform to a specified notion of subset minimality that places preference on fresh symbols. Empirical results are presented over a corpus of ten TBoxes.

Several points are of particular relevance to the work in this thesis. First, to the best of the authors' knowledge, an implemented and evaluated method for computing semantically minimal hypotheses in DLs has not yet been developed. Works that consider semantic minimality most often do so in the absence of disjunction in the computed hypotheses, restricting the problem to cases such as the one in Example 3.3.1. Permitting disjunction fundamentally changes the problem, and the notion of semantic minimality given in Definition 3.3.6 is not sufficient in this context since it permits an unbounded number of redundant explanations as additional disjuncts. Second, as noted earlier most works on abduction in DLs do not consider the ability to restrict hypotheses to a set of abducible symbols, in contrast to other areas such as ALP. The problem of KB abduction is also unsolved in the literature, where the problems of TBox and ABox abduction are treated separately. These factors imply that there is a need for more work on abduction in DL ontologies, and are part of the motivation behind the work in this thesis that is discussed in Chapter 4.

3.4 Second-Order Quantifier Elimination and Forgetting

The abduction approaches developed in this thesis utilise forgetting methods during one of the steps. Therefore, this section contains relevant background on the task of forgetting. Since the forgetting task itself is not the primary topic of this thesis, the material in this section focuses on notions of forgetting that are directly needed to discuss the proposed abduction problems and methods, as well as existing work on the connection between abductive reasoning and forgetting. The exact connection between abductive reasoning and forgetting is discussed in Section 3.6.

In the context of ontologies, *forgetting* is the process of finding a compact representation of an ontology by hiding or removing subsets of symbols within it. Here, the term *symbols* refers to concept and role names present in the ontology. The symbols to be hidden are specified in the *forgetting signature* $\mathcal{F}$, which is a subset of symbols in the ontology $\mathcal{O}$.

There are two notions of forgetting that are directly relevant to this thesis: *weak forgetting* [ZZ10] and *strong forgetting* [LR94, ZZ10]. Weak forgetting can be defined for first-order logic as follows [Koo15]:

Definition 3.4.1. *Let T be a formula expressed in first-order logic and $\mathcal{F}$ be a signature of predicate symbols. T' is a* solution of weakly forgetting *$\mathcal{F}$ in T iff for every formula G not containing any symbol in $\mathcal{F}$, $T \models G$ iff $T' \models G$.*

The notion of weak forgetting has been noted [Zha18] to be the dual notion of uniform interpolation [Hen63]. The duality lies in the fact that the uniform interpolant in the signature $\mathcal{S}$ for a formula T is equivalent to the result of forgetting the symbols in T that are outside of $\mathcal{S}$ via weak forgetting.

A reformulation of the modal logic definition for uniform interpolation in [Hen63] for description logics is as follows [Koo15]:

Definition 3.4.2. *Let $\mathcal{L}$ be a DL language, $\mathcal{O}$ be an ontology in $\mathcal{L}$ and $\mathcal{S} \subseteq sig(\mathcal{O})$*

be a signature. An ontology $\mathcal{V}$ is an $\mathcal{L}$ uniform interpolant *for the signature $\mathcal{S}$ iff the following two conditions hold:*

(i) $sig(\mathcal{V}) \subseteq \mathcal{S}$

(ii) *For every $\mathcal{L}$ axiom β with $sig(\beta) \subseteq \mathcal{S}$ we have $\mathcal{V} \models \beta$ iff $\mathcal{O} \models \beta$*

Given the duality to weak forgetting, Definition 3.4.2 can also be expressed in terms of a signature of forgetting symbols $\mathcal{F}$, where the result $\mathcal{V}$ should be expressed using only symbols in $sig(\mathcal{O}) \setminus \mathcal{F}$. The symbols in $\mathcal{F}$ should be removed from $\mathcal{O}$ while preserving all entailments of $\mathcal{O}$ that can be represented using the signature $sig(\mathcal{O})$ without $\mathcal{F}$. The result is a new ontology, $\mathcal{V}$, which is a *forgetting solution*. This dual perspective is the one used primarily throughout this thesis.

The second notion, called strong forgetting, can be defined as follows [LR94]:

Definition 3.4.3. *Let T be a theory in first-order logic and $\mathcal{F}$ be a set of predicates. A theory T' is* the result of strongly forgetting *$\mathcal{F}$ from T if for every interpretation M, $M \models T'$ iff there exists an interpretation $M' \models T$ such that M and M' differ only on the interpretations of the predicates in $\mathcal{F}$.*

Strong forgetting has also been referred to in the literature as *semantic forgetting* [EW08, ZS15]. To define strong forgetting in DLs, the notion of $\mathcal{F}$-equivalence is useful [ZS17, Zha18]. Let $\mathcal{I}$ and $\mathcal{I}'$ be two interpretations. Then $\mathcal{I}$ and $\mathcal{I}'$ are equivalent up to a set $\mathcal{F}$ of concept and role symbols, referred to as being $\mathcal{F}$-equivalent, if $\mathcal{I}$ and $\mathcal{I}'$ coincide but differ possibly in the interpretations of the symbols in $\mathcal{F}$. This means that both $\mathcal{I}$ and $\mathcal{I}'$ have the same domain $\Delta^{\mathcal{I}} = \Delta^{\mathcal{I}'}$ and interpret all individuals, and all concept and role symbols outside of $\mathcal{F}$, the same, i.e., $a^{\mathcal{I}} = a^{\mathcal{I}'}$ for every $a \in N_I$, $A^{\mathcal{I}} = A^{\mathcal{I}'}$ for every $A \in N_C$ that is not in $\mathcal{F}$ and $r^{\mathcal{I}} = r^{\mathcal{I}'}$ for every $r \in N_r$ that is not in $\mathcal{F}$. The definition for strong forgetting in DLs is then as follows [Zha18]:

Definition 3.4.4. *Let $\mathcal{L}$ be a DL language, $\mathcal{O}$ be an ontology and $\mathcal{F}$ be a set of concept and role symbols, called a forgetting signature, such that $\mathcal{F} \subseteq sig(\mathcal{O})$. An ontology $\mathcal{V}$ is an $\mathcal{L}$* semantic forgetting solution *for eliminating the symbols in $\mathcal{F}$ iff the following two conditions hold:*

(i) *$sig(\mathcal{V}) \subseteq sig(\mathcal{O}) \setminus \mathcal{F}$ and*

(ii) *For any interpretation $\mathcal{I}$: $\mathcal{I} \models \mathcal{O}'$ iff $\mathcal{I}' \models \mathcal{O}$ for some interpretation $\mathcal{I}'$ $\mathcal{F}$-equivalent to $\mathcal{I}$.*

The notion of semantic forgetting is closely related to the problem of second-order quantifier elimination. Second-order quantifier elimination is a generalisation of the problem of forgetting, in which the aim is to compute an equivalent first-order logic formula for a given second-order logic formula by eliminating (existentially) quantified predicate symbols. By eliminating these quantified predicate symbols, the resulting first-order formula preserves all entailments of (is equivalent to) the original second-order formula up to the interpretation of the predicate symbols that have been eliminated. Given this clear connection between second-order quantifier elimination and forgetting, the above proposal of applying second-order quantifier elimination techniques to the problem of abductive reasoning is also related to the application of forgetting to abduction. This connection is therefore relevant to the methods presented in this thesis.

In this thesis, unless stated otherwise, it is assumed that *forgetting* refers to the notion of weak forgetting, i.e., the dual of uniform interpolation. When it is necessary to make a distinction, primarily in Chapter 6, strong forgetting will be explicitly referred to as semantic forgetting.

3.5 Relevant Forgetting Approaches

There is a variety of work on forgetting in DLs [KWW09, LW11, LK14, KS15b, ZS16], a full survey of which is outside the scope of this thesis, which focuses on the problem of abductive reasoning. However, forgetting is utilised as part of the abduction approaches presented in this thesis. As such, the forgetting notions discussed in the previous section are linked directly to two systems that perform forgetting in the domain of DL ontologies. The specific motivations behind focusing on these two systems is discussed in Chapters 5–7. Here the focus is on presenting the calculi and discussing the key properties required for understanding the abduction methods presented in the rest of the thesis.

For weak forgetting or uniform interpolation, the resolution-based system LETHE [KS15b, Koo15] is investigated, while for semantic forgetting the system FAME [ZS15, ZS16, Zha18], which is based around the application of Ackermann's Lemma [Ack35], is used to provide a comparison between the two forms of forgetting. These two state of the art systems for performing forgetting in DL ontologies have shown promising performance over real world ontologies.

3.5.1 Forgetting: LETHE

The first relevant forgetting approach for this work takes the weak forgetting (uniform interpolation) perspective. The approach makes use of a resolution-based calculus [KS15b], referred to here as $Int_{\mathcal{ALC}}$, which is implemented in a system called LETHE [KS15a]. The calculus is utilised in the abduction approach of Chapter 5 and is extended as part of an approach for more expressive abduction problems in Chapter 7.

The calculus $Int_{\mathcal{ALC}}$ can be used to eliminate a set of symbols from a given $\mathcal{ALC}$ ontology, specified in a forgetting signature $\mathcal{F}$, which can include any concept or role symbols in the input ontology. Individual forgetting is not supported by the approach, and as such $\mathcal{F}$ cannot contain any individuals. The approach also supports forgetting for input ontologies with greatest fixpoints, i.e., those expressed in $\mathcal{ALC}\nu$, but for the context of this thesis it is sufficient to consider inputs without fixpoints.

Before the forgetting calculus $Int_{\mathcal{ALC}}$ can be applied, the input ontology must first be transformed to an appropriate normal form. In this normal form, each axiom takes the form of a role assertion $r(a,b)$, where r is a role symbol and a,b are individuals, or one of two forms of clauses [KS15b] as follows:

$$L_1(x) \vee ... \vee L_n(x)$$

$$L_1(a_1) \vee ... \vee L_n(a_n)$$

where each L_i is a concept literal, x is a variable and each a_i is a constant (individual). Concept literals L_i take one of the following forms: A, $\neg A$, $\exists r.D$, $\forall r.D$ where A is a concept name, r is a role name and D is a definer symbol. Definer symbols are *fresh* concept symbols, i.e., they are drawn from a separate set of symbols $D \in N_D$ that do not occur in the signature of the input ontology.

The first form of clause results from the transformation of a TBox axiom: a statement $C_1(x) \vee C_2(x)$ is equivalently representable as the TBox axiom $\top \sqsubseteq C_1 \sqcup C_2$, where x is introduced as part of the implicit universal quantification present in DL TBoxes. The second clausal form, resulting from the transformation of an ABox axiom, corresponds to the notion of a disjunctive ABox assertion. All clauses in the normal form are subject to the restriction that they may contain at most one literal of the form $\neg D(x)$, and no literal of the form $\neg D(a)$, where D is a definer symbol and a is an individual.

The procedure applied to obtain the normal form can be described as follows, where t is either a constant or the (implicit) variable x. Structural transformations are applied to flatten nested quantifiers by representing concepts that fall under the scope of a role restriction, i.e., a clause of the form $C_1 \sqcup \mathcal{Q}r.C_2(t)$, where $\mathcal{Q}$ denotes either $\exists$ or $\forall$, becomes the set of clauses $C_1 \sqcup \mathcal{Q}r.D_1(t), \neg D_1 \sqcup C_2(x)$ where D_1 is a definer symbol. Standard conjunctive normal form (CNF) flattening techniques are used to transform expressions of the form $C_1 \sqcup (C_2 \sqcap C_3)(t)$ into a set of clauses, interpreted conjunctively, of the form $\{C_1 \sqcup C_2(t), C_1 \sqcup C_3(t)\}$. The following example illustrates the transformation of an ontology into the required normal form:

Example 3.5.1. *Consider the following ontology $\mathcal{O}$:*

$$\begin{aligned}\mathcal{O} = \{&A \sqsubseteq \forall r.(E \sqcap C),\\ &B \sqsubseteq \forall s.F,\\ &C \sqsubseteq F,\\ &\exists r.\neg C(a),\\ &s(a,b)\}\end{aligned}$$

The transformation of $\mathcal{O}$ into the normal form results in the following clause set:

$$1)\neg A(x) \vee \forall r.D_1(x)$$

$$2)\neg D_1(x) \vee E(x)$$

$$3)\neg D_1(x) \vee C(x)$$

$$4)\neg B(x) \vee \forall s.D_2(x)$$

$$5)\neg D_2(x) \vee F(x)$$

$$6)\neg C(x) \vee F(x)$$

$$7)\exists r.D_3(a)$$

$$8)\neg D_3(x) \vee \neg C(x)$$

$$9)s(a,b)$$

To flatten the concept $\exists r.(C \sqcap D)$ in the first axiom of $\mathcal{O}$, the definer symbol D_1 is introduced beneath the existential role restriction, and clauses (2) and (3) are added. Similarly, D_2 is introduced to flatten $\forall s.F$. The remaining TBox axiom is expressed in clausal form. A third definer, D_3, is introduced to flatten the concept expression in first ABox axiom, while no further transformation is required for the second ABox axiom.

Once a given input ontology has been transformed into normal form, the calculus $Int_{\mathcal{ALC}}$ can be applied to eliminate symbols in the forgetting signature. This calculus is shown in Figure 3.2.

Given an input ontology $\mathcal{O}$ and a forgetting signature $\mathcal{F}$, the $Int_{\mathcal{ALC}}$ calculus is applied on the normal form of $\mathcal{O}$ until saturation. Resolution inferences are restricted to concepts in $\mathcal{F}$ or definer symbols, while the role propagation rule need only be applied when it is necessary to enable further resolution steps on symbols in $\mathcal{F}$ [Koo15]. The unifier σ in Figure 3.2 refers to a substitution between two terms t_1 and t_2 , which exists only if either t_1 or t_2 are the variable x, or if $t_1 = t_2$ [KS15b]. For example, if $t_1 = x$ and $t_2 = a$, then the unifier σ is the substitution that replaces occurrences of x with a.

Resolution:

$$\frac{C_1 \vee A(t_1) \qquad\qquad C_2 \vee \neg A(t_2)}{(C_1 \vee C_2)(\sigma)}$$

Role Propagation:

$$\frac{C_1 \vee (\forall r.D_1)(t_1) \qquad\qquad C_2 \vee Qr.D_2(t_2)}{(C_1 \vee C_2)(\sigma) \vee Qr.D_{12}(t_1\sigma)}$$

∃-Role Restriction Elimination:

$$\frac{C \vee (\exists r.D)(t) \qquad \neg D(x)}{C}$$

Role Instantiation:

$$\frac{C_1 \vee (\forall r.D)(t_1) \qquad\qquad r(t_2, b)}{C_1(\sigma) \vee D(b)}$$

D_1 and D_2 are definer symbols, Q $\in \{\forall, \exists\}$, t, t_1 and t_2 are terms (variable "x" or a constant), σ is the unifier of t_1 and t_2 if it exists, D_{12} is a new definer symbol for $D_1 \sqcap D_2$ and no clause contains more than one negative definer literal of the form $\neg D(x)$, and none of the form $\neg D(a)$.

Figure 3.2: $Int_{\mathcal{ALC}}$ rules [KS15b] utilised in the forgetting calculus of LETHE.

Example 3.5.2. *Consider the set of clauses (1)–(9) corresponding to the normalised ontology in Example 3.5.1. Let the forgetting signature be $\mathcal{F} = \{s, C\}$, then application of the $Int_{\mathcal{ALC}}$ calculus proceeds as follows:*

10)$\neg D_1(x) \vee F(x)$	*Resolution(3, 6)*
11)$\neg A(a) \vee \exists r.D_{13}$	*Role Propagation(1, 7)*
12)$\neg D_{13}(x) \vee D_1(x)$	
13)$\neg D_{13}(x) \vee D_3(x)$	
14)$\neg D_{13}(x) \vee E(x)$	*Resolution(2, 12)*
15)$\neg D_{13}(x) \vee C(x)$	*Resolution(3, 12)*

$16) \neg D_{13}(x) \vee \neg C(x)$	*Resolution(8, 13)*
$17) \neg D_{13}(x)$	*Resolution(15, 16)*
$18) \neg A(a)$	*∃-Role Restriction Elimination(11, 17)*
$19) \neg B(a) \vee D_2(b)$	*Role Instantiation(4, 9)*

An important aspect of the approach is that definer symbols are introduced only when necessary: if possible, previously introduced definers are reused. This ensures that there is a double exponential bound on the number of clauses derived, ensuring the termination of the approach [KS15b].

Once all possible inferences have been made, any clauses containing symbols in $\mathcal{F}$ are removed. Definer symbols are then eliminated using the rules in Figure 3.3.

Example 3.5.3. *Continuing from Examples 3.5.1 and 3.5.2, consider the full set of clauses (1) – (19). To extract the forgetting solution, first all clauses containing symbols in $\mathcal{F} = \{s, C\}$ are removed. The definer elimination rules in Figure 3.3 are then applied to the remaining clauses, resulting in the following forgetting solution $\mathcal{V}$:*

$$\mathcal{V} = \{A \sqsubseteq \forall r.(F \sqcap E), \neg A(a), \neg B(a) \vee F(b), \exists r.\top(a)\}$$

Once this is done, the result is an ontology $\mathcal{V}$ that is free of all symbols in $\mathcal{F}$. The ontology $\mathcal{V}$ may contain greatest fixpoints as a result of cyclic definer elimination and may contain disjunctive assertions due to the role instantiation rule. As a result, $\mathcal{V}$ is expressed in the DL $\mathcal{ALC}\nu(\vee)$. The use of fixpoints is also required to ensure that the result of forgetting can be finitely represented in the presence of cycles, as not all DLs have the uniform interpolation property [KWW08, LW11]. This means that the forgetting solution does not necessarily exist in these logics, which includes $\mathcal{ALC}$ without fixpoint operators [LW11, Koo15].

The key properties of the $Int_{\mathcal{ALC}}$ calculus with respect to the work in this thesis are the *soundness* and *interpolation completeness* of the approach. The soundness and interpolation completeness of $Int_{\mathcal{ALC}}$ with respect to forgetting over $\mathcal{ALC}\nu(\vee)$ have

Declausification:

$$\frac{\mathcal{O} \cup \{\neg D(x) \vee C_1(x), \ldots, \neg D(x) \vee C_n(x)\}}{\mathcal{O} \cup \{D \sqsubseteq C_1 \sqcap \ldots \sqcap C_n\}} \quad \text{if } D \text{ occurs only positively in } \mathcal{O}$$

Definer purification:

$$\frac{\mathcal{O}}{\mathcal{O}[D/\top]} \quad \text{if } D \text{ occurs only positively in } \mathcal{O}$$

Non-cyclic Definer Elimination

$$\frac{\mathcal{O} \cup \{D \sqsubseteq C\}}{\mathcal{O}[D/C]} \quad \text{if } D \notin sig(C)$$

Cyclic Definer Elimination

$$\frac{\mathcal{O} \cup \{D \sqsubseteq C[D]\}}{\mathcal{O}[D/\nu X.C[X]]} \quad \text{provided } D \in sig(C[D])$$

Figure 3.3: Definer elimination rules used in $Int_{\mathcal{ALC}}$ [Koo15], where $\mathcal{O}$ is an $\mathcal{ALC}\nu$ ontology without definer symbols, each C_i is an $\mathcal{ALC}$ concept, D is a definer symbol and $\nu X.C[X]$ represents a greatest fixpoint, where X is concept variable and $C[D]$ is a concept containing the definer symbol D.

been proven in [KS15b, Koo15].

Soundness of a calculus is defined according to the standard notion: the calculus should derive only those axioms that are entailed by the input. Soundness is defined below with respect to a set of clauses, i.e., the clausal form of an ontology $\mathcal{O}$ [KS15b].

Definition 3.5.1. *A calculus is sound if for any set of clauses $\mathcal{N}$ and any axiom α with $sig(\alpha)$, that does not contain fresh definer symbols, the result $\mathcal{V}$ of applying the calculus to $\mathcal{N}$ satisfies the following property: if $\mathcal{V} \models \alpha$ then $\mathcal{N} \models \alpha$.*

Completeness is defined in the context of strongest necessary conditions, since this is what is required in the context of abductive reasoning. Recalling from Definition 3.6.1, all forgetting solutions are also strongest necessary conditions. Thus, completeness of $Int_{\mathcal{ALC}}$ with respect to forgetting is defined as follows:

Definition 3.5.2. *A calculus is complete for forgetting in the language $\mathcal{L}$ if for any combination of an $\mathcal{L}$ ontology $\mathcal{O}$ and a forgetting signature $\mathcal{F}$, the result $\mathcal{V}$ of forgetting the symbols in $\mathcal{F}$ from $\mathcal{O}$ using the calculus is a forgetting solution, i.e., $\mathcal{V}$ is a strongest necessary condition of $\mathcal{O}$ in the signature $sig(\mathcal{O}) \setminus \mathcal{F}$.*

Further discussion of the $Int_{\mathcal{ALC}}$ calculus and the associated normalisation and definer elimination rules, including proofs, can be found in [KS15b, Koo15].

3.5.2 Semantic Forgetting: FAME

The second forgetting approach utilised in this thesis takes the semantic (strong) forgetting perspective and is implemented in a system called FAME [ZS16, Zha18].

The approach can be used to eliminate a set of concept and role symbols, specified in a forgetting signature $\mathcal{F}$, from a given ontology. As for the previous forgetting approach, $\mathcal{F}$ cannot contain individuals (nor nominals), i.e., individual forgetting is not supported. For the context of this thesis, it is not necessary to utilise the full expressivity of the forgetting approach discussed in [ZS16], since the main aim is to solve a new form of abduction problem in the DL $\mathcal{ALC}$. As such it is sufficient to consider $\mathcal{ALCO}(\nabla)$, that is $\mathcal{ALC}$ extended with nominals ($\mathcal{O}$) and the universal role (∇), ontologies as input. The use of nominals is due to the fact that the approach of FAME requires that all ABox axioms be converted to equivalent TBox axioms. Meanwhile, the use of the top role ∇ relates to the abduction problem. These language considerations are discussed in Chapter 6.

The forgetting approach of FAME is based around Ackermann's Lemma [Ack35]. The original Ackermann's Lemma has been utilised in the task of second-order quantifier to eliminate single predicate symbols from formulae expressed in first-order logic. As such, the process of eliminating a set of symbols specified in a forgetting signature using FAME is reduced to the problem of eliminating one symbol at a time from the set. The symbol to be eliminated on a given iteration is referred to as the *pivot*, where the order of elimination is based upon a heuristic analysis of the frequency of occurrence of each symbol in the input [ZS16].

For both concept and role forgetting, the procedure follows several main phases: clausification, normalisation, forgetting and definer elimination. The latter three phases are repeated for each symbol in the forgetting signature. While a full formal description of each of these phases is outside of the scope of this work, an overview is provided

Surfacing^C:

$$\frac{\mathcal{N}, C \sqcup \forall r.D}{\mathcal{N}, (\forall r^{-}.C) \sqcup D}$$

where: (i) $A \in sig_C(\mathcal{N})$ is the pivot concept.
(ii) A does not occur in the concept C.
(iii) If A occurs *positively* in D, the rule is the Surfacing$^{C,+}$ rule.
If A occurs *negatively* in D, then the rule is the Surfacing$^{C,-}$ rule.

Skolemisation^C:

$$\frac{\mathcal{N}, \neg\{a\} \sqcup \exists r.D}{\mathcal{N}, \neg\{a\} \sqcup \exists r.\{b\}, \neg\{b\} \sqcup D}$$

where: (i) $A \in sig_C(\mathcal{N})$ is the pivot concept.
(ii) $\{\mathrm{b}\}$ is a fresh nominal.
(iii) If A occurs *positively* in D, the rule is the Skolemisation$^{C,+}$ rule.
If A occurs *negatively* in D, then the rule is the Skolemisation$^{C,-}$ rule.

Figure 3.4: Surfacing and skolemisation rules used to compute the A-reduced form in the forgetting system FAME [Zha18].

to give context to the forgetting procedure utilised by FAME so that the impact on the hypotheses obtained from the forgetting-based abduction algorithm can be understood. Here, the focus is on describing the normalisation and forgetting process for concept forgetting, as this captures the main notions required for an intuitive understanding of the forgetting procedure. Full details and proofs for these phases are available in the corresponding works [ZS16, Zha18]. When a concept C is referred to as being *negative* with respect to a concept A, this means that A occurs only *negatively* in C, i.e., under an odd number of negations. If C is *positive* with respect to A, then A occurs under an even number of negations.

The first stage in this transformation is the *clausification* phase, during which any ABox axioms in the input are translated to equivalent TBox axioms in $\mathcal{ALCO}$, making use of nominals. For example, the ABox axioms $C(a)$ and $r(a,b)$ can be transformed into the corresponding TBox axioms $\{a\} \sqsubseteq C$ and $\{a\} \sqsubseteq \exists r.\{b\}$. Then, the resulting

TBox is transformed into a set of clauses $\mathcal{N}$ using standard clausification rules.

Following this, it is necessary to transform the set of clauses $\mathcal{N}$ into the appropriate *reduced forms*, based upon the current pivot. This is done during the *normalisation* phase. Depending on whether the current pivot is either a concept symbol A or a role symbol r, the corresponding reduced form is referred to as the A*-reduced* or the r*-reduced* form respectively. For the A-reduced form, it is necessary to move the pivot A to the top-level of each clause, i.e., ensure that A is not nested beneath existential or universal quantifiers. The rules applied to obtain the A-reduced form are provided in Figure 3.4. Note that if A does not occur beneath an existential or universal quantifier in a clause, then it already occurs at the top-level of the clause. Thus, no additional rule is required to compute the A-reduced form in these cases. The following is an example of transforming an ontology into A-reduced form:

Example 3.5.4. *Consider the ontology $\mathcal{O} = \{B \sqsubseteq A, C \sqsubseteq \forall r.A\}$. Transforming this ontology into clausal form results in:*

$$\neg B \sqcup A$$

$$\neg C \sqcup \forall r.A$$

while the first clause is already in A-reduced form, the second is not: A is nested beneath a universal quantifier. Application of the Surfacing$^{C,+}$ rule to the second clause results in the equivalent clause $(\forall r^-.\neg C) \sqcup A$, where r^- denotes the inverse of the role symbol r. The resulting set of clauses is then in A-reduced form.

Once the A-reduced form is obtained, the *forgetting* phase is performed to eliminate the concept symbol A. The idea behind the concept forgetting procedure in FAME is based on substituting an appropriate definition for each occurrence of the pivot A in the clause set $\mathcal{N}$, thereby eliminating it from the input set. An appropriate definition for the pivot A is, informally, a concept that *defines* A without using A itself [Zha18]. The rules used to eliminate the pivot A are shown in Figure 3.5, and are based upon a generalisation of Ackermann's Lemma [NS95] to the setting of DL ontologies [ZS16, Zha18]. This generalisation can be expressed for generic DL ontologies as follows [Zha18]:

Theorem 3.5.1. *Let $\mathcal{O}$ be a DL ontology containing the axioms $C_1 \sqsubseteq A, \ldots, C_n \sqsubseteq A$ where A is a concept symbol and each C_i for $1 \leq i \leq n$ is a DL concept that does not contain A. Let $\tilde{\mathcal{O}} = \mathcal{O} \setminus \{C_1 \sqsubseteq A, \ldots, C_n \sqsubseteq A\}$ denote the result of excluding the axioms $C_1 \sqsubseteq A, \ldots, C_n \sqsubseteq A$ from $\mathcal{O}$. If $\tilde{\mathcal{O}}$ is* negative *with respect to A, then $\tilde{\mathcal{O}}^A_{C_1 \sqcup \ldots \sqcup C_n}$ is a solution of forgetting the symbol A from $\mathcal{O}$, where $\tilde{\mathcal{O}}^A_{C_1 \sqcup \ldots \sqcup C_n}$ denotes the result of replacing every occurrence of A with $C_1 \sqcup \ldots \sqcup C_n$ in $\tilde{\mathcal{O}}$.*

The generalisation in Theorem 3.5.1 leads to the *Ackermann*$^{C,+}$ rule of Figure 3.5, while the dual theorem in [Zha18] leads to the *Ackermann*$^{C,-}$ rule.

Once the pivot has been eliminated, the process is repeated for each remaining concept symbol in the forgetting signature.

The procedure for role forgetting, including the transformation of the clause set $\mathcal{N}$ into r-reduced form for a pivot r, is conceptually similar. However, obtaining a

Ackermann$^{C,+}$**:** $\dfrac{\mathcal{N} \setminus \{C_1 \sqcup A, \ldots, C_n \sqcup A\}, C_1 \sqcup A, \ldots, C_n \sqcup A}{\mathcal{N} \setminus \{C_1 \sqcup A, \ldots, C_n \sqcup A\}^A_{\neg C_1 \sqcup \ldots \sqcup \neg C_n}}$

where (i) A does not occur in any C_i for $1 \leq i \leq n$
(ii) $\mathcal{N} \setminus \{C_1 \sqcup A, \ldots, C_n \sqcup A\}$ is negative with respect to A

Ackermann$^{C,-}$**:** $\dfrac{\mathcal{N} \setminus \{C_1 \sqcup \neg A, \ldots, C_n \sqcup \neg A\}, C_1 \sqcup \neg A, \ldots, C_n \sqcup \neg A}{\mathcal{N} \setminus \{C_1 \sqcup \neg A, \ldots, C_n \sqcup \neg A\}^A_{C_1 \sqcap \ldots \sqcap C_n}}$

where (i) A does not occur in any C_i for $1 \leq i \leq n$
(ii) $\mathcal{N} \setminus \{C_1 \sqcup A, \ldots, C_n \sqcup A\}$ is positive with respect to A

Purify$^{C,+}$**:** $\dfrac{\mathcal{N}}{\mathcal{N}^A_{\bot}}$

where $\mathcal{N}$ is negative with respect to A

Purify$^{C,+}$**:** $\dfrac{\mathcal{N}}{\mathcal{N}^A_{\top}}$

where $\mathcal{N}$ is positive with respect to A

Figure 3.5: Rules used in the forgetting system FAME to forget a pivot concept A, where $\mathcal{N}^A_C$ denotes the set of clauses obtained from $\mathcal{N}$ by replacing every occurrence of A with C [Zha18].

definition for r is less straightforward. This is due to the fact that role symbols in DLs always occur after an existential or universal role restriction, so it is not obvious how the pivot symbol r can be brought to the top-level of a clause [Zha18]. Thus, the approach used in FAME relies on combining all of the premises of a role restriction to obtain an implicit definition for the pivot r. Details of the full procedure can be found in [ZS16, ZS17, Zha18], while here the role forgetting procedure is described and illustrated by example to avoid deviating from the focus on abduction in this work.

For role forgetting, the *r-reduced* form is as follows. Given a role symbol to be forgotten r, every clause in $\mathcal{N}$ that contains r must be of the form $C \sqcup \forall r.D$ or $C \sqcup \neg\forall r.D$ where C and D are possibly complex concepts that do not contain r. During this transformation, definer symbols may need to be introduced as in Section 3.5.1. Definer symbols are used to incrementally replace the concept symbols C and D in clauses such as the one above, until neither clause contains r. For example:

Example 3.5.5. *Consider the following clause:*

$$\forall r.A \sqcup \forall r.B$$

given a forgetting signature $\mathcal{F} = \{r\}$, a definer is introduced to replace $\forall r.A$. This results in the following two clauses:

$$D_1 \sqcup \forall r.B$$
$$\neg D_1 \sqcup \forall r.A$$

Once the r-reduced form is obtained, as with the concept forgetting procedure described earlier, rules based around Ackermann's Lemma [Ack35] are used to forget the role symbols in the forgetting signature $\mathcal{F}$. Once all role symbols have been eliminated, any definer symbols that were introduced during the transformation to r-reduced form are then eliminated via the use of concept forgetting as described above.

The approach can be used to perform forgetting for $\mathcal{ALCOIH}\mu^+(\nabla,\sqcap)$, i.e., $\mathcal{ALC}$ extended with nominals ($\mathcal{O}$), inverse roles ($\mathcal{I}$), role hierarchies ($\mathcal{H}$), fixpoints (μ and

ν), the top role (∇) and role conjunction ($\sqcap$) [ZS16]. However, for the abduction problems that are solved in this thesis, the full expressivity is not needed. In fact, the input provided to the forgetting approach can be expressed in $\mathcal{ALCO}(\nabla)$, the reasoning behind which is discussed in the context of abduction in Chapter 6. Consequently, the result of semantic forgetting in this context can be represented in $\mathcal{ALCOI}(\nabla)$, due to the fact that role hierarchies of the form $r \sqsubseteq s$ are not present in the input, and role conjunction $r \sqcap s$ is only required in the event that the forgetting result utilises role hierarchies [Zha18]. Therefore, the forgetting approach used by FAME is described here for TBoxes only, while RBoxes can be excluded. While support for fixpoint operators was discussed in [ZS16], later versions of the tool, and described in [Zha18], did not support fixpoints. As such, fixpoints are not present in the forgetting solutions produced by the version of the tool used in this work. Therefore, the use of FAME in this work is restricted to input ontologies expressed in $\mathcal{ALCO}$, while the forgetting solution is expressed in $\mathcal{ALCOI}(\nabla)$.

The forgetting approach of FAME is sound for forgetting in $\mathcal{ALCOIH}\mu^+(\nabla,\sqcap)$ ontologies. However, only forgetting solutions expressed in DLs up to $\mathcal{ALCOI}(\nabla)$ will be required in this work. The soundness of the forgetting approach in this setting is stated in Theorem 3.5.2, which is a weaker form of the corresponding theorem in [ZS16].

Theorem 3.5.2. *For any $\mathcal{ALCOI}(\nabla)$ ontology $\mathcal{O}$ and any signature $\mathcal{F} \subseteq sig(\mathcal{O})$, where $sig(\mathcal{O})$ is the set of concept and role symbols in $\mathcal{O}$, FAME always terminates and returns a set $\mathcal{V}$ of clauses. If $\mathcal{V}$ does not contain any symbols in $\mathcal{F}$, then the symbols in $\mathcal{F}$ were successfully forgotten and the set $\mathcal{V}$ is a solution of forgetting the symbols in $\mathcal{F}$ from $\mathcal{O}$.*

Proof: The original theorem proving the soundness of FAME for the semantic forgetting problem [ZS16] holds for $\mathcal{ALCOIH}\mu^+(\nabla,\sqcup)$. Since $\mathcal{ALCOI}(\nabla)$ is a fragment of $\mathcal{ALCOIH}\mu^+(\nabla,\sqcup)$ excluding fixpoints, role inclusions of the form $r \sqsubseteq s$ and role conjunctions of the form $r \sqcap s$, the original soundness proof also holds in this setting.

The forgetting approach of FAME is, however, not complete [Zha18]. This is due to the known incompleteness of Ackermann-based approaches to forgetting. In general, it is not possible to bring every set of DL clauses into a form where a forgetting symbol can be successfully eliminated [ZS16, ZS17]. For instance, the Skolemisation^C rule applies only to clauses of a specific form: $\neg\{a\} \sqcup \exists r.D$, where $\{a\}$ is a nominal, unlike the Surfacing^C rule. Therefore, it cannot be applied to surface a pivot symbol beneath an existential restriction for all clauses. As such, it is not possible to eliminate the pivot symbol in these cases, since the forgetting rules in Figure 3.5 assume that the input has been transformed into A-reduced form.

3.6 Forgetting and Abductive Reasoning

The connection between forgetting and abductive reasoning has been identified in classical logics [DLS01, GSS08, Wer13] with respect to the related problem of second-order quantifier elimination. Specifically, the notions of strongest necessary conditions and weakest sufficient conditions [Lin01, DLS01] form the basis of the connection between the two tasks. In [Lin01], for propositional theories it was observed that the weakest sufficient condition corresponds to the notion of a weakest abductive explanation. The fact that second-order quantifier elimination has been proposed as an efficient method for computing weakest sufficient and strongest necessary conditions of propositional and first-order theories in [DLS01] establishes the connection between second-order quantifier elimination and abductive reasoning. The use of this connection was expanded upon in subsequent work including [GSS08, Wer13] and utilised in applications such as the analysis of Biochemical Pathways [DKMS04].

This relationship between abduction and forgetting has been a topic of interest in AI. However, the connection has been under explored, particularly in the domain of DL ontologies. Further work is needed to develop this connection before it can be utilised as part of practical abductive reasoning systems in the DL setting.

In DLs, necessary and strongest necessary conditions can be defined as follows:

Definition 3.6.1 (Strongest Necessary Conditions)**.** *Let $\mathcal{O}$ be an ontology, G be a set* *of axioms and $\mathcal{S}$ be a signature of concept and role symbols. A set of axioms T, where $sig(T) \subseteq \mathcal{S}$, is a necessary condition of G under $\mathcal{O}$ in the signature $\mathcal{S}$ iff $\mathcal{O}, G \models T$. It is also a strongest necessary condition if for all other necessary conditions T' such that $sig(T') \subseteq \mathcal{S}$, it is the case that $\mathcal{O}, T \models T'$.*

The definition for sufficient and weakest sufficient conditions is as follows:

Definition 3.6.2 (Weakest Sufficient Conditions)**.** *Let $\mathcal{O}$ be an ontology, G be a set of axioms and $\mathcal{S}$ be a signature of concept and role symbols. A set of axioms T, where $sig(T) \subseteq \mathcal{S}$, is a sufficient condition of G under $\mathcal{O}$ in the signature $\mathcal{S}$ if $\mathcal{O}, T \models G$. It is also a weakest sufficient condition if for all other sufficient conditions T' such that $sig(T') \subseteq \mathcal{S}$, it is the case that $\mathcal{O}, T' \models T$.*

Since the aim of forgetting is to retain all of the entailments representable in the signature $\mathcal{S}$, the forgetting result or uniform interpolant must by definition be the strongest entailment of the input ontology within the specified signature $\mathcal{S}$. It is therefore possible to formulate the definition of forgetting or uniform interpolation in terms of strongest necessary conditions [Koo15]. In general, it holds that:

Theorem 3.6.1. *$\mathcal{V}$ is a uniform interpolant (forgetting solution) of ontology $(\mathcal{O}, G)$ for $\mathcal{S}$ only if $\mathcal{V}$ is a strongest necessary condition of G under $\mathcal{O}$ in $\mathcal{S}$.*

Proof: Let $\mathcal{V}$ be a uniform interpolant of $(\mathcal{O}, G)$ in $\mathcal{S}$.[1] It holds that $\mathcal{V} \models \mathcal{V}$ for any $\mathcal{V}$. Recalling from Definition 3.4.2, since $sig(\mathcal{V}) \subseteq \mathcal{S}$, then $\mathcal{O}, G \models \mathcal{V}$ by the reverse direction of Definition 3.4.2 condition (ii), and so $\mathcal{V}$ is a necessary condition of $(\mathcal{O}, G)$. To show that the uniform interpolant $\mathcal{V}$ is a *strongest* necessary condition of G under $\mathcal{O}$ in $\mathcal{S}$, let $\mathcal{V}'$ be any set of axioms such that $\mathcal{O}, G \models \mathcal{V}'$ and $sig(\mathcal{V}') \subseteq \mathcal{S}$. From this, it follows by Definition 3.4.2 condition (ii) that $\mathcal{V} \models \mathcal{V}'$ and thus $\mathcal{O}, \mathcal{V} \models \mathcal{V}'$ for any $\mathcal{V}'$. Therefore, $\mathcal{V}$ is a strongest necessary condition of G under $\mathcal{O}$ in $\mathcal{S}$.

As discussed in [Lin01], strongest necessary and weakest sufficient conditions are dual notions, i.e., negating the strongest necessary condition gives the weakest sufficient condition of the negated formula and vice versa. It is in this duality that the

[1]Note: from the forgetting perspective, the input $(\mathcal{O}, G)$ is provided as a single ontology.

connection between forgetting and abduction is made via the use of contraposition:

$$\mathcal{O}, \mathcal{H} \models \psi \qquad \text{iff} \qquad \mathcal{O}, \neg\psi \models \neg\mathcal{H}$$

where $\mathcal{H}$ is a hypothesis for the observation ψ. Since forgetting produces a set of entailments of a given input ontology, by setting this input to $\mathcal{O}, \neg\psi$ the forgetting solution $\mathcal{V}$ corresponds directly to the negation of a candidate hypothesis $\neg\mathcal{H}$. Since this set of entailments is also the strongest necessary condition of the input, i.e., $\mathcal{V}$ is the strongest encessary condition of $\neg\mathcal{H}$ under $\mathcal{O}$, the hypothesis obtained via contraposition corresponds to the weakest sufficient condition of ψ under $\mathcal{O}$.

Depending on the form taken by ψ, it may be the case that the negation $\neg\psi$ must be represented using an extended DL language. For example, given an $\mathcal{ALC}$ observation $\psi = C(a) \sqcap D(b)$, the negation $\neg\psi$ is not representable in $\mathcal{ALC}$. Instead, the negation takes the form of a disjunctive assertion $\neg\psi = \neg C(a) \vee \neg D(b)$. This is discussed further in subsequent chapters.

3.7 Why Forgetting?

The connection between forgetting and abductive reasoning provides a promising basis. First, forgetting provides a goal-oriented way to produce hypotheses that conform to the notion of *abducibles* as discussed in Section 3.3. The set of symbols given as input, the forgetting signature, are eliminated and excluded from the forgetting solution. The forgetting signature is therefore a set of *non-abducible* symbols, and can be used to restrict the signature of the hypotheses produced. The possibility of generating hypotheses that take the form of weakest sufficient conditions is also an interesting prospect, as this captures a core notion of abductive reasoning: producing a *least assumptive* hypothesis given the available evidence in the background theory. For the general application of abduction it is natural to assume only that which is necessary to explain the observation. If we do not immediately assume that the first explanation computed is the correct one, then producing the least assumptive hypothesis first ensures that we do not lose the correct explanation by assuming too much initially [Poo89]. By starting at the weakest explanation first, it is also possible to iteratively apply abduction to obtain stronger explanations, which is discussed in Chapter 8.

In the DL context, weakest sufficient conditions coincide with the notion of semantic minimality as in Definition 3.3.6. It should be noted that semantic minimality is a difficult constraint to satisfy, particularly comparated to a constraint such as the syntactic minimality requirement in Definition 3.3.5. Using a generate and test approach, i.e., generating a large number of candidate hypotheses and then eliminating those that do not satisfy the requirements, is in general infeasible when one of the requirements is semantic minimality. To eliminate non-semantically minimal hypotheses would require comparing each pair of candidate hypotheses via entailment: if a given hypothesis is stronger than another, it should be removed. Depending on the complexity of entailment checking in the given DL language, this can become expensive. For example, in $\mathcal{ALC}$ each of these checks would have exponential complexity [Sch94] with respect to the size of the background ontology together with the candidate hypothesis. Therefore, the fact that forgetting together with contraposition can be used to produce semantically minimal hypotheses directly is promising.

However, before this can be used as a starting point to build a practical abduction approach for DL ontologies, a number of challenges must be addressed and solved. This includes identifying a new DL abduction problem, which is the subject of the next chapter.

Chapter 4

Computing Spaces of Independent Explanations

In this chapter, a new abduction problem in the DL setting is defined, where the input ontology and observation are expressed in the DL language $\mathcal{ALC}$. When utilised in an abductive reasoning system, forgetting can provide a starting point for the computation of hypotheses satisfying an otherwise difficult property: semantic minimality. However, the connection between forgetting and abduction has not been sufficiently investigated in the setting of large DL ontologies. For example the problem of redundancy, particularly in the presence of disjunction, is a critical issue that must be overcome. The problem identified in this chapter therefore captures the notion of semantic minimality while permitting disjunction in the abduction solutions, which has not yet been considered in the area. In doing so, the problem offers a perspective on abduction that is new to the setting of DL ontologies. This chapter also motivates the problem by examining the constraints placed on the hypotheses and how these improve the solutions returned.

The problem proposed in this chapter is the core problem solved throughout this thesis. Using the definition provided in this chapter as a basis, subsequent chapters will focus on proposed methods to solve this problem as well as extensions to more expressive abduction problems.

4.1 Challenges

A key difference between the work on second-order quantifier elimination approaches to abduction and the work in this thesis is in the setting of the problem, which brings a number of new challenges. Unlike many of the problems considered in classical logics, DL ontologies are often large by comparison. A consequence of this is that the problem of eliminating redundant parts of the hypothesis becomes more important than in the domain of classical logics, where this problem is either not considered or regarded as an optional post-processing step [Wer13]. This is particularly true if the weakest sufficient condition is interpreted as a disjunction of possible explanations for the given observation, which is the perspective taken in this thesis and in works such as [DKMS04]. In the domain of large ontologies, it is not sufficient to present the negation of a forgetting solution (via contraposition) as a hypothesis for the given observation. This is due to the fact that a majority of the given disjuncts will be redundant in explaining the observation, either because they are consequences of the ontology itself or because they repeat information that is already contained in other disjuncts.

In the DL setting, despite the recent development of effective forgetting procedures for expressive DLs [KWW09, LK14, KS15b, ZS16] including those discussed in the previous chapter, the connection between forgetting and abductive reasoning has yet to be sufficiently studied or utilised. In [KS15a], it was proposed that the forgetting system LETHE could be utilised for a form of TBox (universally quantified) abduction. However, this proposal focuses only on directly negating the result of forgetting to produce a weakest hypothesis that entails a given observation within a restricted signature of abducible symbols, i.e., those that have not been forgotten. Additionally, the use of approaches that utilise forgetting for ABox abduction was identified as an open problem.

Given these considerations, it is clear that progress must be made in addressing several key issues with utilising forgetting for abduction. First, the identification and definition of a new abduction problem is required, including an appropriate form for hypotheses and constraints to capture the issue of redundancies in the hypotheses. Second, the formalisation and development of methods to solve the abduction problem, utilising not only forgetting but also efficient approaches to redundancy elimination to ensure tractability over large DL ontologies and the computation of hypotheses that are not misleading in practice.

The remainder of this chapter will focus on the first point: defining and motivating an abduction problem which is new to the DL setting that takes into account the above considerations. Subsequent chapters will then focus on presenting new methods for abductive reasoning in DLs that produce hypotheses satisfying this problem.

4.2 Defining the Problem: Abductive Constraints

The abduction problem that is proposed and solved in this work is defined below for ontologies and observations expressed in the DL language $\mathcal{ALC}$.

Definition 4.2.1. ***ABox Abduction in $\mathcal{ALC}$ Ontologies.*** *Let $\mathcal{O}$ be an ontology and ψ be a set of concept assertions, both expressed in $\mathcal{ALC}$, such that $\mathcal{O} \not\models \bot$, $\mathcal{O}, \psi \not\models \bot$ and $\mathcal{O} \not\models \psi$. Let $\mathcal{S}_A$ be a set of symbols called* abducibles. *The* ABox abduction problem *is to compute a hypothesis of the form $\mathcal{H} = \alpha_1 \vee ... \vee \alpha_n$ where each α_i for $1 \leq i \leq n$ is a conjunction of $\mathcal{ALC}$ concept assertions. The solution $\mathcal{H}$ must contain only those symbols specified in $\mathcal{S}_A$ and satisfy the following conditions:*

(i) $\mathcal{O}, \mathcal{H} \not\models \bot$

(ii) $\mathcal{O}, \mathcal{H} \models \psi$

(iii) $\mathcal{H}$ does not contain inter-disjunct *redundancy i.e., there is no disjunct α_i in $\mathcal{H}$ such that $\mathcal{O}, \alpha_i \models \alpha_1 \vee ... \vee \alpha_{i-1} \vee \alpha_{i+1} \vee ... \vee \alpha_n$*

(iv) for any $\mathcal{H}'$ expressed in the same language and form as $\mathcal{H}$ that satisfies conditions (i)–(iii), where $sig(\mathcal{H}') \subseteq \mathcal{S}_A$, if $\mathcal{O}, \mathcal{H} \models \mathcal{H}'$ then $\mathcal{O}, \mathcal{H}' \models \mathcal{H}$.

Since the task is to compute a hypothesis satisfying some constraints within a given signature of abducibles $\mathcal{S}_A$, the following condition is imposed:

Abducibles: the hypothesis should be expressed using only those symbols that appear in the set of "allowed symbols", called *abducibles*.

For the problems in this thesis, *it is assumed that the set of abducibles contains only concept and role symbols. It is assumed that the hypotheses produced can contain any individuals specified in the ontology $\mathcal{O}$ or the observation* ψ, i.e., while concept and role symbols can be specified as non-abducible, specific individuals cannot.

The abduction problem is therefore to compute a hypothesis that satisfies the conditions (i)–(iv) with respect to the given signature of abducibles $\mathcal{S}_A$. The constraints specified by Definition 4.2.1 conditions (i)–(iv) can be described as follows:

(i) **Consistency:** the hypothesis should not contradict the information contained within the background knowledge base.

(ii) **Explanation:** adding the hypothesis to the background knowledge base should lead to the entailment of the observation.

(iii) **Inter-disjunct Redundancy**: the hypothesis produced should not contain disjuncts that are redundant, with respect to the abduction problem, given the rest of the hypothesis. If the hypothesis satisfies this condition, then it is said to be a *space of independent explanations*.

(iv) **Semantic Minimality:** the hypothesis should not assume more than is necessary to explain the observation. This can be described as follows: a hypothesis is only considered to be a solution to the abduction problem if all other possible hypotheses, expressed using the same set of abducibles, are either stronger, i.e., more assumptive, than it or are at least equivalent under the background knowledge.

Conceptually, this task can be seen as computing the least assumptive space of independent explanations [Kon92] $\alpha_1, ..., \alpha_n$ for the given observation, rather than a single hypothesis. Each explanation (disjunct) takes the form of a conjunction of $\mathcal{ALC}$ concept assertions, i.e., an ABox. Since the overall hypothesis is a disjunction of these explanations, it is necessary to make use of an extended language compared to the input, i.e., the extension of $\mathcal{ALC}$ that permits disjunctions of ABoxes denoted here as $\mathcal{ALC}(\vee)$.

This problem forms the basic problem that is solved throughout this thesis. In subsequent chapters, this is extended to more expressive abduction problems,[1] though the core of the task remains the same.

4.3 Examining the Constraints in Practice

Since this abduction problem is new to DLs, it is important to examine and motivate the constraints specified in the definition in terms of their impact on the types of hypotheses that are permitted as solutions.

As discussed in Chapter 3, the set of abducibles $\mathcal{S}_A$ defines the subset of concept and role symbols in the ontology that may appear in the hypothesis $\mathcal{H}$. Restricting the hypothesis to a subset of the available signature, while not often considered in the setting of description logic ontologies, is a standard condition in many works on abductive reasoning. This restriction allows a user to restrict acceptable hypotheses based on their own prior knowledge, focusing on parts of the hypothesis space that are useful to the given application. In some cases such as diagnostics, ontologies may be engineered with "causes" and "effects" in mind. Thus, it may be useful to restrict the set of abducibles to symbols representing the available causes rather than simply explaining one effect in terms of another.

Conditions (i) and (ii) of Definition 4.2.1 are standard requirements in most abductive reasoning tasks. Condition (i) requires that all generated hypotheses $\mathcal{H}$ are consistent with the background knowledge in the ontology $\mathcal{O}$. Otherwise the ontology obtained by adding $\mathcal{H}$ to $\mathcal{O}$ will entail $\bot$ and as a result will also trivially entail everything including the observation ψ. Condition (ii) ensures that adding $\mathcal{H}$ to $\mathcal{O}$ leads to

[1]This includes: permitting role assertions in both observations and hypotheses, extending the language used to express each explanation (disjunct) in the hypothesis and lifting the problem from ABox abduction to TBox / Knowledge Base (KB) abduction.

the entailment of the observation ψ, and thus that $\mathcal{H}$ is an explanation for ψ as intuitively expected for abductive reasoning. Without these two constraints, the space of possible solutions will consist mostly of uninformative hypotheses.

A significant difference between the notion of semantic minimality discussed in Chapter 3 and here is the presence of disjunctions, particularly in the hypotheses produced. Condition (iii), inter-disjunct redundancy, captures this difference and successfully resolves a shortcoming in the notion of semantic minimality in this setting. Therefore, before considering condition (iii) it is useful to first examine condition (iv), semantic minimality [KES11, HB12, HBK14], to provide context to the interaction between these two conditions.

As discussed, producing hypotheses that make only the fewest assumptions necessary given some background knowledge is of interest both intuitively [Poo89] and in applications [Sti91, PH17]. This corresponds to the notion of semantic minimality as in Definition 3.3.6, which is captured by condition (iv), where the aim is to produce the weakest hypothesis $\mathcal{H}$ required to explain the observation ψ under the background ontology $\mathcal{O}$. The following example illustrates this notion.

Example 4.3.1. *Consider the following abduction problem:*

$$\begin{aligned}\mathcal{O} = \{&A \sqsubseteq B,\\ &B \sqsubseteq C\}\\ \psi =& C(a)\\ \mathcal{S}_A = \{&A, B\}\end{aligned}$$

and the two candidate hypotheses:

$$\begin{aligned}\mathcal{H}_1 &= B(a)\\ \mathcal{H}_2 &= A(a)\end{aligned}$$

Both of these hypotheses satisfy the conditions in Definition 4.2.1(i) and (ii). However,

$\mathcal{H}_2$ does not satisfy condition (iv), since $\mathcal{O},\mathcal{H}_2 \models \mathcal{H}_1$, but the reverse does not hold. Thus $\mathcal{H}_2$ is a stronger, i.e., "less minimal" hypothesis than $\mathcal{H}_1$.

From this, it can be seen that Definition 4.2.1(iv) rejects semantically stronger hypotheses, formalising the idea of not assuming more than is necessary to explain a given observation. Note that stronger hypotheses can be sought in a principled fashion via the specification of abducible symbols. For example, if the set of abducibles in the above example was instead $\mathcal{S}_A = \{A\}$, the preferred hypothesis under Definition 4.2.1(iv) would be $\mathcal{H} = A(a)$.

Semantic minimality is usually only considered in the absence of disjunction in the hypotheses. For example, several works considering semantic minimality in DLs have restricted the solutions to $\mathcal{ALE}$, the conjunctive variant of $\mathcal{ALC}$ [KES11, HB12, HBK14]. As a result of this restriction on the space of solutions, the standard semantic minimality constraint in condition (iv) is sufficient to capture the notion of computing least assumptive hypotheses. However in more expressive DL languages which allow disjunction, such as full $\mathcal{ALC}$ and its extensions, other forms of redundancy need to be taken into account. This adds an additional challenge to the problem of computing "semantically minimal" hypotheses, as illustrated by Example 4.3.2.

Example 4.3.2. *Consider the following instance of the abduction problem:*

$$
\begin{aligned}
\mathcal{O} = \{ & \exists hD.BD \sqsubseteq \exists hS.Headache, \\
& TiredScientist \sqsubseteq \exists hS.Headache, \\
& \exists cO.BDV1 \sqsubseteq \exists hD.BD, \\
& TiredAccountant \sqsubseteq \exists hS.Headache, \\
& \neg TiredAccountant(\mathrm{p1})\} \\
\psi = \{ & \exists hS.Headache(p1)\}
\end{aligned}
$$

where Brain Drain (BD) is a disease, "BDV1" is a virus, p1 is a patient and the acronyms hD, hS *and* cO *stand for "hasDisease", "hasSymptom" and "carrierOf" respectively. Suppose the set of abducibles $\mathcal{S}_A$ includes all symbols in $\mathcal{O}$ except Headache* *and consider the hypotheses*[2]*:*

$$\mathcal{H} = \exists \mathit{hD.BD(p1)} \vee \mathit{TiredScientist(p1)} \qquad \textit{and}$$

$$\mathcal{H}' = \exists \mathit{hD.BD(p1)} \vee \mathit{TiredScientist(p1)} \vee \exists \mathit{cO.BDV1(p1)} \vee \mathit{TiredAccountant(p1)}$$

With respect to Definition 4.2.1, $\mathcal{H}'$ does not satisfy condition (iii) as it contains the redundant disjuncts $\exists cO.BDV1(p1)$ and TiredAccountant$(p1)$. Thus, the preferred solution is $\mathcal{H}$.

Both hypotheses in Example 4.3.2 satisfy Definition 4.2.1 conditions (i) and (ii). In the absence of condition (iii), both $\mathcal{H}$ and $\mathcal{H}'$ would also satisfy the semantic minimality requirement in condition (iv): there are no other hypotheses that are expressible without *Headache* which are strictly weaker than $\mathcal{H}$ or $\mathcal{H}'$. It is also the case that $\mathcal{O}, \mathcal{H} \equiv \mathcal{O}, \mathcal{H}'$. This means that, according to the standard semantic minimality definition, both hypotheses are "as minimal" as one another. However, there are two redundant disjuncts in $\mathcal{H}'$: *TiredAccountant(p1)* and $\exists$*cO.(BDV1)(p1)*. The first is inconsistent with the ontology $\mathcal{O}$. The second is not independent: it is simply stronger than the disjunct $\exists$*hD.BD(p1)* in $\mathcal{H}'$. Thus, condition (iii) excludes these disjuncts, resulting in the preferred hypothesis $\mathcal{H}$.

Attempting to produce semantically minimal hypotheses without any notion of disjunctive redundancy is problematic. Intuitively, a disjunctive hypothesis can be viewed as a set of alternative explanations for the given observation. In Example 4.3.2, adding either ∃hD.BD(p1) or TiredScientist(p1) from the overall hypothesis $\mathcal{H}$ is sufficient to explain the observation that the individual $p1$ has the symptom $Headache$. This highlights an advantage of computing a hypothesis as a disjunction of conjoined statements, i.e., a space of explanations: it is possible to select the most suitable explanation from the presented set of possibilities, rather than being presented with a hypothesis that represents only one explanation which may not be the most suitable explanation.

[2] In this example, both $\mathcal{H}$ and $\mathcal{H}'$ can be represented equivalently in $\mathcal{ALC}$ as the disjunction occurs over a single individual, e.g. $\mathcal{H} = (\exists \text{hD.BD} \sqcup \text{TiredScientist})(p1)$. However, to maintain consistency with the general case of the proposed abduction problem, hypotheses will be represented using disjunctive assertions as in the example.

Without the additional inter-disjunct redundancy constraint in condition (iii), hypotheses consisting of any number of redundant explanations, i.e., disjuncts, can be considered as a "semantically minimal" solution to the abduction problem. As a result, the benefit of presenting multiple alternative explanations is lost. If a disjunct is inconsistent with the background knowledge, then it is not an informative explanation for the observation since everything is trivially entailed when it is added to the background knowledge. Alternatively, if a disjunct is stronger than another disjunct or disjuncts in the rest of the hypothesis, then the intuitive notion of semantic minimality is violated: the explanation represented by this disjunct would make more assumptions than strictly necessary to explain the observation. In the context of large DL ontologies, the severity of this problem is more pronounced. It is difficult for a user to determine which explanations are valid and how each explanation is related in terms of strength, particularly when the background knowledge base is large and the hypothesis computed could consist of a large number of disjuncts.

The inter-disjunct redundancy condition excludes only those candidate hypotheses that contain disjunctive redundancies, i.e., those containing explanations that are *redundant* with respect to the above notions, as possible solutions to the abduction problem. Conceptually, these can be split into three general categories: inconsistent disjuncts, disjuncts that are strictly stronger than another subset of disjuncts in the hypothesis $\mathcal{H}$ or disjuncts that are equivalent to another subset of disjuncts in the hypothesis.

To illustrate the third case, consider the following example:

Example 4.3.3. *Consider the following abduction problem:*

$$\begin{aligned} \mathcal{O} = \{&C \sqcup D \equiv A, \\ &A \sqsubseteq B, \\ &C \sqsubseteq B, \\ &D \sqsubseteq B\} \\ \psi =&B(a) \end{aligned}$$

$$\mathcal{S}_A = \{A, C, D\}$$

and consider the following candidate hypothesis:

$$\mathcal{H}_1 = A(a) \vee C(a) \vee D(a)$$

This hypothesis is not an acceptable solution to the abduction problem in Definition 4.2.1, since $\mathcal{O}, C(a) \models A(a)$, $\mathcal{O}, D(a) \models A(a)$ and $\mathcal{O}, A(a) \models C(a) \vee D(a)$. Thus, it is necessary to eliminate either $C(a) \vee D(a)$ or $A(a)$ to obtain a satisfactory hypothesis, i.e., a space of independent explanations for ψ. If $A(a)$ is eliminated first, then the hypothesis obtained would be $\mathcal{H}_2 = C(a) \vee D(a)$ and since neither $\mathcal{O}, C(a) \models D(a)$ nor $\mathcal{O}, D(a) \models C(a)$ hold, $\mathcal{H}_2$ is a satisfactory hypothesis. If either $C(a)$ or $D(a)$ are eliminated first, then $A(a)$ will no longer be redundant since neither $\mathcal{O}, A(a) \models C(a)$ nor $\mathcal{O}, A(a) \models D(a)$ hold. Thus, both $C(a)$ and $D(a)$ will be eliminated in either order, leaving the hypothesis $\mathcal{H}_3 = A(a)$ which also satisfies the abductive constraints.

Without the above redundancies, the resulting hypothesis must also be semantically minimal under condition (iv). In the case where the hypothesis is a disjunction, this also means that each disjunct must be a minimal explanation as shown by the following Lemma.

Lemma 4.3.1. *For a hypothesis $\mathcal{H} = \alpha_1 \vee ... \vee \alpha_n$ that satisfies conditions (i)–(iii) of Definition 4.2.1 to also satisfy condition (iv), it must be the case that each disjunct α_i in $\mathcal{H}$ is also minimal in the sense that there is no other statement γ_i, where $sig(\gamma_i) \subseteq \mathcal{S}_A$, such that $\mathcal{O}, \gamma_i \models \psi$ and $\mathcal{O}, \alpha_i \models \gamma_i$ but $\mathcal{O}, \gamma_i \not\models \alpha_i$.*

Proof: Consider an arbitrary disjunct α_i in $\mathcal{H}$. The case of inconsistent disjuncts $\mathcal{O}, \alpha_i \models \bot$ is already excluded by condition (iii). Similarly, the case where α_i entails some other subset of the disjuncts in $\mathcal{H}$ under $\mathcal{O}$ is excluded. This leaves the case for which there is an alternative statement γ_i which is also an explanation for ψ under $\mathcal{O}$ such that $\mathcal{O}, \alpha_i \models \gamma_i$ but $\mathcal{O}, \gamma_i \not\models \alpha_i$, i.e., a weaker alternative to one of the existing disjuncts in $\mathcal{H}$. Consider a hypothesis $\mathcal{H}'$ constructed by replacing α_i in $\mathcal{H}$ with γ_i. The hypothesis $\mathcal{H}'$ satisfies conditions (i) – (iii), since $\mathcal{O}, \alpha_i \models \gamma_i$ and given that $\mathcal{H}$ is free of inter-disjunct redundancy, it also holds that $\mathcal{O}, \gamma_i \not\models \alpha_1 \vee ... \vee \alpha_{i-1} \vee \alpha_{i+1} \vee ... \vee \alpha_n$. Then the following holds: $\mathcal{O}, \mathcal{H} \models \mathcal{H}'$ but $\mathcal{O}, \mathcal{H}' \not\models \mathcal{H}$, and thus $\mathcal{H}$ does not satisfy condition (iv).

Example 4.3.3 also demonstrates that solutions to the problem in Definition 4.2.1 are not necessarily unique: both $\mathcal{H}_2$ and $\mathcal{H}_3$ are solutions to the problem satisfying both semantic minimality and the inter-disjunct redundancy condition.

In some cases, it is also possible that no suitable solution exists for the specified abduction problem, as in Example 4.3.4.

Example 4.3.4. *Consider the following abduction problem:*

$$\begin{aligned}\mathcal{O} = \{&A \sqsubseteq C,\\ &B \sqsubseteq C,\\ &E(a)\}\\ \psi =&C(a)\\ \mathcal{S}_A = &\{E\}\end{aligned}$$

For this problem, there is no suitable hypothesis: the signature of abducibles includes only E and it is clear that $E(a)$ would not be an explanation for $\psi = C(a)$, i.e., $\mathcal{O}, E(a) \not\models C(a)$. Therefore, there is no hypothesis satisfying Definition 4.2.1 in the signature $\mathcal{S}_A$.

It is also worth noting that Definition 4.2.1(i), consistency, is a direct consequence of condition (iii). Since each disjunct α_i in $\mathcal{H}$ must not be inconsistent with $\mathcal{O}$, the overall disjunction $\alpha_1 \vee ... \vee \alpha_n$ will also be consistent with $\mathcal{O}$. However, as consistency is a key condition in most abduction contexts it is explicitly included in the definition for both clarity and to emphasise its importance. This also provides the option to seek a consistent hypothesis that only partially fulfils the stronger inter-disjunct redundancy requirement.

Definition 4.2.1 does not remove all forms of redundancy from the hypothesis. The focus is on removing redundant explanations via condition (iii), and hypotheses that make assumptions that are too strong via conditions (iii) and (iv). Therefore, the final hypothesis is the least assumptive space of explanations. However, there exist other forms of redundancy that are not covered by Definition 4.2.1. For example, another abductive criteria in the literature focuses on the length of the produced explanations: if the shortest possible explanations are preferred, then the abduction problem seeks hypotheses that conform to the notion of syntactic minimality. A definition for this notion is provided in Definition 3.3.5 of Chapter 3. Syntactic minimality is not a specified constraint in Definition 4.2.1, and the following example illustrates how syntactic redundancy is treated under this definition:

Example 4.3.5. *Consider the following abduction problem:*

$$\begin{aligned}
\mathcal{O} =& \{\exists r.B \sqsubseteq A, \\
& C \sqsubseteq A, \\
& D \sqsubseteq C, \\
& E \sqsubseteq F, \\
& E(ind1)\} \\
\psi =& A(ind1) \\
\mathcal{S}_A =& \{r, B, C, D, E, F\}
\end{aligned}$$

and the following candidate hypotheses:

$$\begin{aligned}
\mathcal{H}_1 &= \exists r.B(ind1) \vee C(ind1) \\
\mathcal{H}_2 &= \exists r.B(ind1) \vee (C \sqcap F)(ind1) \\
\mathcal{H}_3 &= \exists r.B(ind1) \vee (C \sqcap D)(ind1)
\end{aligned}$$

Under Definition 4.2.1, $\mathcal{H}_1$, $\mathcal{H}_2$ and $\mathcal{H}_3$ all satisfy condition (i): they are all consistent with $\mathcal{O}$. They also satisfy condition (ii): $\mathcal{O}, \mathcal{H} \models \psi$ for all three candidate hypotheses.

For condition (iii): $\mathcal{O}, \{\exists r.B(ind1)\} \not\models C(ind1)$ *and* $\mathcal{O}, \{C(ind1)\} \not\models \exists r.B(ind1)$, *thus* $\mathcal{H}_1$ *is free of inter-disjunct redundancy. The same is true of both* $\mathcal{H}_2$ *and* $\mathcal{H}_3$, *despite the additional conjuncts in* $C \sqcap F$ *and* $C \sqcap D$ *respectively. Finally,* $\mathcal{H}_1$ *and* $\mathcal{H}_2$ *satisfy the semantic minimality requirement in condition (iv): while* $\mathcal{O}, \mathcal{H}_2 \models \mathcal{H}_1$, *it is also the case that* $\mathcal{O}, \mathcal{H}_1 \models \mathcal{H}_2$ *as* $F(ind1)$ *follows from* $\mathcal{O}$. *However,* $\mathcal{H}_3$ *does not satisfy condition (iv), as* $\mathcal{O}, \{(C \sqcap D)(ind1)\} \models C(ind1)$, *but* $\mathcal{O}, \{C(ind1)\} \not\models (C \sqcap D)(ind1)$ *due to the axiom* $D \sqsubseteq C$ *in* $\mathcal{O}$. *Therefore, it follows that* $\mathcal{O}, \mathcal{H}_3 \models \mathcal{H}_1$ *but* $\mathcal{O}, \mathcal{H}_1 \not\models \mathcal{H}_3$. *The same is true for* $\mathcal{H}_2$, *where* $\mathcal{O}, \mathcal{H}_3 \models \mathcal{H}_2$ *but* $\mathcal{O}, \mathcal{H}_2 \not\models \mathcal{H}_3$. *As a result,* $\mathcal{H}_3$ *is not semantically minimal as it is stronger than both* $\mathcal{H}_1$ *and* $\mathcal{H}_2$.

In the above example, $\mathcal{H}_2$ demonstrates one form of syntactic redundancy that is not accounted for by Definition 4.2.1: neither the explanation $(C \sqcap F)(a)$ nor $C(a)$ are preferred under the definition, since neither is stronger under the background ontology. However, if the presence of an additional conjunct makes the hypothesis stronger as in $\mathcal{H}_3$, this would be excluded via the semantic minimality constraint in condition (iv).

Similarly, no preference is made on the number of explanations presented as disjunctions in the hypothesis, provided that the overall space of explanations is still semantically minimal. This is illustrated by Example 4.3.3: without further syntactic constraints, under Definition 4.2.1 both $\mathcal{H}_2 = C(a) \vee D(a)$ and $\mathcal{H}_3 = A(a)$ are satisfactory hypotheses since neither is strictly stronger than the other under the background ontology.

One possibility for simplifying cases such as Example 4.3.3 would be to introduce specific orderings over the disjuncts in $\mathcal{H}$. Definition 4.2.1(iii) can then be applied based upon the given ordering. In Example 4.3.3, consider an ordering $\succ$ on the preferred symbols in explanations such that $A \succ C \succ D$. By performing redundancy checks for symbols lower down the ordering first, in the presence of an equivalence, less preferred symbols will be eliminated rather than those ranked higher in the ordering. This would mean that the disjuncts $C(a) \vee D(a)$ would be eliminated first in Example 4.3.3. Alternatively, the ordering could be based on the number of symbols present in each disjunct, preferring the shortest in the case of two equivalent explanations.

There are a variety of approaches to defining and realising preference handling [CMP96, PPU03, DSTW04]. However, this work focuses on computing the weakest possible space of independent explanations, rather than ensuring each individual explanation takes the simplest form with respect to criteria such as syntactic redundancy.

As noted, the problem of computing spaces of independent explanations as presented in Definition 4.2.1 has not been addressed in the DL literature. As a result, it is not feasible to provide a direct, empirical comparison between the above methods and the abduction approaches presented in this thesis. The primary reason for this is the fact that several of the conditions, namely inter-disjunct redundancy and semantic minimality in this context, differ significantly from existing works on abduction. Therefore, the problem is fundamentally different to the forms of abduction discussed in Section 3.3 of Chapter 3. Additionally, most works in DLs do not permit the specification of abducible symbols to constrain the hypotheses produced, which again changes the problem fundamentally. Consequently, the experimental results presented in subsequent chapters focus on the performance of the abduction approaches presented in this thesis in a variety of different scenarios.

Chapter 5

An ABox Abduction Approach for $\mathcal{ALC}$

This chapter presents an approach for computing hypotheses satisfying the constraints identified in the previous chapter. To develop such an approach, it is first necessary to identify a suitable forgetting approach that can be used to compute the weakest sufficient condition of the negated observation under the background ontology, yielding a set of entailments that can be negated under contraposition to obtain a semantically minimal hypothesis $\mathcal{H}$, where filtering can then be applied to ensure that the disjunctive redundancy requirement is met. The use of contraposition for abduction presents a challenge for an $\mathcal{ALC}$ observations that take the form of sets of ABox axioms over different individuals. While $\mathcal{ALC}$ concepts are closed under negation, the negation of sets of $\mathcal{ALC}$ ABox axioms presents a challenge. For example, given $\psi = C(a) \sqcap D(b)$ where C, D are concepts and a, b are two different individuals, the negation $\neg\psi$ is not representable in pure $\mathcal{ALC}$. To obtain a hypothesis such as $\mathcal{H} = C(a) \sqcap D(b)$ also requires the computation of $\neg\mathcal{H}$, which likewise requires the use of disjunctive assertions. Therefore, the forgetting approach used must be capable of handling and producing these disjunctive assertions.

The forgetting system LETHE, described in Section 3.5.1, is suited to this problem, and is therefore used to compute the entailments required for contrapositive reasoning.

Since LETHE also provides support for representing cyclic results using fixpoint operators, the abduction problem identified in the previous chapter can be extended to include explanations requiring least fixpoints. Several filtering methods are proposed to eliminate disjunctive redundancies, emphasising efficiency in practice and with respect to potential applications.

The proposed abduction approach can perform ABox abduction for input ontologies and observations expressed in the DL language $\mathcal{ALC}$, producing hypotheses expressed in $\mathcal{ALC}\mu(\vee)$, i.e., $\mathcal{ALC}$ with least fixpoints and disjunctive ABoxes. The capabilities and shortcomings of the resulting approach to the abduction problem are identified and discussed, particularly in terms of extensions that must be made to solve more expressive problems such as observations and hypotheses including role assertions. An experimental evaluation of the resulting algorithm is performed over a corpus of real world ontologies to provide an indication of the practicality of the approach in applications.

5.1 Problem Definition

The problem tackled in this chapter is defined as follows.

Definition 5.1.1. *Let $\langle \mathcal{O}, \psi, \mathcal{S}_A \rangle$ be an ABox abduction problem that extends Definition 4.2.1 in the following way: each disjunct α in the hypothesis $\mathcal{H}$ (and α' in $\mathcal{H}'$) is expressed in $\mathcal{ALC}\mu$ and thus the abduction solution is expressed in $\mathcal{ALC}\mu(\vee)$. To be a solution, $\mathcal{H}$ must satisfy conditions (i)–(iv) of Definition 4.2.1.*

This problem takes the same form as the abduction problem presented in Definition 4.2.1, following the same motivations. As before, the language for $\mathcal{O}, \psi$ is assumed to be $\mathcal{ALC}$. For $\mathcal{H}$, the form taken is still a disjunction of conjunctions of ABox axioms, i.e., a disjunction of ABoxes. However, Definition 5.1.1 extends the problem in that each disjunct in $\mathcal{H}$ can be expressed in an extended language: $\mathcal{ALC}$ with least fixpoints ($\mathcal{ALC}\mu$). This is due to the fact that the forgetting approach LETHE, which can handle disjunctive assertions as required, supports the use of fixpoint operators to represent cyclic entailments of the given input. This enables the representation of cyclic explanations if required. As a result, $\mathcal{H}$ is expressed in the language $\mathcal{ALC}\mu(\vee)$ where μ indicates least fixpoints and $\vee$ indicates the use of disjunctive ABoxes. The focus remains specifically on ABox abduction, where both ψ and $\mathcal{H}$ are ground.

As discussed in the previous chapter, the main restriction applied to the abduction problem is that neither ψ nor $\mathcal{H}$ can contain role assertions. Effectively, this means that the hypothesis is a disjunction of conjunctions of concept assertions, i.e., a disjunction of $\mathcal{ALC}\mu$ ABoxes without role assertions. The reasoning behind the language extension required to express the computed hypothesis $\mathcal{H}$ and the restrictions applied are discussed alongside the proposed approach.

5.2 ABox Abduction Algorithm

The proposed forgetting-based abduction algorithm takes as input an ontology $\mathcal{O}$ expressed in the DL $\mathcal{ALC}$, an observation ψ as a set of $\mathcal{ALC}$ axioms and a forgetting signature $\mathcal{F}$. The set of abducibles is the complement of the forgetting signature, i.e., $\mathcal{F} = sig(\mathcal{O}, \psi) \setminus \mathcal{S}_A$ where $\mathcal{S}_A$ is a set of concept and role symbols, and so the forgetting signature specifies the set of non-abducibles.[1] The output is then a hypothesis expressed as a disjunction of axioms, which uses only symbols in $\mathcal{S}_A$ and any individuals occurring in $\mathcal{O}, \psi$. Since the problem to be solved is ABox abduction, both ψ and $\mathcal{H}$ consist only of ABox axioms (excluding role assertions), i.e., ground statements.

The algorithm reduces the task of computing abductive hypotheses for the observation ψ to the task of forgetting, using the following steps:

(1) Negate the observation ψ and add this to the background ontology $\mathcal{O}$.

(2) Compute the forgetting solution $\mathcal{V}$ of $(\mathcal{O}, \neg\psi)$ with respect to the forgetting signature $\mathcal{F}$.

(3) Extract the reduced forgetting solution $\mathcal{V}^*$, which is the set $\mathcal{V}^* \subseteq \mathcal{V}$ obtained by

[1] As stated in Chapter 4, it is assumed that the set of abducibles contains all individuals in $(\mathcal{O}, \psi)$.

omitting all axioms in $\mathcal{V}$ that are redundant under the dual of Definition 5.1.1 condition (iii).

(4) Obtain the hypothesis $\mathcal{H}$ by negating the set $\mathcal{V}^*$.

These steps are illustrated in Figure 5.1. The observation ψ takes the form of a set of ABox axioms:

$$\psi = \{C_1(a_1), ..., C_k(a_k)\}$$

where the C_i are $\mathcal{ALC}$ concepts and the a_i are individuals. The negation takes the form

$$\neg\psi = \neg C_1(a_1) \vee ... \vee \neg C_k(a_k)$$

which is a disjunction of ABox axioms. The negation of ψ is required for contrapositive reasoning. As discussed earlier, forgetting can be used to compute the set of entailments, $\neg\mathcal{H}$, required for contraposition:

$$\mathcal{O}, \mathcal{H} \models \psi \qquad \text{iff} \qquad \mathcal{O}, \neg\psi \models \neg\mathcal{H}$$

where in this case the set of entailments $\neg\mathcal{H}$ is equal to the forgetting solution $\mathcal{V}$ of $(\mathcal{O}, \neg\psi)$, which takes the following form:

$$\mathcal{V} = \{\beta_1, ..., \beta_m\}$$

where each β_i is an $\mathcal{ALC}\nu$ axiom. This is obtained by forgetting the concept names in $\mathcal{F}$ using an appropriate forgetting calculus. As discussed previously, forgetting is the dual task of uniform interpolation, and therefore the forgetting solution $\mathcal{V}$ satisfies the conditions specified in Definition 3.4.2 for the signature of abducibles $\mathcal{S}_A$.

If forgetting was used in isolation, the hypothesis obtained would be the negation of the forgetting solution $\mathcal{V}$. However, this is only guaranteed to satisfy the notions of

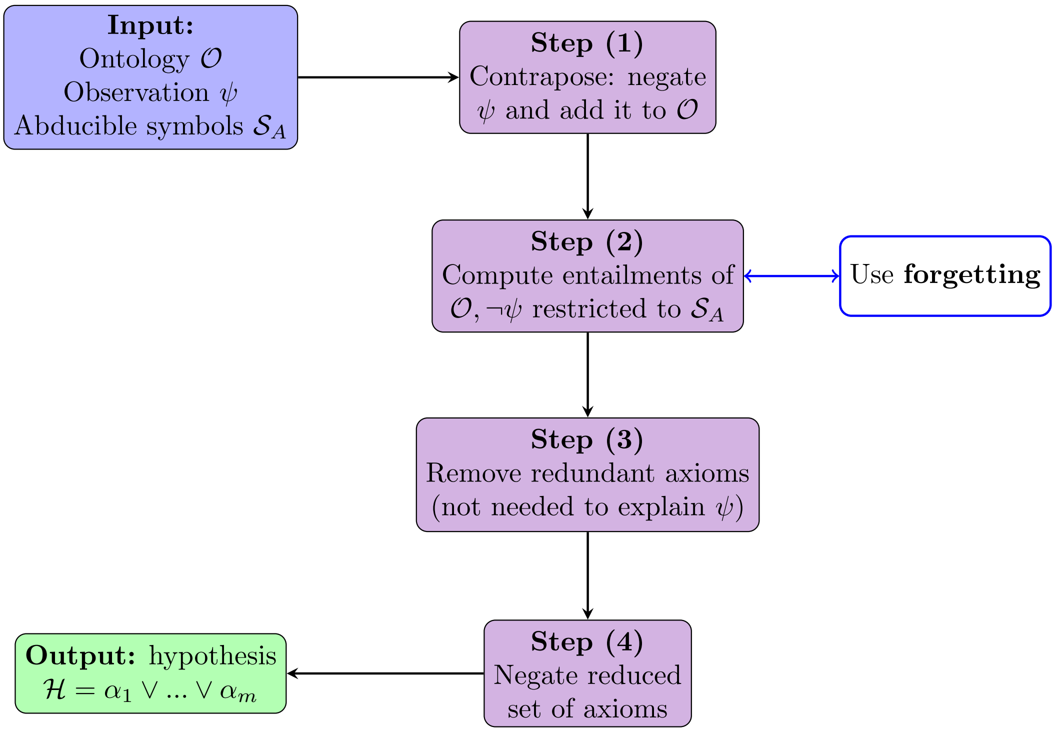

Figure 5.1: Steps in the abduction algorithm for computing hypotheses as spaces of independent explanations.

entailment and semantic minimality, without considering redundancy, captured in conditions (ii) and (iv) of Definition 5.1.1. This follows from the fact that $\mathcal{V}$ is a strongest necessary condition of $\neg\psi$ under $\mathcal{O}$ in $\mathcal{S}_A$ as in Theorem 3.6.1 where $\mathcal{O}, \neg\psi \models \mathcal{V}$, and its negation would be the weakest sufficient condition of ψ [Lin01, DLS01] where $\mathcal{O}, \neg\mathcal{V} \models \psi$ under contrapositive reasoning. Thus the hypothesis would entail the observation under $\mathcal{O}$, satisfying condition (ii), and would be semantically minimal in $\mathcal{S}_A$, satisfying the notion in condition (iv) if redundancy was not considered. However, it would not necessarily satisfy Definition 5.1.1 condition (iii). This is discussed further in Section 5.4.2 and is illustrated by Example 5.4.1. Additionally, in the event that there is no suitable hypothesis for ψ in the signature $\mathcal{S}_A$, the result obtained by negating $\mathcal{V}$ would simply be inconsistent with $\mathcal{O}$. This is due to the fact that all of the axioms in $\mathcal{V}$ would follow directly from $\mathcal{O}$. For example:

Example 5.2.1. *Consider an ontology $\mathcal{O} = \{A \sqsubseteq C, B \sqsubseteq C, E(a))\}$ and observation*

$\psi = C(a)$. Let the signature of abducibles be $\mathcal{S}_A = \{E\}$. Then the forgetting solution obtained by eliminating A, B and C is $\mathcal{V} = \{E(a)\}$. The negation of $\mathcal{V}$ is $\neg E(a)$, which is inconsistent with $\mathcal{O}$ and is therefore not a suitable hypothesis. The result should instead be $\mathcal{H} = \emptyset$.

The purpose of Step (3) is to omit unnecessary information in $\mathcal{V}$, to ensure that the hypothesis fully satisfies Definition 5.1.1. The unnecessary information corresponds to axioms in $\mathcal{V}$ that follow from the background knowledge $\mathcal{O}$ together with other axioms in $\mathcal{V}$ itself. This check is the dual of the inter-disjunct redundancy check in Definition 5.1.1(iii), i.e., an axiom β_i in $\mathcal{V}$ is redundant if the following holds:

$$\mathcal{O}, \beta_1, ..., \beta_{i-1}, \beta_{i+1}, ..., \beta_m \models \beta_i$$

This check therefore eliminates inter-disjunct redundancies such as those in Example 4.3.2. It is assumed that if an axiom β_i is redundant, then it is removed from $\mathcal{V}$ immediately and the following checks are performed with respect to the remaining axioms.[2] This is discussed further in the next section.

The result is a *reduced forgetting solution* $\mathcal{V}^*$ which takes the form:

$$\mathcal{V}^* = \{\beta_1, ..., \beta_n\}$$

where each β_i is expressed in $\mathcal{ALC}\nu(\vee)$.[3] It is worth noting that each of the axioms $\beta_i \in \mathcal{V}^*$ will be an ABox axiom: either a disjunction over a single individual or a disjunctive assertion over multiple individuals. This is due to the fact that all axioms in the forgetting solution $\mathcal{V}$ that are derived solely from the background knowledge in $\mathcal{O}$ will not satisfy the dual of the inter-disjunct requirement in Definition 5.1.1(iii), as they are entailed by $\mathcal{O}$ alone, and will thus not be present in the reduced forgetting solution

[2]This ensures that if the redundancy occurs due to equivalence, then one of the equivalent axioms is retained in the reduced forgetting solution.

[3]Note: the forgetting solution $\mathcal{V}$ can contain greatest fixpoints ν, leading to the use of $\mathcal{ALC}\nu(\vee)$. Meanwhile, the hypothesis can contain least fixpoints μ and is expressed in $\mathcal{ALC}\mu(\vee)$, since negation is pushed inwards and greatest / least fixpoints are dual notions.

$\mathcal{V}^*$. Axioms that do satisfy the requirement, and therefore appear in $\mathcal{V}^*$, must be derived via inferences with $\neg\psi$. Since $\neg\psi$ is a disjunction of ABox axioms, any axioms dervied in this way must also be ABox axioms: it is not possible to derive a TBox axiom (universally quantified) via an inference under the $Int_{\mathcal{ALC}}$ rules, shown in Figure 3.2, where one of the premises is an ABox axiom (ground). Note however that not all axioms derived via inferences with $\neg\psi$ will be present in $\mathcal{V}^*$. These characteristics of $\mathcal{V}^*$ are discussed further in Section 5.4.2.

These redundancies can be eliminated by performing entailment checking for the dual of the inter-disjunct redundancy requirement for each axiom in $\mathcal{V}$. For real world ontologies this is not practical due to the complexity of entailment checking in $\mathcal{ALC}$ and the fact that many of these ontologies are large. This necessitates the use of more efficient approaches to performing Step (3), which is discussed in the next section.

In Step (4) the reduced forgetting solution $\mathcal{V}^*$ is negated. The result of this is a hypothesis of the form:

$$\mathcal{H} = \alpha_1 \vee ... \vee \alpha_n$$

where each disjunct α_i is a conjunction of $\mathcal{ALC}\mu$ concept assertions of the form $\alpha_i = D_1(a_1) \sqcap ... \sqcap D_o(a_o)$ where for $1 \leq j \leq o$ each D_j is an $\mathcal{ALC}\mu$ concept and each a_j is an individual. Thus, $\mathcal{H}$ can be viewed as a disjunction of ABoxes where $\mathcal{H}$ is assumed to be in disjunctive normal form (DNF).

5.3 Forgetting Step

As discussed in Chapter 3, the connection between forgetting and abduction lies in contrapositive reasoning, where forgetting is used to compute the strongest set of entailments of $\mathcal{O}, \neg\psi$ within the restricted signature $\mathcal{S}_A$. Therefore, to realise a forgetting-based approach to abductive reasoning it is first necessary to identify and apply an appropriate forgetting calculus.

For the work in this chapter, the resolution-based calculus by Koopmann and Schmidt is investigated [KS13, KS15b, KS15a]. The calculus itself, referred to as $Int_{\mathcal{ALC}}$, is presented in Figure 3.2 and is discussed in Section 3.5 of Chapter 3.

The motivations for utilising $Int_{\mathcal{ALC}}$ specifically include:

1. Forgetting can be performed over $\mathcal{ALC}$ ontologies with ABoxes [KS15b].
2. Potentially infinite forgetting solutions can be represented finitely using fixpoint operators.
3. Disjunctive assertions of the form $C_1(a_1) \vee ... \vee C_n(a_n)$ are supported.
4. The size of the forgetting solution is constrained to at most a double exponential size with respect to the input ontology, and the method is guaranteed to terminate.

The ability to perform forgetting in $\mathcal{ALC}$ is essential given the setting of the abduction problem specified in Definition 5.1.1. Similarly, the ability to perform forgetting in the presence of an ABox is also essential for producing a hypothesis as a disjunction of ABox axioms to explain an ABox observation.

The option to represent infinite results is important for abduction, since it is possible that the hypothesis obtained may involve cycles as illustrated by the following example:

Example 5.3.1. *Consider the following ontology $\mathcal{O}$:*

$$Mammal \sqsubseteq \exists hasParent.Mammal$$

and an observation $\psi = \{\neg Mammal(a^*)\}$ with a forgetting signature $\mathcal{F} = \{Mammal\}$. In Step (1), the observation is negated to obtain $\neg\psi = Mammal(a^*)$, and is added to $\mathcal{O}$. In Step (2), $Int_{\mathcal{ALC}}$ is applied as follows:

$$1. \neg Mammal(x) \vee \exists hasParent.D_1(x)$$

$$2. \neg D_1(x) \vee Mammal(x)$$

$$3.\mathit{Mammal}(a^*)$$

$$4.\neg D_1(x) \vee \exists \mathit{hasParent}.D_1(x) \qquad \mathit{Resolution}(1,2)$$

$$5.\exists \mathit{hasParent}.D_1(a^*) \qquad \mathit{Resolution}(1,3)$$

At this point, all inferences have been made. Now definer symbols are eliminated and clauses containing symbols in $\mathcal{F} = \{\mathit{Mammal}\}$ are removed. The elimination of D_1 results in the introduction of a greatest fixpoint operator, representing a potentially infinite chain under the hasParent *relation in axioms 4 and 5. The resulting uniform interpolant is*

$$\mathcal{V} = \{\exists \mathit{hasParent}.\nu X.(\exists \mathit{hasParent}.X)(a^*)\}$$

where $\nu X.(\exists \mathit{hasParent}.X)$ is the greatest fixpoint. In Step (3), since there are no redundant axioms with respect to Definition 5.1.1 condition (iii), the reduced uniform interpolant $\mathcal{V}^*$ is simply equal to $\mathcal{V}$. In Step (4), $\mathcal{V}^*$ is negated to obtain the hypothesis:

$$\mathcal{H} = \forall \mathit{hasParent}.\mu X.(\forall \mathit{hasParent}.X)(a^*)$$

where μX is a least fixpoint operator.

The introduction of fixpoint operators via $Int_{\mathcal{ALC}}$ can be seen by seen by comparing the elimination of non-cylic and cyclic definers under Ackermann's lemma [KS13] shown in Figure 3.3, where the latter case results in a greatest fixpoint operator being used. Since the forgetting solution is negated as part of the abduction procedure, this means that least fixpoints, and only least fixpoints, could appear in the abduction hypothesis.

The introduction of a greatest fixpoint operator is due to the presence of axiom 4 in the above example. Effectively, the meaning of the least fixpoint in $\mathcal{H}$ is that, if a^* is not a Mammal as in ψ, then it must "not have a parent" or "must have a parent who does not have a parent..." and so on. In this limited ontology, this is the semantically minimal hypothesis in the signature $\mathcal{S}_A$, i.e, not involving the concept Mammal.

Thus in theory the result of forgetting (and abduction) can involve an infinite chain of axioms, which can be represented finitely via fixpoint operators. In practice, fixpoints are rarely required: in previous work only 7.2% of uniform interpolants contained cycles [KS13]. In the abduction context of Definition 5.1.1, it is reasonable to expect that fixpoints would not be required frequently: the cycle would need to occur over symbols in ψ that are part of the forgetting signature $\mathcal{F}$, and the axiom expressed using a fixpoint would need to be derivable from $\neg\psi$ to ensure that it is kept in the reduced forgetting solution $\mathcal{V}^*$. This intuition is supported by the results in Section 5.6, where fixpoints did not occur in any case in practice.

The forgetting calculus $Int_{\mathcal{ALC}}$ can also handle disjunctive ABox assertions which are not representable in pure $\mathcal{ALC}$. This is needed since, for instance, if the observation is a set of assertions over different individuals then the negation of the observation will take the form of a disjunctive assertion. In addition, disjunctive assertions will be needed for some abduction cases involving multiple individuals in the hypothesis:

Example 5.3.2. *Consider the following abduction problem:*

$$\begin{aligned}
\mathcal{O} &= \{A \sqsubseteq C, \\
&\quad\;\; B \sqsubseteq D\} \\
\psi &= \{C(a), \\
&\quad\;\; D(b)\} \\
\mathcal{S}_A &= \{A, B\}
\end{aligned}$$

In Step (1), $\neg\psi = \neg C(a) \vee \neg D(b)$ is added to $\mathcal{O}$. In Step (2), $(\mathcal{O}, \neg\psi)$ is translated to the normal form required by $Int_{\mathcal{ALC}}$ and the following inferences are performed as part of the forgetting procedure:

$1)\neg A(x) \vee C(x)$

$2)\neg B(x) \vee D(x)$

$3)\neg C(a) \vee \neg D(b)$

$4)\neg A(a) \vee \neg D(b)$ *Resolution(1, 3)*

$5)\neg C(a) \vee \neg B(b)$ *Resolution(2, 3)*

$6)\neg A(a) \vee \neg B(b)$ *Resolution(1, 5) or Resolution(2, 4)*

All clauses containing symbols in $\mathcal{F}$ are then removed, leaving only clause 6. The resulting forgetting solution is then $\mathcal{V} = \{\neg A(a) \vee \neg B(b)\}$. Here, Step (3) simply returns $\mathcal{V}^* = \mathcal{V}$ as there are no redundant clauses under the dual of Definition 5.1.1 condition (iii). The hypothesis returned is then:

$$\mathcal{H} = A(a) \sqcap B(b)$$

As the above example shows, when computing a hypothesis involving a conjunction over multiple individuals, disjunctive assertions are still required due to contrapositive reasoning.

In terms of efficiency, termination of the method is important to ensure that a hypothesis is returned if one exists. The bound on the size of the forgetting result is also beneficial as this will have an impact on the performance of the resulting abduction method, particularly during the filtering process in Step (3). However, since this bound still leaves the possibility of a forgetting solution that is double exponential in size with respect to the input, efficient methods for this procedure are still essential as presented in the next section.

Note that the hypothesis specified in Definition 5.1.1 takes the form of a disjunction of conjunctions of concept assertions. Thus, the hypothesis is in disjunctive normal form (DNF). Since the transformation rules required by $Int_{\mathcal{ALC}}$ present the input ontology, in this case $\mathcal{O}, \neg\psi$ as a set of clauses in CNF, the hypothesis obtained by negating the forgetting result will be in DNF as required.

Two further properties of the $Int_{\mathcal{ALC}}$ method that are also important to the proposed abduction method are: (i) *Soundness*: any ontology $\mathcal{O}'$ returned by applying $Int_{\mathcal{ALC}}$ to an ontology $\mathcal{O}$ is a forgetting solution. (ii) *Interpolation Completeness*: if there exists a forgetting solution $\mathcal{O}'$ of ontology $\mathcal{O}$, then the result of $Int_{\mathcal{ALC}}$ is an ontology $\mathcal{V}$ such that $\mathcal{V} \equiv \mathcal{O}'$. Thus, for any $\mathcal{ALC}$ ontology $\mathcal{O}$ and any forgetting signature $\mathcal{F}$, $Int_{\mathcal{ALC}}$ always returns a finite forgetting solution. Soundness and completeness of the forgetting procedure naturally relate to the soundness and completeness of the resulting abduction procedure, since the filtering and negation steps operate under the assumption that the initial forgetting solution $\mathcal{V}$ is correct.

Aside from these main benefits, it is also important to consider the language used to express the hypothesis $\mathcal{H}$, and how this compares to the language used for the input. For a first investigation of forgetting-based approaches to abduction in DLs, choosing a method that only minimally extends the language used to express $\mathcal{O}$ and ψ has benefits: adding a hypothesis expressed in a more expressive language than the background ontology adds additional complexity to further reasoning problems. From an engineering perspective, it also fundamentally changes the modelling of information represented in subsequent iterations of a knowledge base. This may be problematic in applications for which efficient reasoning and careful modelling are important issues. Also, starting with a commonly used expressive DL such as $\mathcal{ALC}$ provides a good basis for characterising the capabilities of forgetting-based approaches to abduction in DLs.

5.4 Practical Realisation

5.4.1 Assumptions on Input and Output

Several assumptions are made regarding the input to the algorithm. The method $Int_{\mathcal{ALC}}$ does not cater for negated role assertions as can be seen in Figure 3.2. Thus, it is not possible to take an observation with a role assertion as input due to contraposition, and it is not possible to produce a role assertion as a hypothesis.

Another assumption is based on the notion of semantic minimality: if the forgetting signature $\mathcal{F}$ does not contain at least one symbol in the observation ψ, the semantically minimal hypothesis will simply be ψ itself, i.e., $\mathcal{H} = \psi$. This is reflected in the fact that no inferences would occur between $\mathcal{O}$ and $\neg\psi$ under $Int_{\mathcal{ALC}}$. To avoid this trivial hypothesis, $\mathcal{F}$ should contain at least one symbol in the signature of ψ.

Several assumptions are also made regarding the output $\mathcal{H}$. As discussed previously, two extensions of $\mathcal{ALC}$ are potentially required to express $\mathcal{H}$. In the event that $\mathcal{F}$ contains concepts that occur within a cycle in $\mathcal{O}$, the forgetting result obtained using $Int_{\mathcal{ALC}}$ may contain greatest fixpoints [KS13] to finitely represent infinite forgetting solutions. For our method, this means that $\mathcal{H}$ may contain least fixpoints due to the negation of greatest fixpoints under contraposition. In these cases, the output language would be $\mathcal{ALC}\mu$. Disjunctive assertions are required during both the forgetting step, to represent the negation of an observation over multiple individuals, and potentially to represent the final hypothesis $\mathcal{H}$ as a disjunction of assertions over multiple individuals.

Thus, with these two extensions, the most expressive language required to represent $\mathcal{H}$ is assumed to be $\mathcal{ALC}\mu(\vee)$, i.e., $\mathcal{ALC}$ extended with least fixpoint operators and disjunctive assertions.

5.4.2 Practical Elimination of Redundant Hypotheses

As discussed in Chapter 4, the inter-disjunct redundancy condition (iii) in Definition 5.1.1 is required to eliminate redundant disjuncts in the computed hypotheses. Since the hypotheses obtained via this approach are interpreted as sets of explanations, this ensures that the explanations obtained are consistent with the background ontology $\mathcal{O}$ and express a unique avenue of explanation in the sense that they are not equivalent to or stronger than other explanations in the hypothesis. The following example illustrates how such redundancies may be derived using the forgetting-based approach outlined in the previous section:

Example 5.4.1. *Consider the following abduction problem:*

$$\mathcal{O} = \{A \sqcap D \sqsubseteq \bot,$$

$$B \sqsubseteq C,$$

$$\begin{aligned} E &\sqsubseteq C, \\ B &\sqsubseteq E, \\ D&(a)\} \\ \psi =& C(a) \\ \mathcal{S}_A = & \{A, B, D, E\} \end{aligned}$$

In Step (1), $\neg\psi = \neg C(a)$ is added to $\mathcal{O}$. The result of forgetting C in Step (2) is then:

$$\begin{aligned} \mathcal{V} = \{ & A \sqcap D \sqsubseteq \bot, \\ & B \sqsubseteq E, \\ & D(a), \\ & \neg B(a), \\ & \neg E(a), \\ & \neg A(a)\} \end{aligned}$$

In the forgetting context, $\mathcal{V}$ is a correct solution to the forgetting problem, i.e., it is a strongest necessary condition of $\neg\psi$ under $\mathcal{O}$ in $\mathcal{S}_A$. If this was negated directly, the candidate hypothesis would be the following space of explanations:

$$\mathcal{H} = A(a^*) \sqcap D(a^*) \vee \neg B(a^*) \sqcap E(a^*) \vee \neg D(a) \vee B(a) \vee E(a) \vee A(a)$$

where a^* is a fresh individual, since the negation of a TBox axiom of the form $A \sqsubseteq B$ can be equivalently represented as $\neg A(a^*) \sqcap B(a^*)$.[4] However, in the abduction context the hypothesis obtained by negating $\mathcal{V}$ directly consists mostly of redundant explanations. Instead of negating $\mathcal{V}$ directly, a subset of relevant information should first be extracted. The two TBox axioms, as well as $D(a)$, follow directly from the background ontology $\mathcal{O}$ and should be discarded as redundant: they are contained

[4] Since TBox axioms are implicitly universally quantified, the universal quantifier becomes an existential quantifier once negation is pushed inwards. Application of Skolemization results in the form expressed above.

within $\mathcal{O}$ and therefore will not contribute to an explanation for ψ when negated. Similarly, $\neg A(a)$ follows from $A \sqcap D \sqsubseteq \bot$ together with $D(a)$ and should be discarded. Finally, it is the case that $\mathcal{O}, \neg E(a) \models \neg B(a)$. Therefore, $B(a)$ is redundant under the dual of the interdisjunct redundancy condition (iii) in Definition 5.1.1. The final hypothesis after redundancy elimination is therefore $\mathcal{H} = E(a)$.

Step (3) of Figure 5.1 is also necessary to ensure that an entirely redundant hypothesis is not returned. In the worst case, the hypothesis obtained could simply be inconsistent with the ontology $\mathcal{O}$, as in the following example:

Example 5.4.2. *Consider the abduction problem in Example 5.4.1. If the signature of abducibles was restricted to $\mathcal{S}_A = \{A, D\}$, the forgetting solution would instead be:*

$$\begin{aligned} \mathcal{V} = \{ & A \sqcap D \sqsubseteq \bot, \\ & D(a), \\ & \neg A(a)\} \end{aligned}$$

As before, these three axioms are redundant with respect to the abduction problem, since each follows directly from $\mathcal{O}$. Thus, if the negated forgetting solution was returned directly without additional filtering, the resulting hypothesis $\mathcal{H}$ would be inconsistent with $\mathcal{O}$.

The situation illustrated by Example 5.4.2 occurs in general when there is no hypothesis to explain the observation using the specified abducibles.

As demonstrated by the above examples: the hypothesis obtained by directly negating the forgetting solution is unlikely to satisfy Definition 5.1.1(iii). In practice, most of the disjuncts in such a hypothesis will be redundant: the forgetting solution is the strongest set of entailments of $\mathcal{O}, \neg\psi$, and will thus contain many entailments purely from $\mathcal{O}$. For large ontologies, most of the entailments will indeed follow purely from $\mathcal{O}$. Also, any redundancy contained within $\mathcal{O}$ will be reflected in this hypothesis, as demonstrated by Example 5.4.1. And it is not reasonable to assume that real world ontologies are free of redundancies. In fact, many of these "redundant" entailments may be intentional design choices when constructing an ontology in an ontology engineering setting: it may be necessary to emphasise subset relations directly, or to make extensive use of equivalences.

If an axiom β_i is redundant, it is removed from $\mathcal{V}$ immediately. For the following disjuncts, the check is performed against the remaining axioms in $\mathcal{V}$. This avoids discarding too many axioms: if multiple axioms express the same information, i.e. are equivalent under $\mathcal{O}$, one of them should be retained in the final hypothesis $\mathcal{H}$. This is illustrated by the following example:

Example 5.4.3. *Consider the following abduction problem:*

$$\begin{aligned} \mathcal{O} = \{&A \equiv B, \\ &A \sqsubseteq C, \\ &B \sqsubseteq C, \\ &\exists r.D \sqsubseteq C\} \\ \psi =& C(a) \\ \mathcal{S}_A = \{&r, A, B, D\} \end{aligned}$$

and a candidate hypothesis $\mathcal{H} = A(a) \vee B(a) \vee \exists r.D(a)$. The redundancy check proceeds as follows: since $\mathcal{O}, A(a) \models B(a) \vee \exists r.D(a)$, the disjunct $A(a)$ is considered redundant with respect to Definition 5.1.1(iii). If the check $\mathcal{O}, B(a) \models A(a) \vee \exists r.D(a)$ is then performed, then $B(a)$ will also be a redundant explanation and the hypothesis returned would be $\mathcal{H}_2 = \exists r.D(a)$. However, if $A(a)$ had been removed immediately, the check $\mathcal{O}, B(a) \models \exists r.D(a)$ would return false and the returned hypothesis would instead be $\mathcal{H}_3 = B(a) \vee \exists r.D(a)$. Since $\mathcal{O}, \mathcal{H}_2 \models \mathcal{H}_3$, but $\mathcal{O}, \mathcal{H}_3 \not\models \mathcal{H}_2$, $\mathcal{H}_2$ does not satisfy the semantic minimality condition. Thus, the correct solution under Definition 5.1.1 is $\mathcal{H}_3$.

In cases involving equivalent disjuncts, there are multiple possible solutions depending on the order in which the disjuncts are checked. Thus, it would be possible to apply an additional preference relation [CMP96] on the possible explanations to ensure which hypotheses would be returned in these cases. For example, it would be possible to check disjuncts in order of decreasing length, thereby preferring the shortest of multiple equivalent disjuncts. Here, since the aim is not to satisfy such additional preferences, the order in which the axioms are checked is random unless stated otherwise.

To perform the redundancy elimination, Step (3) requires checking the relation $\mathcal{O}, \mathcal{V} \setminus \beta_i \not\models \beta_i$ for every axiom β_i in $\mathcal{V}$. This could be performed using an external DL reasoner. However, in practice this check is likely to be intractable, particularly over large ontologies. In $\mathcal{ALC}$, entailment checking has exponential complexity [Sch94]. In the worst case, the forgetting solution $\mathcal{V}$ obtained via application of $Int_{\mathcal{ALC}}$ can be double exponential in size with respect to the input [KS15b]. As a result, there could be a double exponential number of entailment checks required to eliminate redundancies in Step (3). Thus, the worst-case complexity of Step (3) would be 3EXPTIME.

Regardless, Step (3) is essential; without it there will be a large number of redundant explanations, under Definition 5.1.1(iii), in the hypotheses obtained. Therefore, for a forgetting-based abduction approach to be viable, the question arises if a computationally feasible alternative can be devised. The required extension is illustrated in Figure 5.2.

To ensure that the computational cost of performing Step (3) is lowered, the number of entailment checks performed must be reduced. Our implementation of this step begins by tracing the dependency of axioms in $\mathcal{V}$ on the negated observation $\neg\psi$. To define the notion of dependency clearly, we will need the following notions. Each premise in an application of an inference rule in $Int_{\mathcal{ALC}}$ is referred to as a *parent* of the conclusion of the rule. The *ancestor* relation is defined as the reflexive, transitive closure of the parent relation.

Therefore, an axiom β is defined as *dependent* upon $\neg\psi$ if in the derivation using $Int_{\mathcal{ALC}}$ it has at least one ancestor axiom in $\neg\psi$. The set of axioms dependent on $\neg\psi$ is in general a superset of the reduced uniform interpolant $\mathcal{V}^*$ and is referred to as $\mathcal{V}^*_{app}$, i.e., an *approximation* of $\mathcal{V}^*$. The notion of dependency is used to devise

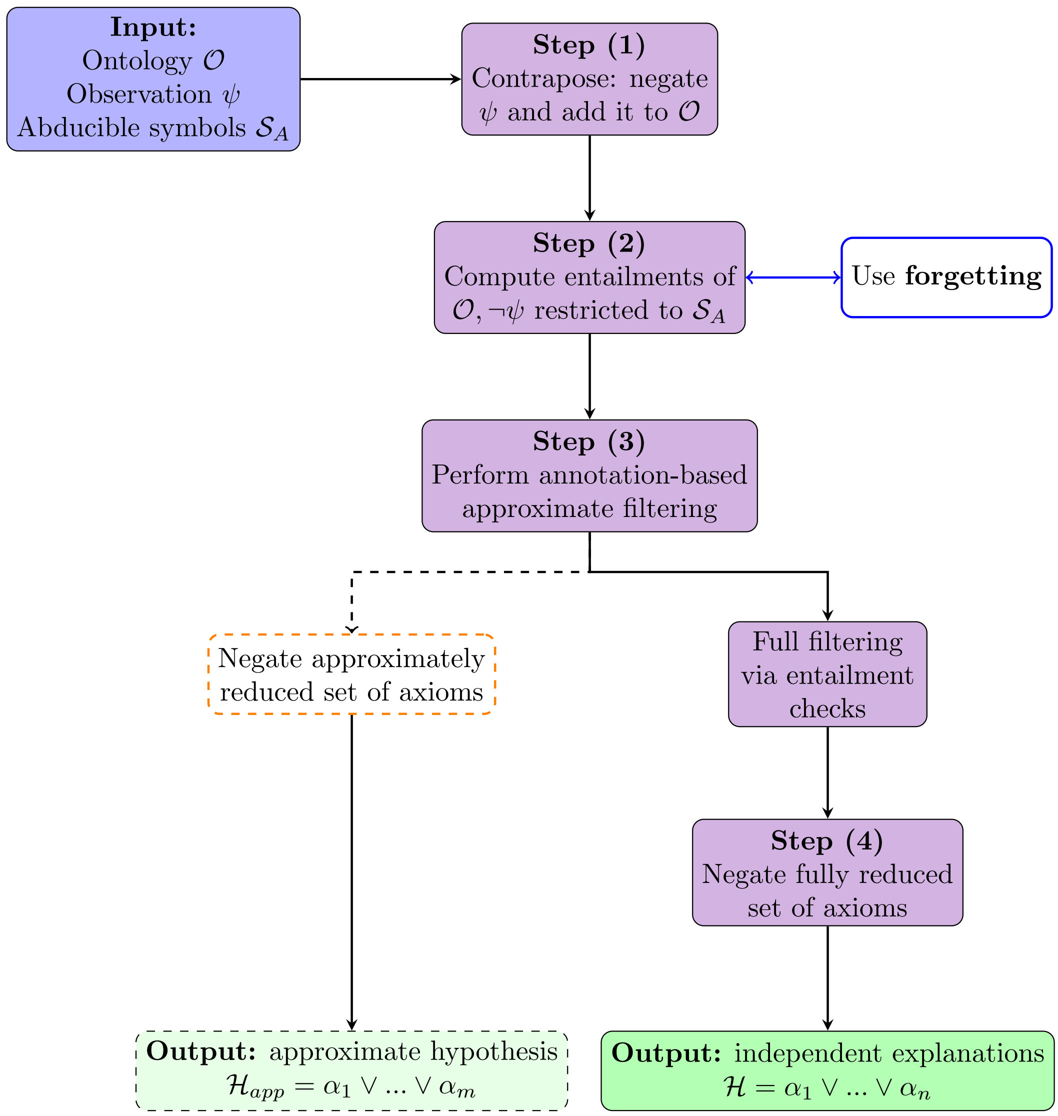

Figure 5.2: Steps in the abduction algorithm for computing hypotheses as spaces of independent explanations, utilising annotation-based filtering for efficient redundancy elimination.

an alternative approach to Step (3) that reduces the computational cost of filtering while maintaining the property that the hypothesis takes the form of a disjunction of independent explanations.

The following example illustrates the above notions.

Example 5.4.4. *Consider the ontology:*

$$\begin{aligned}\mathcal{O} = \{&A \sqsubseteq \exists r.B,\\ &E \sqsubseteq C,\\ &C \sqsubseteq F,\\ &C(e),\\ &r(a,b)\}\end{aligned}$$

with an observation consisting of two concept assertions: $\psi = \{\exists r.B(a), C(d)\}$. If the set of abducibles is specified as $\mathcal{S}_A = \{A, B, E, F\}$, then the following is provided as input to the forgetting step in Figure 5.1 after the negation of ψ and the normalisation of $(\mathcal{O}, \neg\psi)$:

1)$\neg A(x) \vee \exists r.D_1(x)$

2)$\neg D_1(x) \vee B(x)$

3)$\neg E(x) \vee C(x)$

4)$\neg C(x) \vee F(x)$

5)$C(e)$

6)$r(a,b)$

7)$\forall r.D_2(a) \vee \neg C(d)$ *(from $\neg\psi$)*

8)$\neg D_2(x) \vee \neg B(x)$

the result of forgetting $\mathcal{F} = \{r, C\}$ is then computed as follows:

9)$F(e)$	*Resolution(4, 5)*
10)$\forall r.D_2(a) \vee \neg E(d)$	*Resolution(3, 7)*
11)$\neg E(x) \vee F(x)$	*Resolution(3, 4)*
12)$\neg A(a) \vee \neg C(d) \vee \exists r.D_{12}(a)$	*Role Propagation(1, 7)*

13)$\neg D_{12}(x) \vee D_1(x)$	
14)$\neg D_{12}(x) \vee D_2(x)$	
15)$\neg D_{12}(x) \vee B(x)$	*Resolution(2, 13)*
16)$\neg D_{12}(x) \vee \neg B(x)$	*Resolution(8, 14)*
17)$\neg D_{12}(x)$	*Resolution(15, 16)*
18)$\neg A(a) \vee \neg C(d)$	$\exists$ *-Role Restriction Elimination(12, 17)*
19)$\neg A(a) \vee \neg E(d)$	*Resolution(3, 18)*
20)$\neg C(d) \vee D_2(b)$	*Role Instantiation(6, 10)*
21)$\neg C(d) \vee \neg B(b)$	*Resolution(8, 20)*
22)$\neg E(d) \vee \neg B(b)$	*Resolution(3, 21)*

after elimination of definers and elimination of clauses containing symbols in $\mathcal{F}$, the following forgetting solution is obtained:

$$\mathcal{V} = \{E \sqsubseteq F, F(e), \neg A(a) \vee \neg E(d), \neg B(b) \vee \neg E(d)\}$$

Here, the axioms $E \sqsubseteq C$ and $C \sqsubseteq F$ in $\mathcal{O}$, corresponding to clauses (3) and (4), are the parents *of the axiom $E \sqsubseteq F$ corresponding to clause (11). Thus, $E \sqsubseteq F$ is* not *dependent on $\neg\psi$. The same can be said of $F(e)$, which has the parents $C \sqsubseteq F$ and $C(e)$, both of which occur in $\mathcal{O}$. The remaining two axioms in $\mathcal{V}$ depend upon the observation $\neg\psi$, since they have an ancestor in $\neg\psi$ represented by clause (7). Thus, the forgetting solution $\mathcal{V}$ can be reduced to obtain:*

$$\mathcal{V}^* = \{\neg A(a) \vee \neg E(d), \neg B(b) \vee \neg E(d)\}$$

In this paper, dependency tracing is achieved by using *annotations*, similar to [KS17, KC17, PMIMS17]. These take the form of fresh concept names that do not occur in the signature of the ontology nor the observation. Annotations act as labels that are disjunctively appended to existing axioms. They are then used to trace which axioms are the ancestors of inferred axioms. This relies on the fact that the annotation concept is not included in the forgetting signature $\mathcal{F}$. Thus, it will carry over from the parent to the result of any inference in $Int_{\mathcal{ALC}}$, as formalised in the following property:

Theorem 5.4.1. *Let $\mathcal{O}$ be an ontology, ψ an observation as a set of ABox axioms, $\mathcal{F}$ a forgetting signature and ℓ a fresh annotator concept added as an extra disjunct to each clause in the clausal form of $\neg\psi$ where $\ell \notin sig(\mathcal{O} \cup \psi)$ and $\ell \notin \mathcal{F}$. For every axiom β in the solution of forgetting $\mathcal{F}$ from $(\mathcal{O}, \neg\psi)$ using the calculus $Int_{\mathcal{ALC}}$, denoted as $\mathcal{V}$, β is dependent on $\neg\psi$ iff $\ell \in sig(\beta)$.*

Therefore, the presence of the annotation concept in the signature of an inferred axiom indicates that the axiom has at least one ancestor in $\neg\psi$. Since the aim is to trace dependency specifically on $\neg\psi$, only clauses that are part of $\neg\psi$ need to be annotated. As it is not important which specific clauses in $\neg\psi$ were used in the derivation of dependent axioms, only one annotation concept name is required. This will be referred to as ℓ. Using this technique, the process of extracting $\mathcal{V}^*_{app}$ from the uniform interpolant $\mathcal{V}$ is a matter of removing all axioms in $\mathcal{V}$ that do not contain ℓ. Then, ℓ can be replaced with $\bot$ to obtain the annotation-free set $\mathcal{V}^*_{app}$.

The use of dependency tracing to reduce the cost of Step (3) in Figure 5.1 is motivated by the fact that all axioms in the forgetting solution $\mathcal{V}$ that are not dependent on $\neg\psi$ are guaranteed to be redundant with respect to the abduction problem. Lemma 5.4.1 captures this fact.

Lemma 5.4.1. *Let $\mathcal{V}$ be the forgetting solution of $\mathcal{O}, \neg\psi$ in a given signature of abducibles $\mathcal{S}_A$, and let β_i be an axiom in $\mathcal{V}$ such that β_i is not dependent on $\neg\psi$. Then it is the case that β_i is redundant with respect to the abduction problem in Definition 5.1.1.*

Proof: For every axiom β_i in $\mathcal{V}$ we have that $\mathcal{O}, \neg\psi \models \beta_i$ from the definition of uniform interpolation and the soundness of $Int_{\mathcal{ALC}}$. If β_i is not dependent on $\neg\psi$ then it has no ancestors in $\neg\psi$, i.e., it was derived solely from axioms in $\mathcal{O}$. Therefore it must be the case that $\mathcal{O} \models \beta_i$. As a result, β_i will not satisfy the dual of Definition 5.1.1 condition (iii):

$$\mathcal{O}, \beta_1, ..., \beta_{i-1}, \beta_{i+1}, ..., \beta_n \not\models \beta_i$$

and the hypothesis containing $\neg\beta_i$ as a disjunct will contain inter-disjunct redundancy.

Since this annotation-based filtering is sound, i.e., it only removes axioms that are not dependent on ψ, as these are directly derivable from $\mathcal{O}$ and are thus guaranteed to be redundant, it can be used at the start of Step (3) to compute $\mathcal{V}^*_{app}$. The soundness of the annotation-based filtering is shown in Section 5.5.

To guarantee the computation of the reduced uniform interpolant $\mathcal{V}^*$, the entailment check corresponding to the dual of the inter-disjunct redundancy condition (iii) in Definition 5.1.1 must then be performed for each axiom $\beta \in \mathcal{V}^*_{app}$. This is due to the fact that not all axioms that are dependent upon $\neg\psi$ are guaranteed to be relevant to the abduction problem. Since some axioms may have multiple derivations, they can contain the annotation concept but still be redundant with respect to the dual of Definition 5.1.1(iii). For example:

Example 5.4.5. *Consider the following abduction problem:*

$$\begin{aligned}\mathcal{O} =& \{A \sqsubseteq C,\\ & B \sqsubseteq C,\\ & A \sqcap D \sqsubseteq \bot,\\ & D(a)\}\\ \psi =& C(a)\\ \mathcal{S}_A =& \{A, B, D\}\end{aligned}$$

The annotated form of $\neg\psi$ is $\neg\psi = \ell \sqcup \neg C(a)$. Using $\mathcal{F} = \{C\}$, the result of forgetting in Step (2) is:

$$\mathcal{V} = \{A \sqcap D \sqsubseteq \bot, D(a), (\ell \sqcup \neg A)(a), (\ell \sqcup \neg B)(a)\}$$

Note: no inference is made with $D(a)$, since $D \notin \mathcal{F}$. In Step (3) extracting all axioms with annotations and setting $\ell = \bot$ results in the following set of axioms:

$$\mathcal{V}^*_{app} = \{\neg A(a), \neg B(a)\}$$

Despite $\neg A(a)$ being derivable using $\neg\psi$, it follows from the original ontology $\mathcal{O}$ and is therefore redundant with respect to Definition 5.1.1(iii). This can be detected by performing the entailment check in Step (3), and this axiom should therefore be removed from $\mathcal{V}^*_{app}$ to obtain the reduced forgetting solution $\mathcal{V}^*$.

As discussed earlier, $\mathcal{V}^*$ will only contain ABox axioms, possibly including disjunctive assertions. This is due to the fact that $\mathcal{V}^*_{app}$, and by extension $\mathcal{V}^*$, contain only axioms that have at least one ancestor from $\neg\psi$. Since $\neg\psi$ is composed entirely of concept assertions (ground), any descendents of $\neg\psi$ must also be ground. This can be confirmed by examining the rules in Figure 3.2, where at least one if not both of the premises are ground.

This method of filtering out redundancies has several advantages. First, it is not specific to $\mathcal{ALC}$ and can be applied if the abduction method is later extended to more expressive DL languages. Second, by removing axioms that are not dependent on ψ, the method reduces the cost of Step (3) as checking the signature of each axiom for the presence of ℓ is linear in the size of $\mathcal{V}$. In the worst case $\mathcal{V}^*_{app}$ is equal to $\mathcal{V}$ and a double exponential number of entailment checks are still required. For this worst case scenario to occur, it must be the case that all axioms in $\mathcal{V}$ have at least one ancestor in $\neg\psi$. Since it is usually the case that the observation ψ is smaller than the background knowledge in $\mathcal{O}$, this would require one of several situations to occur: ψ contains a collection of the most common symbols in $\mathcal{O}$, which are then specified as non-abducible, or most of the signature of $\mathcal{O}$ must be forgotten. In practice, as demonstrated by the experiments in Section 5.6, this is unlikely: $\mathcal{V}^*_{app}$ is usually a small fraction of $\mathcal{V}$ as shown by the results in Table 5.3 (pg. 127).

Outside of the worst case, each redundancy eliminated from $\mathcal{V}$ to $\mathcal{V}^*_{app}$ replaces an exponential check with a linear one, reducing the computational cost of Step (3).

The entailment checks that must be performed on $\mathcal{V}^*_{app}$ to compute $\mathcal{V}^*$ may still be costly in the event that many axioms are dependent on ψ in $\mathcal{V}$. Therefore, we propose that in some cases it may be pragmatic to relax the allowed hypotheses by negating $\mathcal{V}^*_{app}$ instead of the reduced uniform interpolant $\mathcal{V}^*$ itself. In this case, an additional check, $\mathcal{O},\mathcal{H} \not\models \bot$, is required to rule out inconsistent hypotheses if all of the axioms in $\mathcal{V}^*_{app}$ are redundant. This can occur if there is no explanation for the observation ψ within the signature of abducibles $\mathcal{S}_A$. This approximate approach results in a hypothesis $\mathcal{H}_{app}$ which satisfies conditions (i), (ii) and a weaker form of (iv) in Definition 5.1.1, i.e., it does not satisfy condition (iii).

To summarise, we suggest two realisations of Step (3) of the proposed abduction method. (a) *Approximate* filtering. This computes an approximation of the hypothesis denoted by $\mathcal{H}_{app}$ by negating the approximately reduced $\mathcal{V}^*_{app}$. (b) *Full* filtering, which performs the entailment check in Step (3) for each axiom in $\mathcal{V}^*_{app}$ to obtain $\mathcal{V}^*$ and thus the hypothesis $\mathcal{H}$ which is guaranteed to fully satisfy Definition 5.1.1. Note that for setting (b), the approximation step is still used to reduce the overall cost of Step (3).

In practice, the full filtering step uses an external DL reasoner to check the dual of Definition 5.1.1 condition (iii), i.e., the following:

$$\mathcal{O}, \beta_1, ..., \beta_{i-1}, \beta_{i+1}, ..., \beta_n \not\models \beta_i$$

for each axiom $\beta_i \in \mathcal{V}^*_{app}$, thereby extracting the reduced forgetting solution $\mathcal{V}^*$. As a consequence of this, since there are no DL reasoners that can currently handle fixpoint operators, it is not possible to determine whether or not a disjunct containing a fixpoint operator is redundant with respect to condition (iii). However, greatest fixpoints can be simulated using fresh concept symbols: by replacing $\nu X.C[X]$ with a fresh concept symbol D and adding an axiom of the form $D \sqsubseteq C[D \to X]$. As the forgetting solution, and hence $\mathcal{V}^*_{app}$, are guaranteed to only contain greatest fixpoint operators, it is possible to retain fixpoints when checking the redundancy of all of the other axioms. For example, using the reasoner HermiT [GHM$^+$14], the check above is reduced to the satisfiability check $\mathcal{O}, \beta_1, ..., \beta_{i-1}, \beta_{i+1}, ..., \beta_n, \neg\beta_i \not\models \bot$. If a greatest fixpoint occurs in any of the β_j such that $j \neq i$, then the check proceeds as normal. Otherwise, if a fixpoint operator occurs in β_i, i.e., under negation, then it is not possible to complete the check. As a result, in the presence of redundant fixpoints in $\mathcal{V}^*_{app}$, the final hypothesis will still be guaranteed to satisfy conditions (i), (ii) and (iv) of Definition 5.1.1, with the caveat that condition (iii) is satisfied up to possible redundancy of disjuncts in $\mathcal{H}$ containing fixpoint operators.

There are also several ways of applying the above filtering to eliminate redundant disjuncts based on whether or not flattening is applied to $\mathcal{V}^*_{app}$ and by extension the hypothesis obtained. Flattening refers to the process of pulling out nested disjunctions, or in the forgetting solution conjunctions. This is illustrated by Example 5.4.6.

Example 5.4.6. *Consider the following abduction problem:*

$$\begin{aligned}\mathcal{O} = \{&\exists r.(A \sqcup B) \sqsubseteq \exists r.C,\\ &\exists r.B \sqsubseteq D,\\ &D \sqsubseteq \exists r.C\}\\ \psi =& \exists r.C(a)\\ \mathcal{S}_A =& \{r, A, B, D\}\end{aligned}$$

Application of $Int_{\mathcal{ALC}}$ proceeds as follows, where clauses (1)–(8) are the result of the normal form transformation on $\mathcal{O}, \neg\psi$:

1) $\forall r.D_1(x) \vee \exists r.D_2(x)$

2) $\neg D_1(x) \vee \neg A(x)$

3) $\neg D_1(x) \vee \neg B(x)$

4) $\neg D_2(x) \vee C(x)$

5) $\forall r.D_3(x) \vee D(x)$

6) $\neg D(x) \vee \exists r.D_2(x)$

7)$\forall r.D_3 \vee \ell(a)$	
8)$\neg D_3(x) \vee \neg C(x)$	
9)$\forall r.D_1 \vee \exists r.D_{23} \vee \ell(a)$	*Role Propagation(1, 7)*
10)$\neg D_{23}(x) \vee D_2(x)$	
11)$\neg D_{23}(x) \vee D_3(x)$	
12)$\neg D_{23}(x) \vee C(x)$	*Resolution(4, 10)*
13)$\neg D_{23}(x) \vee \neg C(x)$	*Resolution(8, 11)*
14)$\neg D_{23}(x)$	*Resolution(12, 13)*
15)$\forall r.D_1 \vee \ell(a)$	$\exists$*-Role Restriction Elimination(9, 14)*
16)$\neg D(x) \vee \exists r.D_{23} \vee \ell(a)$	*Role Propagation(6, 7)*
17)$\neg D(x) \vee \ell(a)$	$\exists$*-Role Restriction Elimination(16, 17)*

Note that role propagation is only applied to enable further resolution inferences on symbols in $\mathcal{F}$ [Koo15]. After the application of definer elimination, removal of clauses containing $\mathcal{F}$, subsumption deletion and elimination of the annotation concept ℓ, the approximate reduced uniform interpolant consists of clause 15, where D_1 is replaced by $\neg A \sqcap \neg B$ due to definer elimination with clauses 2 and 3, and clause 17:

$$\mathcal{V}^*_{app} = \{\forall r.(\neg A \sqcap \neg B)(a), \neg D(a)\}$$

To extract $\mathcal{V}^*$, the dual of Definition 5.1.1(iii) is then checked using an external reasoner. Since $\mathcal{O}, \forall r.(\neg A \sqcap \neg B)(a) \not\models \neg D(a)$ and $\mathcal{O}, \neg D(a) \not\models \forall r.(\neg A \sqcap \neg B)(a)$, the final hypothesis is:

$$\mathcal{H}_1 = \exists r.(A \sqcup B)(a) \vee D(a)$$

However, if flattening is first applied, then the universal quantifier in $\forall r.(\neg A \sqcap \neg B)(a)$

distributes, resulting in:

$$\mathcal{V}^*_{app} = \{\forall r.\neg A(a), \forall r.\neg B(a), \neg D(a)\}$$

Performing the entailment checks to extract $\mathcal{V}^*$ then proceeds as before. This time, $\mathcal{O}, \neg D(a) \models \forall r.\neg B(a)$, and thus this axiom is removed. The resulting hypothesis is then:

$$\mathcal{H}_2 = \exists r.A(a) \vee D(a)$$

In the above example if we consider "disjuncts" to be possibly unflattened, i.e. $\exists r.(A \sqcup B)$ would be seen as a single disjunct, then both candidate hypotheses satisfy Definition 5.1.1 since both satisfy condition (iii) and $\mathcal{O}, \mathcal{H}_1 \models \mathcal{H}_2$ and $\mathcal{O}, \mathcal{H}_2 \models \mathcal{H}_1$. However, if flattening is assumed to be applied to $\mathcal{V}^*_{app}$, the optimal solution is $\mathcal{H}_2$.

To achieve this it is necessary to ensure that all conjunctions in $\mathcal{V}^*_{app}$, or dually all disjunctions in $\mathcal{H}$, are pulled out prior to performing the inter-disjunct redundancy check. Since the output of forgetting in Step (2) is already assumed to be in CNF, the following transformation is sufficient to ensure that all conjunctions in $\mathcal{V}$, and hence all disjuncts in $\mathcal{H}$ are pulled out and checked for redundancy according to Definition 5.1.1(iii) individually:

$$\forall r.(C \sqcap D) \qquad \Longleftrightarrow \qquad \forall r.C \sqcap \forall r.D$$

This transformation ensures that all concept assertions of the form $\exists r.(C \sqcup D)(a)$ occuring in $\mathcal{H}$ are flattened to $\exists r.C(a) \vee \exists r.D(a)$.

5.4.3 Remark on Role Forgetting

In prior work [DS19a], only concept forgetting was utilised for abduction, i.e., it was assumed that the signature of abducibles $\mathcal{S}_A$ contained all role symbols in $sig(\mathcal{O}, \psi)$. However, it has since been determined that role forgetting is complete for the abduction setting in this chapter, assuming that role assertions are not present in the observation nor the hypothesis.

Example 5.4.7. *Consider the following abduction problem:*

$$\mathcal{O} = \{C \sqsubseteq \exists r.D\}$$
$$\psi = \{\exists r.D(a)\}$$
$$\mathcal{S}_A = \{C, D\}$$

The following derivation should occur during forgetting in practice:

$1)\neg C \vee \exists r.D_1$	
$2)\neg D_1 \vee D$	
$3)(\forall r.D_2 \vee \ell)(a)$	
$4)\neg D_2 \vee \neg D$	
$5)(\neg C \vee \exists r.D_{12} \vee \ell)(a)$	$RolePropagation(1,3)$
$6)\neg D_{12} \vee D_1$	
$7)\neg D_{12} \vee D_2$	
$8)\neg D_{12} \vee D$	$Resolution(2,6)$
$9)\neg D_{12} \vee \neg D$	$Resolution(4,7)$
$10)\neg D_{12}$	$Resolution(8,9)$
$11)(\neg C \vee \ell)(a)$	$\exists - RoleRestrictionElimination(5,10)$

the reduced forgetting solution, after elimination of definer symbols and clauses containing non-abducibles, is $\mathcal{V}^* = \neg C(a)$. The resulting hypothesis is therefore:

$$\mathcal{H} = C(a)$$

as expected. Note, this hypothesis can also be reached by specifying the signature *of abducibles as $\mathcal{S}_A = \{r, C\}$. However, as the above shows, role forgetting is not problematic in this case.*

Due to an implementation issue in the version of LETHE used at the time of publication, the forgetting solution obtained for the above example was instead $\mathcal{V} = \{\emptyset\}$. Thus, no hypothesis could be found despite the existence of a hypothesis satisfying Definition 5.1.1. Specifically, it is necessary to ensure that the restriction on performing resolution only on symbols in $\mathcal{F}$ is loosened for pairs of clauses of the form $\{\neg D_{12} \vee C, \neg D_{12} \vee \neg C\}$ [Koo15]. This ensures that the existential role restriction elimination rule can be applied as in Example 5.4.7.

However, the incompleteness of the $Int_{\mathcal{ALC}}$ calculus for role forgetting with respect to abduction is still apparent in the case of TBox abduction, for example:

Example 5.4.8. *Consider the following instance of the abduction problem:*

$$\begin{aligned}
\mathcal{O} =& \{\emptyset\} \\
\psi =& \{\exists r.C \sqsubseteq \exists r.D\} \\
\mathcal{S}_A =& \{C, D\}
\end{aligned}$$

There are several ways to represent the negation of the observation. Here, it is assumed that it takes the form $\neg\psi = (\exists r.C \sqcap \forall r.\neg D)(a^*)$, where a^* is a fresh individual. Including the annotation process, inferences could then be performed as follows:

$1)(\exists r.D_1 \vee \ell)(a*)$

$2)\neg D_1 \vee C$

$3)(\forall r.D_2 \vee \ell)(a*)$

$4)\neg D_2 \vee \neg D$

$5)(\exists r.D_{12} \vee \ell)(a*)$ $\quad RolePropagation(1,3)$

$\ldots$

Even if role propagation is applied between clauses 1 and 5, the resulting reduced *forgetting solution would be $\mathcal{V}^* = \{\emptyset\}$. This is due to the fact that the only clauses dependent on $\neg\psi$ contain the symbol r, all of which will be deleted during the removal of clauses containing symbols in $\mathcal{F}$.*

In the above TBox abduction example, role forgetting is indeed incomplete, since the expected TBox abduction hypothesis in the signature $\mathcal{S}_A = \{C, D\}$ is:

$$\mathcal{H} = C \sqsubseteq D$$

TBox abduction is studied in more detail in Chapter 7, which lifts the solution to include TBox abduction as part of the more general task of Knowledge Base abduction.

5.5 Properties of the Approach

In this section proofs for the soundness and completeness of the abduction approach, with respect to returning hypotheses satisfying the conditions in Definition 5.1.1, are provided. The approach taken is the full approach in Figure 5.2: computing the semantically minimal space of independent explanations using annotation-based filtering as a pre-processing step, before eliminating any remaining redundant explanations. Therefore, this section will also cover the soundness of the annotation-based filtering approach, with respect to eliminating only redundant explanations, since this approximation step is necessary to ensure that the method remains tractable over large ontologies.

5.5.1 Soundness and Completeness

To begin proving the soundness and completeness of the forgetting-based abduction approach, it is necessary to consider the soundness and completeness of the forgetting step in Step (2), and the filtering step in Step (3). For the forgetting step, it is essential that the calculus used is sound and complete with respect to the forgetting problem. From the abduction context, this means that the forgetting solution returned is correct and satisfies the expected characteristic of being the strongest necessary condition of the input $\mathcal{O}, \neg\psi$ provided in Step (2).

Recalling from Definition 3.5.1 and Definition 3.5.2, the soundness and interpolation completeness of $Int_{\mathcal{ALC}}$ with respect to forgetting over $\mathcal{ALC}\nu(\vee)$ ontologies have been proven in [KS15b, Koo15]. Since the abduction problem in this chapter requires forgetting over ontologies expressed in $\mathcal{ALC}(\vee)$, i.e. $\mathcal{O}, \neg\psi$ where ψ is a conjunction of $\mathcal{ALC}$ concept assertions, these proofs also hold in the context provided in this section. Importantly, this means that the forgetting solution returned in Step (2) is indeed a strongest necessary condition of the input $\mathcal{O}, \neg\psi$.

Theorem 5.5.1. *The calculus $Int_{\mathcal{ALC}}$ is sound and interpolation complete for computing forgetting solutions of ontologies expressed in $\mathcal{ALC}\nu$ with disjunctive assertions.*

From this it is the case that if a result $\mathcal{V}$ is returned in Step (2), then it is a correct solution to forgetting non-abducibles from $\mathcal{O}, \neg\psi$. This means that it is also the strongest necessary condition of $\mathcal{O}, \neg\psi$ in the signature $\mathcal{S}_A$ as in Theorem 3.6.1.

In Step (3), the filtering step ensures that the returned hypothesis satisfies the inter-disjunct redundancy requirement in Definition 5.1.1 condition (iii). This is done by eliminating all axioms in $\mathcal{V}$ that do not satisfy the dual of the aforementioned condition. There are two cases to be considered: first, the case where the check:

$$\mathcal{O}, \beta_1, ..., \beta_{i-1}, \beta_{i+1}, ..., \beta_n \models \beta_i$$

is applied exhaustively to every axiom β_i in $\mathcal{V}$. The second case is to apply the above check only to the result of the approximate filtering step $\mathcal{V}^*_{app}$. In the first case, performing the above entailment check exhaustively using an external reasoner guarantees that the reduced forgetting solution $\mathcal{V}^*$ is obtained by removing every axiom β_i in $\mathcal{V}$ for which the entailment holds.

The following Lemmas cover key properties of the forgetting solution $\mathcal{V}$ and the reduced forgetting solution $\mathcal{V}^*$ that are useful in proving the soundness and completeness of the approach with respect to Definition 5.1.1.

Lemma 5.5.1. $\mathcal{O}, \neg\psi \models \mathcal{V}$

Proof: The soundness of $Int_{\mathcal{ALC}}$ has been proven in [KS13, KS15b]. Thus, the set $\mathcal{V}$ is a forgetting solution of the input $(\mathcal{O}, \neg\psi)$ and satisfies the conditions in Definition 3.4.2 for the signature $\mathcal{S}_A$. The lemma then follows from Theorem 3.6.1: if $\mathcal{V}$ is a strongest necessary condition of $\neg\psi$ under $\mathcal{O}$ in the signature $\mathcal{S}_A$, then trivially $\mathcal{O}, \neg\psi \models \mathcal{V}$.

The subsequent Lemmas follow from the definition of Step (3) of the method: the reduction of the forgetting solution to only the set $\mathcal{V}^*$ of axioms such that for each $\beta_i \in \mathcal{V}^*$, $\mathcal{O}, \beta_1, ..., \beta_{i-1}, \beta_{i+1}, ..., \beta_n \not\models \beta_i$.

Lemma 5.5.2. $\mathcal{O}, \neg\psi \models \mathcal{V}^*$

Proof: Given that $\mathcal{O}, \neg\psi \models \mathcal{V}$ and $\mathcal{V}^* \subseteq \mathcal{V}$, it then follows that $\mathcal{O}, \neg\psi \models \mathcal{V}^*$.

Lemma 5.5.3. $\mathcal{O} \not\models \beta$ *for every* $\beta \in \mathcal{V}^*$

Proof: Since the extraction of $\mathcal{V}^*$ from $\mathcal{V}$ requires omitting all axioms $\beta_i \in \mathcal{V}$ such that $\mathcal{O}, \beta_1, ..., \beta_{i-1}, \beta_{i+1}, ..., \beta_n \models \beta_i$ via the annotation-based filtering followed by entailment checking on any remaining axioms, it follows that $\beta_i \notin \mathcal{V}^*$ and for every $\beta \in \mathcal{V}^*$ it is the case that $\mathcal{O} \not\models \beta$.

Note that Lemma 5.5.3 implies that $\mathcal{O} \not\models \mathcal{V}^*$.

Lemma 5.5.4. $\mathcal{O}, \mathcal{V}^* \models \mathcal{V} \setminus \mathcal{V}^*$

Proof: The reduction of $\mathcal{V}$ to $\mathcal{V}^*$ is performed sequentially. Thus, we can define a sequence:

$$\mathcal{V}_0, \mathcal{V}_1, ..., \mathcal{V}_n$$

where $\mathcal{V}_0 = \mathcal{V}$, $\mathcal{V}_n = \mathcal{V}^*$ and for each i with $1 \leq i < n$:

$$\mathcal{O}, \mathcal{V}_i \setminus \{\beta_i\} \models \beta_i \tag{5.1}$$

and

$$\mathcal{V}_{i+1} = \mathcal{V}_i \setminus \{\beta_i\} \tag{5.2}$$

where $\beta_i \in \mathcal{V} \setminus \mathcal{V}^*$ is the redundant axiom removed at step i. Now we can prove the lemma by induction.

The base case is as follows:

$$\mathcal{O}, \mathcal{V}_0 \models \beta$$

i.e., where no axioms have been identified as redundant and thus $\mathcal{V}^* = \{\emptyset\}$. The base case holds trivially since $\beta \in \mathcal{V} \setminus \mathcal{V}^*$ reduces to $\beta \in \mathcal{V}$ as $\mathcal{V}^* = \{\emptyset\}$. We now define the induction hypothesis as follows:

$$\mathcal{O}, \mathcal{V}_i \setminus \{\beta_i\} \models \beta$$

and the induction step:

$$\mathcal{O}, \mathcal{V}_{i+1} \setminus \{\beta_{i+1}\} \models \beta$$

There are three possible cases for the induction step. (i) $\beta \notin \beta_0, ..., \beta_{i+1}$, i.e., the axiom β has not yet been discarded as of step $i+1$. Then the induction step holds since $\beta \in \mathcal{V}_{i+1} \setminus \beta_{i+1}$. (ii) $\beta = \beta_{i+1}$, i.e., the axiom β is removed at step $i+1$. Then statement (1) holds at step $i+1$ under the definition of redundancy in Definition 5.1.1 condition (iii), and thus the induction step holds. (iii) $\beta \in \beta_0, ..., \beta_i$, i.e., β was checked and discarded prior to step $i+1$. Then from (1):

$$\mathcal{O}, \mathcal{V}_{i+1} \setminus \{\beta_{i+1}\} \models \beta_{i+1}$$

and we can also write:

$$\mathcal{O}, \mathcal{V}_{i+1} \setminus \{\beta_{i+1}\} \models \mathcal{O}, \mathcal{V}_{i+1} \setminus \{\beta_{i+1}\}, \beta_{i+1}$$

Which simplifies to $\mathcal{O}, \mathcal{V}_{i+1} \setminus \{\beta_{i+1}\} \models \mathcal{O}, \mathcal{V}_{i+1}$. By substituting statement (2) into this, we obtain:

$$\mathcal{O}, \mathcal{V}_{i+1} \setminus \{\beta_{i+1}\} \models \mathcal{O}, \mathcal{V}_i \setminus \{\beta_i\}$$

From the induction hypothesis, $\mathcal{O}, \mathcal{V}_i \setminus \{\beta_i\} \models \beta$. Thus, the following holds:

$$\mathcal{O}, \mathcal{V}_{i+1} \setminus \{\beta_{i+1}\} \models \beta$$

meaning that the induction step holds for all $\beta \in \mathcal{V} \setminus \mathcal{V}^*$. As a result, we have that:

$$\mathcal{O}, \mathcal{V}_i \setminus \{\beta_i\} \models \beta$$

for all $1 \leq i < n$.

Lemma 5.5.5. *For any $\mathcal{W}$ in the signature $\mathcal{S}_A$ such that $\mathcal{O}, \beta_1, ..., \beta_{i-1}, \beta_{i+1}, ..., \beta_n \not\models \beta_i$ for every $\beta_i \in \mathcal{W}$ and $\mathcal{O}, \neg\psi \models \mathcal{W}$, $\mathcal{O}, \mathcal{V}^* \models \mathcal{W}$.*

Proof: We have that $\mathcal{O}, \neg\psi \models \mathcal{W}$. We also have that $\mathcal{O}, \mathcal{V} \models \mathcal{W}$, since $\mathcal{V}$ is a strongest necessary condition of $\neg\psi$ under $\mathcal{O}$ in the signature $\mathcal{S}_A$. Since we can write $\mathcal{V}$ in the following way: $\mathcal{V} = (\mathcal{V} \setminus \mathcal{V}^*) \cup \mathcal{V}^*$, it is the case that $\mathcal{O}, (\mathcal{V} \setminus \mathcal{V}^*), \mathcal{V}^* \models \mathcal{W}$

From Lemma 5.5.4, we have that $\mathcal{O}, \mathcal{V}^* \models \mathcal{V} \setminus \mathcal{V}^*$ and from this $\mathcal{O}, \mathcal{V}^* \models \mathcal{O}, \mathcal{V} \setminus \mathcal{V}^*, \mathcal{V}^*$ trivially follows, therefore we can write:

$$\mathcal{O}, \mathcal{V}^* \models \mathcal{W}$$

as required.

Using Lemmas 5.5.1–5.5.4, it is possible to prove the soundness of the method with respect to the abduction problem in Definition 5.1.1.

Theorem 5.5.2. *For any abduction problem $\langle \mathcal{O}, \psi, \mathcal{S}_A \rangle$, where $\mathcal{O}$ and ψ are expressed in $\mathcal{ALC}$ and the set of solutions excludes any hypothesis containing a role assertion, the abduction method produces a hypothesis satisfying Definition 5.1.1, up to potential redundancy of disjuncts containing fixpoint operators if present.*

Proof: **Soundness.** We obtain the hypothesis $\mathcal{H}$ by negating $\mathcal{V}^*$ under contrapositive reasoning, which is then added to $\mathcal{O}$. Thus, the consistency requirement in Definition 5.1.1 condition (i) follows from Lemma 5.5.3 since $\beta_i \equiv \neg\alpha_i$ for all $\beta_i \in \mathcal{V}^*$ and thus $\mathcal{O}, \alpha_i \not\models \bot$ for every disjunct $\alpha_i \in \mathcal{H}$. Condition (ii) follows from Lemma 5.5.2: since $\mathcal{O}, \neg\psi \models \mathcal{V}^*$, under contraposition $\mathcal{O}, \mathcal{H} \models \psi$ where $\mathcal{H} \equiv \neg\mathcal{V}^*$. Condition (iii) is guaranteed via the strict check performed in Step (3) of the method, which is the dual of condition (iii). Thus, since $\mathcal{H}$ is obtained by applying contraposition to $\mathcal{V}^*$, and all axioms in $\mathcal{V}^*$ satisfy the check in Step (3), $\mathcal{H}$ will satisfy condition (iii). Condition (iv) follows from Lemma 5.5.5, which shows that if there exists a set of axioms $\mathcal{W}$ in the signature $\mathcal{S}_A$ such that $\mathcal{O}, \neg\psi \models \mathcal{W}$ and the set $\mathcal{W}$ satisfies the dual of condition (iii) then $\mathcal{O}, \mathcal{V}^* \models \mathcal{W}$. Since the hypothesis $\mathcal{H}$ is obtained by negating $\mathcal{V}^*$, the dual of Lemma 5.5.5 holds for $\mathcal{H}$: i.e., if there exists a $\mathcal{H}'$ such that $\mathcal{H}' = \neg\mathcal{W}$ then $\mathcal{O}, \mathcal{H} \models \mathcal{H}'$.

Completeness. This property follows directly from the interpolation completeness of $Int_{\mathcal{ALC}}$ [Koo15]. For any given combination of an ontology $\mathcal{O}$, negated observation $\neg\psi$ and forgetting signature $\mathcal{F}$, a uniform interpolant $\mathcal{V}$ is returned using $Int_{\mathcal{ALC}}$ such that for any other uniform interpolant $\mathcal{V}'$ of $(\mathcal{O}, \neg\psi)^{-\mathcal{F}}$, the property $\mathcal{V} \equiv \mathcal{V}'$ holds. Thus, the set $\mathcal{V}^*$ which satisfies the properties in Lemma 5.5.5 is always obtained from $\mathcal{V}$ as required. The result of applying contrapositive reasoning to $\mathcal{V}^*$ is then the hypothesis $\mathcal{H}$ such that for any other consistent hypothesis $\mathcal{H}'$ in the restricted signature $\mathcal{S}_A$, $\mathcal{O}, \mathcal{H} \models \mathcal{H}'$.

In Theorem 5.5.2, the restriction that disjuncts containing fixpoint operators may be redundant under condition (iii) is due to the fact that, in practice, there are currently no DL reasoners that can support fixpoints as pointed out in Section 5.4.2. Therefore, it is not possible to check whether a disjunct containing a fixpoint is entailed by the rest of the hypothesis. However, since the full filtering check is performed over the axioms in $\mathcal{V}^*_{app}$, and these are guaranteed to only contain greatest (not least) fixpoints, it is possible to retain disjuncts containing fixpoints when testing the redundancy of all of the other disjuncts in the hypothesis. This is due to the fact that greatest fixpoints can be simulated using fresh symbols by replacing $\nu X.C[X]$ with a fresh concept symbol

D and adding an axiom $D \sqsubseteq C[D \rightarrow X]$. For the check $\mathcal{O}, \beta_1, ..., \beta_{i-1}, \beta_{i+1}, ..., \beta_n \models \beta_i$ if a fixpoint occurs within some β_j such that $1 \leq j \leq n, j \neq i$ then the check proceeds as normal, while if a fixpoint occurs in β_i, then it is not possible to determine if the entailment holds in practice. As a result, any other redundancies in the hypothesis are eliminated, including those that are redundant with respect to a fixpoint operator. Consequently, conditions (i), (ii) and (iv) are still satisfied in the presence of fixpoint operators in $\mathcal{H}$.

5.5.2 Soundness of Annotation-Based Filtering

The annotation-based filtering method is used as preprocessing to reduce the cost of Step (3) by first computing $\mathcal{V}^*_{app}$ before applying the entailment check. Given that the soundness of the abduction method without this optimisation has been proven, it is now necessary to prove that this procedure is also sound to ensure that the overall method remains both sound and complete.

From this Lemma, it is safe to assume that removing all axioms that are not dependent on $\neg\psi$ will not jeopardise the soundness of the abduction procedure: no valid explanations are lost and thus the semantic minimality of the resulting hypothesis is maintained. Now it remains to show that the annotation-based filtering is sound in the sense that it only removes axioms that are not dependent on $\neg\psi$. To do this, it is necessary to prove that the annotator concept ℓ carries from the premises of an inference in $Int_{\mathcal{ALC}}$ to the conclusion, and thus that ℓ will appear in any axiom in $\mathcal{V}$ that is dependent on $\neg\psi$ in the presented method.

Theorem 5.4.1 *Let $\mathcal{O}$ be an ontology, ψ an observation as a set of ABox axioms, $\mathcal{F}$ a forgetting signature and ℓ an annotator concept added as an extra disjunct to each clause in the clausal form of $\neg\psi$ where $\ell \notin sig(\mathcal{O} \cup \psi)$ and $\ell \notin \mathcal{F}$. For every axiom β in the uniform interpolant $\mathcal{V} = (\mathcal{O}, \neg\psi)^{-\mathcal{F}}$, β is dependent on $\neg\psi$ iff $\ell \in sig(\beta)$.*

Proof: The proof is by induction over the way the derivation is constructed in $Int_{\mathcal{ALC}}$. The base case is the start of the derivation, where no inference has been performed yet. So we consider any clause β in $(\mathcal{O}, \neg\psi)$: the input to the abduction method. For an axiom β to be dependent on $\neg\psi$, it must have at least one ancestor in $Cls(\neg\psi)$, where Cls denotes the clausal form of $\neg\psi$. Since in the base case no inferences have been performed, the only way for an axiom β to have an ancestor in $Cls(\neg\psi)$ is if $\beta \in Cls(\neg\psi)$, as no other dependent axioms have been derived. Thus, the only axioms dependent on $\neg\psi$ are those in the negated observation due to the reflexivity of the ancestor relation.

Now consider the following set of axioms $\{\nu_1, ..., \nu_k, \nu_{k+1}\}$ where each ν_i is the conclusion of an inference rule in $Int_{\mathcal{ALC}}$ between ν_{i-1} and another axiom, where $\nu_1 \in \neg\psi$. Since we must prove a characteristic of dependent axioms, the inferences must all have at least one ancestor in $\neg\psi$. Thus, the set of inferences begins with $\nu_1 \in \neg\psi$. The induction hypothesis is that ν_k contains the annotator concept ℓ: i.e., $\ell \in sig(\nu_k)$. Now for the induction step: two cases must be considered for the axiom ν_{k+1} . Given that one of the parents of ν_{k+1} is ν_k, the other parent β can be (a) an axiom not dependent on $\neg\psi$ or (b) another axiom that is dependent on $\neg\psi$. In both cases, the inference can be made using any of the rules in Figure 3.2. The case where both parent axioms are not dependent on $\neg\psi$ need not be considered, since the aim is to show that ℓ is present in all axioms dependent on $\neg\psi$ and the definition of dependency requires at least one ancestor to be in $\neg\psi$.

For case (a), where β does not depend on $\neg\psi$:

(1) Resolution: Consider $\beta = (C_1 \vee \neg C_2)(t_1)$ and $\nu_k = C_2 \vee \ell(t_2)$ where σ is the unifier of t_1 and t_2 if it exists. Resolution occurs on C_2 as follows:

$$\frac{(C_1 \vee \neg C_2)(t_1) \qquad C_2 \vee \ell(t_2)}{(C_1 \vee \ell)\sigma}$$

therefore $\ell \in sig(\nu_{k+1})$, since $\ell\sigma = \ell$.

(2) Role propagation: Two cases are considered: (i) ν_k contains an existential quantifier and (ii) ν_k contains a universal quantifier. Consider $\beta = (C \vee \forall r.D_1)(t_1)$ and (i) $\nu_k = \exists r.D_2 \vee \ell(t_2)$, (ii) $\nu_k = \forall r.D_2 \vee \ell(t_2)$ where D_1 and D_2 are definer symbols and σ

is the unifier of t_1 and t_2 if it exists. Role propagation occurs on r as follows:

$$\text{(i)}\quad \frac{(C \vee \forall r.D_1)(t_1) \qquad \exists r.D_2 \vee \ell(t_2)}{(C \vee \ell)\sigma \vee \exists r.D_{12}(t_1\sigma)}$$

therefore $\ell \in sig(\nu_{k+1})$.

$$\text{(ii)}\quad \frac{(C \vee \forall r.D_1)(t_1) \qquad \forall r.D_2 \vee \ell(t_2)}{(C \vee \ell)\sigma \vee \forall r.D_{12}(t_1\sigma)}$$

therefore $\ell \in sig(\nu_{k+1})$.

(3) Existential role restriction elimination: Only one case needs to be considered for ν_k, which is: $\nu_k = (C \vee \ell \vee \exists r.D_1)(t)$. This is due to the fact that the annotator concept ℓ is appended disjunctively to clauses in $\neg\psi$ and thus does not occur under quantifiers. Thus, no definer symbols will be introduced in place of concepts containing ℓ. Let $\beta = \neg D(x)$, then existential role restriction elimination is applied as follows:

$$\frac{(C \vee \ell \vee \exists r.D_1)(t) \qquad \neg D_1(x)}{C \vee \ell}$$

therefore $\ell \in sig(\nu_{k+1})$.

(4) Role instantiation: Since the observation may not contain role assertions, only one case needs to be considered for ν_k, which is: $\nu_k = (C_1 \vee \ell \vee (\forall.D)(t_1)$. Let $\beta = r(t_2, b)$, then role instantiation is applied as follows:

$$\frac{(C_1 \vee \ell \vee (\forall r.D))(t_1) \qquad r(t_2, b)}{(C_1 \vee \ell)\sigma \vee D(b)}$$

therefore $\ell \in sig(\nu_{k+1})$.

For case (b), where β is dependent on $\neg\psi$: the derivations are largely the same. For the resolution and role propagation rule the result of the inference simply contains $\ell \vee \ell$, which simplifies to ℓ. For the existential role restriction elimination and role instantiation rules, it is not possible for the second axiom to also be dependent on $\neg\psi$. This is due to the fact that ℓ does not fall under the scope of a quantifier, thus there will be no clause of the form $\neg D \vee \ell$, where D is a definer, introduced during the transformation to the normal form required by $Int_{\mathcal{ALC}}$. Additionally, ψ does not contain negated role assertions, so there will also be no clause of the form $r(a,b) \vee \ell$.

Thus, each axiom in $\mathcal{V}$ that has at least one ancestor in $\neg\psi$ will contain the annotator concept ℓ. Since ℓ is not present in $\mathcal{O}$, having been disjunctively appended to $\neg\psi$, the approximation of the reduced uniform interpolant $\mathcal{V}^*$ obtained by eliminating all axioms β_i such that $\ell \notin sig(\beta_i)$ will always take the form of a set of axioms that are dependent on $\neg\psi$.

Since the annotation-based filtering is sound, i.e. it only removes a subset of the axioms in $\mathcal{V}$ that are redundant under Definition 5.1.1(iii), it does not compromise the soundness and completeness of the method that was proven earlier.

5.5.3 Complexity

Theorem 5.5.3. *In the worst case, computing a hypothesis $\mathcal{H}$ using the proposed forgetting-based abduction method has 3EXPTIME upper bound complexity for running time and the size of $\mathcal{H}$ can be double exponential in the size of* $(\mathcal{O}, \psi)$.

Proof: The main source of complexity for the presented abduction method is in the use of the forgetting method $Int_{\mathcal{ALC}}$. Computing the uniform interpolant $\mathcal{V}$ has 2EXPTIME complexity and the number of clauses in the uniform interpolant is double exponential in the size of the input ontology [KS15b].

As for the extraction of hypotheses from uniform interpolants, this is done by first approximating $\mathcal{V}^*$ using the annotation-based filtering method described in Section 5.4.2, resulting in $\mathcal{V}^*_{app}$. This relies on checking the signature of each axiom in $\mathcal{V}$ for the presence of the annotator concept ℓ. The complexity of this filtering method is linear in the size of the uniform interpolant $\mathcal{V}$.

In the case where the fully reduced uniform interpolant $\mathcal{V}^*$ is computed, the additional check $\mathcal{O}, \{\beta_1, ..., \beta_{i-1}, \beta_{i+1}, ..., \beta_n\} \not\models \beta_i$ for each remaining axiom β_i in $\mathcal{V}^*_{app}$ is needed to remove remaining redundancies. In the worst case, $\mathcal{V}^*_{app}$ could be equal to $\mathcal{V}$, and thus could be double exponential in size with respect to $(\mathcal{O}, \psi)$. In this case, this additional step would require a double exponential number of exponential time entailment checks. Thus, the worst case complexity of this step is 3EXPTIME and the overall worst case time complexity of the proposed abduction method is 3EXPTIME.

It is worth noting that the worst-case time complexity and hypothesis size above is unlikely to occur in practice for large ontologies, since this requires $\mathcal{V}^*$ to be equal to $\mathcal{V}$. For this to occur, it must be the case that every inference performed on $\mathcal{O}, \neg\psi$ using $Int_{\mathcal{ALC}}$ involved an axiom in, or dependent on, $\neg\psi$. In other words, for $\mathcal{V}^*_{app}$ to be equal to $\mathcal{V}$, every axiom in $\mathcal{V}^*$ corresponds to a non-redundant disjunct in $\mathcal{H}$ when negated.

In practice, the experiments in Table 5.3 indicate that the size of $\mathcal{V}^*_{app}$ tends to be much smaller than that of $\mathcal{V}$, supporting the intuition above.

5.6 Experimental Evaluation

A Java prototype was implemented using the OWL-API[5] and the forgetting tool LETHE which implements the $Int_{\mathcal{ALC}}$ method.[6] Using this, two experiments were carried out over a corpus of real world ontologies, which were preprocessed into their $\mathcal{ALC}$ fragments. Axioms not representable in $\mathcal{ALC}$, such as number restrictions of the form $\leq nr.C$ where r is a role symbol and C is a concept symbol, were removed. Others were represented using appropriate $\mathcal{ALC}$ axioms where possible. For example, a range restriction $\exists \mathrm{r}^-.\top \sqsubseteq C$ was converted to $\top \sqsubseteq \forall r.C$, where r^- is the inverse role of r.

The experiments in this section provide an initial evaluation of forgetting-based approaches to abduction in DL ontologies. A small corpus was used for these experiments so that a fine-grained approach could be taken to examining the size of the

[5]http://owlapi.sourceforge.net/
[6]http://www.cs.man.ac.uk/ koopmanp/lethe/index.html

hypotheses returned, the number of redundancies eliminated and the benefit of the filtering approaches described in Section 5.4.2 for each individual ontology in the corpus.

The choice of ontologies was based on several factors. They must be consistent, parsable using LETHE and the OWL API and must vary in size to determine how this impacts performance. Since many real-world ontologies are encoded in less expressive DLs such as $\mathcal{EL}$, the corpus was also split between $\mathcal{EL}$ and $\mathcal{ALC}$ to determine if the performance over $\mathcal{EL}$ suffers as a result of the additional capabilities of the method for $\mathcal{ALC}$. The final corpus contains ontologies from the NCBO Bioportal and OBO repositories,[7,8] and the LUBM [GPH05] and Semintec ontologies.[9] The characteristics of the corpus are summarised in Table 5.1. The experiments were performed on a machine using a 4.00GHz Intel Core i7-6700K CPU and 16GB RAM.

Ontology Name	DL	TBox Size	ABox Size	Num. Concepts	Num. Roles
BFO	$\mathcal{EL}$	52	0	35	0
LUBM	$\mathcal{EL}$	87	0	44	24
HOM	$\mathcal{EL}$	83	0	66	0
DOID	$\mathcal{EL}$	7892	0	11663	15
SYN	$\mathcal{EL}$	15352	0	14462	0
ICF	$\mathcal{ALC}$	1910	6597	1597	41
Semintec	$\mathcal{ALC}$	199	65189	61	16
OBI	$\mathcal{ALC}$	28888	196	3691	67
NATPRO	$\mathcal{ALC}$	68565	42763	9464	12

Table 5.1: Characteristics of the experimental corpus.

For each ontology in the corpus, 30 observations satisfying the requirements for the abduction problem in Definition 5.1.1 were generated, i.e., the observations were each consistent with and not entailed by the given ontology. The observations were generated, prior to the running of the experiments, in the following way: each axiom in the ontology was processed and the concepts occurring in them were stored. For a TBox axiom of the form $C \sqsubseteq D$ or $C \equiv D$, where C and D are any arbitrary concepts expressible in $\mathcal{ALC}$, this means that the concepts C and D were be stored. For the

[7]https://bioportal.bioontology.org/
[8]http://www.obofoundry.org/
[9]http://www.cs.put.poznan.pl/alawrynowicz/semintec.htm

ABox, concepts occuring in concept assertions of the form $C(a)$ were stored. Using the stored concepts, observations as concept assertions were generated using concepts from those stored from the background ontology. The concept occurring in an observation was randomly chosen from one of the following, where C and D are two concepts from the stored list: C, $\neg C$, $C \sqcap D$, $C \sqcup D$, $\forall r.C$ and $\exists r.C$. Each candidate observation was checked against the requirements in Definition 5.1.1. If a candidate observation did not satisfy these requirements, it was discarded and the generation procedure was reattempted.

It was necessary to develop a new experimental design for evaluating abductive reasoning in DL ontologies due to the lack of existing benchmarks for the task. Few experimental evaluations of abductive reasoning in DLs exist, and the existing methods are not applicable to the problem solved by the algorithm in this Chapter: to the best of our knowledge, no other abductive reasoning method in DLs produces a semantically minimal space of independent explanations as required in Definition 5.1.1. The motivation behind the approach used for observation generation was to avoid trivial observations, i.e. observations that are too simple or do not satisfy the abduction problem requirements, while simultaneously avoiding the problem of generating arbitrarily complex observations. As a result, the concepts in each ontology were used as a guide for the minimum complexity of unseen observations, while randomly combining these with $\mathcal{ALC}$ operators encouraged variety particularly in simpler ontologies..

For the first experiment, the forgetting signature $\mathcal{F}$ was specified as one random concept symbol from $sig(\psi)$. Therefore the task was to compute the "most semantically minimal" space of independent explanations for a given observation, i.e. the semantically minimal hypothesis for the largest set of abducibles. The assumption was that users may first seek the most general hypothesis as a starting point for investigating unseen phenomena. This also allows the user to pursue stronger hypotheses subsequently by forgetting further symbols from the initial space of explanations incrementally. The temporal results for first experiment (Table 5.2) are therefore indicative of the expected time taken for a single increment, i.e., computing the next strongest hypothesis, while the results on hypothesis size and number of redundancies (Table 5.3) are indicative of the size of the initial hypothesis and the number of redundancies removed from the forgetting solution.

The aim of the second experiment was to investigate how the size of the forgetting signature, i.e., the number of non-abducibles specified, impacted the performance of the abduction algorithm. While the impact of forgetting signature size has been investigated for the problem of forgetting [KS15b, Koo15], the impact in the setting of abduction has not. Particularly, the number of inferences made during the forgetting step using $Int_{\mathcal{ALC}}$ is expected to have a direct impact on the performance of the filtering step. The benefit of using the annotation-based filtering approach as the forgetting signature size increases was therefore investigated via this experiment. Three ontologies from the corpus in Table 5.1 were used: DOID, ICF and SYN. These ontologies were chosen as they have a sufficiently large signature of concepts and LETHE did not time out when forgetting in any case. Thus, results for the time taken during the filtering step were available for every run of the experiment. In all cases, at least one symbol from ψ was included in $\mathcal{F}$ to avoid trivial hypotheses.

In both experiments the two filtering approaches for Step (3), illustrated in Figure 5.2 and described in Section 5.4.2, were compared for the same observations and same random selection of $\mathcal{F}$. The first is the *approximate* filtering approach, which uses the annotation-based filtering and the second is the *full* filtering approach, which performs the entailment check corresponding to the dual of Definition 5.1.1 condition (iii) over the approximate result. The DL reasoner HermiT [GHM$^+$14] was used to perform these entailment checks. Thus, the tradeoff between the additional time for entailment checking and redundancy in the final hypothesis was evaluated. In all cases, LETHE was subject to a 300 second time limit, while the filtering step was not subject to a time limit.

However, in extreme cases the filtering step was terminated if the runtime for each individual abduction problem exceeded several hours. In each of these cases, indicated by the *t.o.* entires in Table 5.2, the whole experiment was terminated over the corresponding ontology. This decision was made both for practical purposes and due to the purpose of the corresponding experiments:

The prototype used to perform the experiments in this Chapter makes use of the OWL-API, which does not allow disjunctive assertions over multiple individuals. As a result, it was not possible to represent observations consisting of a conjunction of concept assertions over different individuals. Due to this limitation, the experiments in this section were limited to observations involving one individual. For the filtering in Step (3), the preference relation used in these experiments was simply based on order of appearance of each disjunct.

Tables 5.2 and 5.3 show the results for the first experiment. Across all of the ontologies the amount of time taken to filter the forgetting solution, i.e., to ensure the hypothesis returned was a space of independent explanations satisfying Definition 5.1.1, was shorter when using the annotation-based filtering as opposed to performing entailment checks for every axiom in the forgetting solution. This is illustrated by the results in the $\mathcal{V}^*$ and "$\mathcal{V}^*$ no app" columns in Table 5.2: for the smaller $\mathcal{EL}$ ontologies, BFO, LUBM and HOM, the time taken using the proposed filtering approach was between 11.1%–27.8% of the time taken using only entailment checking. For the

Ont.	Mean Time Taken /s			Max Time Taken /s		
Name	$\mathcal{V}^*_{app}$	$\mathcal{V}^*$	$\mathcal{V}^*$ no app.	$\mathcal{V}^*_{app}$	$\mathcal{V}^*$	$\mathcal{V}^*$ no app.
BFO	0.01	0.01	0.09	0.01	0.07	0.14
LUBM	0.02	0.03	0.30	0.11	0.16	1.21
HOM	0.03	0.05	0.18	0.40	0.54	0.86
DOID	0.44	1.09	1071.35	1.11	6.98	1095.07
SYN	0.95	3.92	2421.96	2.33	61.52	2593.13
ICF	0.30	0.56	t.o.	0.52	1.58	t.o.
Semin.	3.13	5.12	t.o.	9.29	15.36	t.o.
OBI*	3.82	32.17	t.o.	25.18	95.37	t.o.
NATP.	26.54	179.70	t.o.	39.51	544.50	t.o.

Table 5.2: Computation time statistics for 30 observations using a forgetting signature size of 1. * indicates that LETHE did not terminate within the 300s time limit in at least one case, "t.o." indicates that the experiment was terminated after several days runtime.

Ont.	Mean Redund. Removed		Size $\mathcal{H}$ /disjuncts		Mean % of
	$\mathcal{V} \rightarrow \mathcal{V}^*_{app}$	$\mathcal{V}^*_{app} \rightarrow \mathcal{V}^*$	Mean	Max	$\mathcal{H}_{app}$ Redund.
BFO	52	0	1.97	4	0
LUBM	90	0.80	2.73	11	29.30
HOM	82	0.03	2.07	13	1.45
DOID	7891	0	7.23	104	0
SYN	15351	0.03	20.63	457	0.15
ICF	8505	0	2.30	7	0
Semin.	72827	0.03	3.60	10	0.83
OBI*	29191	6.48	52.48	161	12.35
NATP.	111318	0.03	48.70	204	0.06

Table 5.3: Redundancy removal and hypothesis size statistics over 30 observations using a forgetting signature size of 1. The $\mathcal{H}$ for which mean and maximum sizes are reported is the result of negating the fully reduced forgetting solution $\mathcal{V}^*$ (fully satisfying the abduction problem). * indicates that LETHE did not terminate within the 300s time limit in at least one case.

largest $\mathcal{EL}$ ontologies, DOID and SYN, the corresponding results were 0.1% and 0.2%. Over the $\mathcal{ALC}$ ontologies, the difference was more pronounced: using only entailment checking the time taken to eliminate redundancies in the forgetting solution exceeded several hours. These results indicate that the benefit of the proposed filtering approach is more pronounced, both in absolute and proportional terms, as the size of the ontology increases and particularly as the complexity of the language is increased from $\mathcal{EL}$ to $\mathcal{ALC}$. This is as expected: for a larger ontology, the corresponding forgetting solution when eliminated the chosen concept is likely to be larger and thus the number of entailment checks required to ensure no redundant explanations are present in the hypothesis will increase. Since the complexity of entailment checking is polynomial in $\mathcal{EL}$ and exponential in $\mathcal{ALC}$, the effect of replacing an entailment check with a linear one is more pronounced in the $\mathcal{ALC}$ ontologies.

The results in Table 5.2 also provide a comparison between the approximate filtering and the full filtering settings for Step (3) in Figure 5.2. For the smaller $\mathcal{EL}$ ontologies, the difference in time taken between the approximate and full filtering was small. For the larger ontologies the cost of the full filtering was more pronounced, taking 313%, 742% and 577% longer across the SYN, OBI and NATPRO ontologies respectively.

With respect to the number of redundancies eliminated: in all cases, it can be seen from Table 5.3 that the annotation-based filtering eliminated the majority of redundancies, while entailment checking was required for a small portion of the total redundancies. This is illustrated by the $\mathcal{V} \to \mathcal{V}^*_{app}$ and $\mathcal{V}^*_{app} \to \mathcal{V}^*$ columns, which indicate the mean number of redundancies removed in the two stages of Step (3) of Figure 5.2: the annotation-based filtering and additional entailment checks respectively. Over the BFO, DOID and ICF ontologies for all 30 observations all of the redundancies were eliminated by the annotation-based filtering, and additional entailment checks were not required to reduce the forgetting solution any further. For the remaining ontologies, the redundancies that required entailment checks to remove accounted for no more than 1% of the total redundancies in the forgetting solution.

Figure 5.3 shows the results of the second experiment, the aim of which was to investigate the impact of the forgetting signature size, i.e., number of non-abducibles, on the time taken for both forgetting and filtering, as in Steps (2) and (3) of Figure 5.2. The time taken for the forgetting step, Step (2), increased almost linearly with the size of $\mathcal{F}$. This was expected due to a higher number of inferences needed to compute $\mathcal{V}$. The time taken for filtering, Step (3), did not increase with the size of $\mathcal{F}$. This is likely due to the fact that the size of the forgetting signature does not correlate directly with the size of the resulting forgetting solution. As a result, the number of checks for the annotation concept ℓ and the number of entailment checks do not depend upon the size of the forgetting signature. However, for each ontology, maxima were observed for different sizes of $\mathcal{F}$. This may be due to the fact that the content of the forgetting signature has a more significant impact on the number of axioms in the forgetting solution than the absolute size of the forgetting signature itself, i.e., including certain symbols in $\mathcal{F}$ increases the filtering time. This may be explained by the possibility that forgetting commonly used concepts can result in more inferences and a larger $\mathcal{V}$, therefore increasing the number of checks that must be performed during the filtering step. The full filtering took an average of 27, 11 and 70 times longer than the approximate case

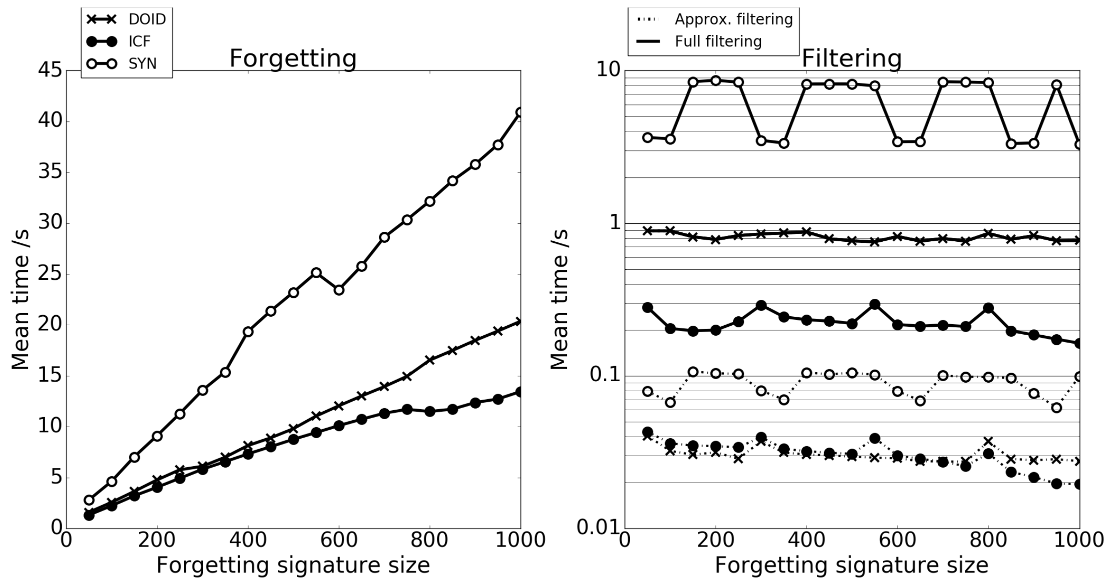

Figure 5.3: Mean forgetting and filtering times with varying $\mathcal{F}$ signature sizes for the ICF, DOID and SYN ontologies.

for the DOID, ICF and SYN ontologies respectively. This indicates that the cost of the full entailment check increased with the size of the ontology, particularly the size of the TBox, not the size of $\mathcal{F}$.

In 100% of cases for both experiments the hypotheses were represented without fixpoints, indicating that cyclic, semantically minimal hypotheses seem rare in practice.

Chapter 6

ABox Abduction via Semantic Forgetting

In this Chapter, the use of semantic forgetting as part of the second step in the proposed abduction method is investigated. Specifically, the problem is to compute the least assumptive set of independent explanations as in previous Chapters. The setting of the problem remains ABox abduction in the DL $\mathcal{ALC}$, thereby providing a comparison to the approach presented in the previous chapter. Semantic forgetting is investigated via the forgetting system FAME, described in Section 3.5.2, which is utilised in the proposed abduction method where the hypotheses are expressed in $\mathcal{ALCOI}(\nabla)$. This Chapter aims to compare the two approaches to forgetting in terms of their use in the forgetting step of the forgetting-based abduction algorithm. The necessary adaptations to the abduction procedure that are necessary to utilise semantic forgetting as implemented by FAME are discussed, and the comparison between the two forgetting approaches is made through the use of specific examples and an experimental evaluation over a corpus of ontologies.

6.1 Motivation for Investigation

Investigating the use of forgetting for abductive reasoning in the setting of DL ontologies is one of the aims of this thesis. The forgetting method utilised as part of the abduction system in the previous chapter takes the *uniform interpolation* perspective on forgetting, i.e., the notion of *weak forgetting* described in Section 3.4. As part of the investigation into forgetting-based approaches to abduction, it is useful to also consider the *semantic (strong) forgetting* perspective on abduction [ZS15, ZS16, Zha18]. As discussed in Chapter 3, this view is closely related to second-order quantifier elimination [DLS01, GSS08] and the model-theoretic notion of forgetting in Definition 3.4.3. This is a different perspective to the consequence-based notion of uniform interpolation, which aims to preserve all entailments that are representable in the restricted signature $\mathcal{S}_A$. However, from the perspective of abduction, the semantic forgetting and uniform interpolation solutions to the forgetting problem serve the same purpose. This purpose is to produce the strongest necessary condition in the, language and signature specified, for the provided input which is in this case the background ontology $\mathcal{O}$ together with the negated observation $\neg\psi$. Another motivation for the investigation in this chapter is to extend the capabilities of forgetting-based abduction compared to the approach used in Chapter 5. The main limitation of the approach presented in Chapter 5 is the fact that neither the observations nor the hypotheses are permitted to contain role assertions of the form $r(a,b)$. This limitation restricts the ability to utilise existing (ground) information regarding specific individuals in the background knowledge, as the following example illustrates:

Example 6.1.1. *Consider the following abduction problem:*

$$\begin{aligned}\mathcal{O} = \{&\exists r.B \sqsubseteq A,\\ &B(b)\}\\ \psi =&A(a)\\ \mathcal{S}_A = &\{B, r\}\end{aligned}$$

For the signature $\mathcal{S}_A$, the hypothesis satisfying Definition 5.1.1 would be:

$$\mathcal{H}_1 = \exists r.B(a)$$

If the signature of abducibles is changed to $\mathcal{S}_A = \{r\}$, then the correct hypothesis is:

$$\mathcal{H}_2 = r(a,b)$$

The first hypothesis $\mathcal{H}_1$ is reachable using the $Int_{\mathcal{ALC}}$ calculus (Figure 3.2) for the forgetting step. However, the hypothesis $\mathcal{H}_2$ is not. This is due to the fact that the $Int_{\mathcal{ALC}}$ calculus does not support deriving negated role assertions of the form $\neg r(a,b)$, as it is designed only to preserve all entailments of the input that are expressible in $\mathcal{ALC}$. Thus, these are not obtained in the reduced forgetting solution $\mathcal{V}^*$ and role assertions will be absent from the hypothesis when $\mathcal{V}^*$ is negated in the final step of the abduction algorithm. This restriction means that the system is not able to utilise existing relationships between individuals in the observation and those in the ABox of the background ontology when generating hypotheses. Thus, many of the more specific hypotheses such as $\mathcal{H}_2$ in the example above are not reachable.

Therefore in this chapter an alternative forgetting approach, taking the semantic forgetting perspective, is investigated to compare how the use of such an approach compares not only in terms of computational cost but also in terms of the forms of the hypotheses produced. An alternative to overcome the limitation in Example 6.1.1 would be to extend the calculus of $Int_{\mathcal{ALC}}$ to enable inferences on negated role assertions and thus enable both the input and output of the system to contain role assertions. This option is the subject of Chapter 7.

6.2 Extending the ABox Abduction Approach

The general steps in the abduction algorithm specified in Figures 5.1 and 5.2 remain the same. Since the method relies on contraposition, Step (1) negates the observation $\neg\psi$

and provides this, as well as a signature of non-abducibles $\mathcal{F}$ to be forgotten, as input to Step (2). Step (3) eliminates the redundancies in the forgetting solution $\mathcal{V}$, resulting in a reduced forgetting solution $\mathcal{V}^*$ and thereby eliminating redundant explanations (disjuncts) in the hypothesis. Finally, in Step (4) the reduced forgetting solution $\mathcal{V}^*$ is negated resulting in a hypothesis $\mathcal{H}$ as a space, represented as a disjunction, of independent explanations for the observation $\neg\psi$.

However, to compare the weak and semantic (strong) forgetting approaches for the abduction problem, here Step (2) of the abduction algorithm utilises an implementation of the semantic forgetting approach as opposed to the uniform interpolation approach in Chapter 5. This is provided by the algorithm FAME [ZS16], the calculus and properties of which are discussed in Chapter 3 Section 3.5. The system has been implemented and evaluated via experiments in recent work [ZS15, ZS16], showing promising performance for the forgetting problem. As such, it provides a promising basis for investigating the use of semantic forgetting as part of the abduction approach for solving the problem specified in Definition 4.2.1. By utilising FAME for the forgetting step of the abduction algorithm in Figure 5.1, the expressivity of the problems that can be solved is increased. In this work, the main benefit of this additional expressivity is the ability to include nominals in the input $\mathcal{O}, \neg\psi$. Particularly, this provides an alternative way to represent the negation of observations that include role assertions.

The other steps of the abduction algorithm presented in the previous chapter must also be modified to account for the use of a new forgetting procedure and the fact that the DL language used to represent the forgetting solutions is more expressive. Furthermore, the abduction problems that are being solved are now also more expressive, including the presence of role assertions in both observations and hypotheses. As such, care must be taken to ensure that the hypotheses take the form required in Definition 4.2.1 and that redundancy elimination is performed appropriately.

As for the previous abduction problems, the input remains an $\mathcal{ALC}$ ontology $\mathcal{O}$ together with an $\mathcal{ALC}$ observation ψ, which is a conjunctive set of $\mathcal{ALC}$ ABox axioms that may now include role assertions. Since the forgetting approach utilised by FAME requires the conversion of ABox axioms into TBox axioms, which is achievable via the use of nominals, the input can be expressed in $\mathcal{ALCO}$.

However, it is necessary to formulate an appropriate representation for the negation of the observation ψ and the form of the hypothesis $\mathcal{H}$, since FAME operates on and produces TBox axioms containing nominals rather than the corresponding ABox axioms. Example 6.2.1 illustrates a simple case of this.

Example 6.2.1. *Consider the abduction problem in Example 6.1.1. This problem can be reformulated in $\mathcal{ALCO}$ as follows:*

$$\begin{aligned} \mathcal{O} = &\{\exists r.B \sqsubseteq A, \\ &\{b\} \sqsubseteq B\} \\ \psi = &\{a\} \sqsubseteq A \end{aligned}$$

Given the set of abducibles $\mathcal{S}_A = \{B, r\}$, the hypothesis obtained using FAME for the forgetting step of Figure 5.1 is:

$$\mathcal{H}_1 = \{a\} \sqsubseteq \exists r.B$$

If instead $\mathcal{S}_A = \{r\}$, the corresponding hypothesis is:

$$\mathcal{H}_2 = \{a\} \sqsubseteq \exists r.\{b\}$$

Both $\mathcal{H}_1$ and $\mathcal{H}_2$ satisfy the notions in the general abduction problem presented in Definition 4.2.1.

The benefit of the additional expressivity of FAME is demonstrated by $\mathcal{H}_2$, which is equivalent to the role assertion $r(a,b)$. As a result the instance of the general abduction problem, provided in Definition 4.2.1, that is addressed in this Chapter can be expanded compared Chapter 5. Specifically the restriction that each explanation, i.e., disjunct, in $\mathcal{H}$ must take the form of a conjunction of concept assertions can be lifted.

However, before an appropriate instance of the abduction problem can be defined, it is first necessary to determine how the observation and hypothesis should be represented for this problem. In cases where either the observation or the hypothesis take the form of a conjunction or disjunction of ABox assertions, the reformulation is less obvious than for simple cases such as that in Example 6.2.1. For more complex cases, it is possible to take advantage of the fact that FAME can perform forgetting in the presence of the top role ∇.

Example 6.2.2. *Consider the following abduction problem:*

$$\begin{aligned}\mathcal{O} &= \{\exists r.B \sqsubseteq A,\\ &\quad C \sqsubseteq D,\\ &\quad \{b\} \sqsubseteq B\}\\ \psi &= \{A(a),\\ &\quad D(c)\}\\ \mathcal{S}_A &= \{r, C\}\end{aligned}$$

The expected hypothesis under Definition 4.2.1 should be equivalent to:

$$\mathcal{H}_1 = r(a,b) \sqcap C(c)$$

In $\mathcal{ALC}$ with disjunctive assertions, and negated role assertions, the negation of ψ is as follows:

$$\neg\psi = \neg A(a) \vee \neg D(c)$$

An equivalent representation for $\neg\psi$ expressed using nominals and ∇ is as follows:

$$\neg\psi = \top \sqsubseteq \forall\nabla.(\neg\{a\} \sqcup \neg A) \sqcup \forall\nabla.(\neg\{c\} \sqcup \neg D)$$

Following the steps in Figure 5.1 using FAME for the forgetting step, where $\mathcal{F} =$

$\{A,B,D\}$, the hypothesis obtained can be represented as:

$$\mathcal{H}_2 = \top \sqsubseteq \exists\nabla.(\neg\{a\} \sqcup \exists r.\{b\}) \sqcap \exists\nabla.(\neg\{c\} \sqcup C)$$

This is equivalent to the expected hypothesis $\mathcal{H}_1 = r(a,b) \sqcap C(c)$.

Lifting the forms taken in the example above, the general form used for the negated observations is shown in Figure 6.1, noting that the observation in this setting can now also contain role assertions of the form $r(a,b)$, as well as concept assertions. As indicated in Figure 6.1, since FAME does not operate on ABox assertions directly nor disjunctive ABox assertions, the negation of ψ must be expressed as a TBox in $\mathcal{ALCO}(\nabla)$. The form taken by the hypotheses produced is as follows:

$$\mathcal{H}_F = \top \sqsubseteq \exists\nabla.D_1 \sqcup \ldots \sqcup \exists\nabla.D_n \qquad (2)$$

where each D_i is an $\mathcal{ALCOI}(\nabla)$ concept.

Observation, ψ:		**Negated Observation, $\neg\psi$:**
$C(a)$	$\Longrightarrow$	$\top \sqsubseteq \forall\nabla.(\neg\{a\} \sqcup \neg C)$
$r(a,b)$	$\Longrightarrow$	$\top \sqsubseteq \forall\nabla.(\neg\{a\} \sqcup \forall r.\neg\{b\})$
$\prod_{i=1}^{n} C_i(a_i) \sqcap \prod_{j=1}^{m} r(b_j,c_j)$	$\Longrightarrow$	$\top \sqsubseteq \bigsqcup_{i=1}^{n} \forall\nabla.(\neg\{a_i\} \sqcup \neg C_i) \sqcup \ldots$ $\bigsqcup_{j=1}^{m} \forall\nabla.(\neg\{b_j\} \sqcup \forall r.\neg\{c_j\})$

where C and D are arbitrary $\mathcal{ALC}$ concepts, a and b are individuals with corresponding nominals $\{a\}$ and $\{b\}$ and r is a role.

Figure 6.1: Transformation rules for negating an observation when using semantic forgetting based abduction.

Given that the observations may now contain role assertions, it is necessary to extend the proposed abduction problem once more. Additionally, as the hypothesis takes on a different form it is necessary to reformulate the definition to account for this and to suitably retain the notion of a disjunction of individual explanations. The necessary reformulation is given in Definition 6.2.1.

Definition 6.2.1. *Let $\mathcal{O}$ be an ontology and ψ be a set of ABox assertions, both expressed in $\mathcal{ALC}$, such that $\mathcal{O} \not\models \bot$, $\mathcal{O}, \psi \not\models \bot$ and $\mathcal{O} \not\models \psi$. Let $\mathcal{S}_A$ be a set of symbols called* abducibles. *The* ABox abduction problem *is to compute a hypothesis $\mathcal{H} = \alpha_1 \vee ... \vee \alpha_n$ where each α_i takes the form $\top \sqsubseteq \exists \nabla . D_i$ where D_i is an $\mathcal{ALCOI}(\nabla)$ concept. The solution $\mathcal{H}$ must contain only those symbols specified in $\mathcal{S}_A$ and satisfy conditions (i)–(iv) of Definition 4.2.1.*

The general form of the abduction problem remains the same as the one proposed in Definition 4.2.1. The aim is still to compute a hypothesis $\mathcal{H}$ which takes the form of a space of independent explanations for the observation ψ. The main differences here are that the observations may contain role assertions, the hypothesis $\mathcal{H}$ is represented in the DL $\mathcal{ALCOI}(\nabla)$ and that each explanation or disjunct is now in the form $\top \sqsubseteq \exists \nabla . D_i$ where D_i is an $\mathcal{ALCOI}(\nabla)$ concept.

Since the result of the forgetting step is still the strongest necessary entailment of $\mathcal{O}, \neg\psi$, the same rationale can be applied to the result of applying FAME as was applied to the result of LETHE in Step (2) of Figure 5.1. It is still possible to satisfy conditions (iii) and (iv) of Definition 4.2.1 using this representation. However, it is first necessary to adapt the filtering method of step 3 in Figure 5.1 to obtain the reduced forgetting solution $\mathcal{V}^*$, as this is an important part of the feasibility of the approach in practice [DS19a].

As in Chapter 5, an annotation concept ℓ is used to efficiently trace any dependencies on the negated observation $\neg\psi$ in the forgetting result $\mathcal{V}$. Any axioms which do not contain the concept ℓ are removed from $\mathcal{V}$, thereby removing the majority of the axioms that are redundant with respect to the inter-disjunct redundancy requirement in Definition 5.1.1 condition (iii). Fortunately, extending this approach to the current setting is straightforward. Here, the negated observation provided as input to Step (2) of Figure 5.2 is annotated as follows:

$$\neg\psi = \top \sqsubseteq \forall\nabla.(\neg\{a_1\} \sqcup \neg C_1 \sqcup \ell) \sqcup \ldots \sqcup \forall\nabla.(\neg\{a_k\} \sqcup \neg C_k \sqcup \ell)$$

where as before, ℓ is a fresh concept symbol that does not occur in $(\mathcal{O}, \psi)$, nor in the signature $\mathcal{F}$.

As outlined in Chapter 5, the filtering Step (3) in Figure 5.1 can be applied in two ways: *approximate* or *full*. As before, the approximate filtering utilises the annotation-based method to inexpensively remove all redundancies that can be captured using this approach, i.e., removing all axioms in $\mathcal{V}$ that are not dependent upon $\neg\psi$. The result is an approximation of the reduced forgetting result $\mathcal{V}^*$, denoted by $\mathcal{V}^*_{app}$. This is then negated in Step (4) of Figure 5.1 to return an approximate hypothesis. Alternatively, the full filtering setting further performs the dual entailment check of Definition 4.2.1(iii) over each axiom in $\mathcal{V}^*_{app}$ using an external reasoner. This eliminates any remaining redundancies that cannot be captured using annotations, as demonstrated in Example 5.4.5. The result is then $\mathcal{V}^*$, which is negated to return a hypothesis satisfying Definition 6.2.1.

Example 6.2.3 illustrates the full procedure.

Example 6.2.3. *Consider the abduction problem in Example 6.2.2. The input to FAME in Step (2) of Figure 5.1 is:*

$$\begin{aligned}
\mathcal{O} =& \{\exists r.B \sqsubseteq A, \\
& C \sqsubseteq D, \\
& \{b\} \sqsubseteq B\} \\
\neg\psi =& \{\top \sqsubseteq \forall\nabla.(\neg\{a\} \sqcup \neg A \sqcup \ell) \sqcup \forall\nabla.(\neg\{c\} \sqcup \neg D \sqcup \ell)\} \\
\mathcal{S}_A =& \{C, r\}
\end{aligned}$$

From $\mathcal{S}_A$, the set of symbols to be forgotten is $\mathcal{F} = \{A, B, D\}$. During the forgetting *process, $\mathcal{O}, \neg\psi$ is transformed into clausal form:*

$$\forall r.\neg B \sqcup A$$
$$\neg C \sqcup D$$
$$\neg\{b\} \sqcup B$$

while for forgetting A, $\neg\psi$ must also be transformed into A-reduced form:

$$\neg\{a\} \sqcup \neg A \sqcup \ell \sqcup \forall\nabla^{-}.(\forall\nabla.(\neg\{c\} \sqcup \neg D \sqcup \ell))$$

using the Surfacing^C rule of Figure 3.4. The concept A can then be eliminated using the Ackermann rules in Figure 3.5. The same procedure is then performed to eliminate the concept D. After forgetting both A and D, the result is:

$$\neg\{b\} \sqcup B$$
$$\forall\nabla.(\neg\{a\} \sqcup \forall r.\neg B \sqcup \ell) \sqcup \forall\nabla.(\neg\{c\} \sqcup \neg C \sqcup \ell)$$

Forgetting the concept B then produces:

$$\forall\nabla.(\neg\{a\} \sqcup \forall r.\neg\{b\} \sqcup \ell) \sqcup \forall\nabla.(\neg\{c\} \sqcup \neg C \sqcup \ell)$$

which is the forgetting result $\mathcal{V}$. In the filtering step (3) of Figure 5.1, the axiom is retained and the annotation concept ℓ is set to $\bot$. Neither disjunct in this hypothesis is redundant with respect to the dual of Definition 4.2.1(iii) and thus both are retained in the reduced forgetting result $\mathcal{V}^*$, which is then negated in step (3) to produce the hypothesis:

$$\mathcal{H}_F = \top \sqsubseteq \exists\nabla.(\{a\} \sqcap \exists r.\{b\}) \sqcap \exists\nabla.(\{c\} \sqcap C)$$

This is equivalent to the suggested hypothesis $\mathcal{H} = r(a,b) \sqcap C(c)$.

6.3 Comparing Hypotheses

While the main aim of this work is to produce a hypothesis satisfying the notions specified in the general abduction problem in Definition 4.2.1, there is another factor that is important to consider: the syntactic form taken by the hypotheses. Since each explanation in the space represented by the hypothesis $\mathcal{H}$ should represent some independent insight to explain the new observation ψ, the clarity and readability of the hypotheses is important. This is in contrast to the problem of forgetting, where restricting the original ontology while preserving all representable entailments [KS15b] or obtaining an equivalent set of formulae [ZS15] is the main goal and the readability of the forgetting solution is usually not considered as a priority. Thus, the readability of the forgetting result has so far received little attention. Additionally, if one or several of the explanations in $\mathcal{H}$ is to be added to a knowledge base to explain ψ, then producing explanations in a more expressive language than the source ontology could be problematic in some scenarios. For example, it will impact the efficiency of future reasoning applied to the updated ontology, which may be problematic in practical scenarios.

Therefore, it is useful to compare the hypotheses produced by both approaches to forgetting-based abduction: the first using the uniform interpolation approach LETHE, and the second using the semantic forgetting approach of FAME. From here, to differentiate between the hypotheses produced via both approaches: $\mathcal{H}_F$ is used to denote the hypothesis obtained using FAME for the forgetting process in Step (2) of Figure 5.1 and $\mathcal{H}_L$ is used to refer to the one obtained using LETHE in Step (2). Consider the following example:

Example 6.3.1. *Consider the following abduction problem:*

$$
\begin{aligned}
\mathcal{O} = \{ & Pogona \sqsubseteq \exists livesIn.(Arid \sqcap Woodlands), \\
& Woodlands \sqsubseteq Habitat,
\end{aligned}
$$

$$\begin{aligned}
&EucalyptForest \sqsubseteq Woodlands, \\
&EucalpytForest(SpringbrookPark)\} \\
\psi = &\exists livesIn.Woodlands(Gary)
\end{aligned}$$

Case (1): let $\mathcal{S}_A$ include all symbols in $\mathcal{O}$ except Woodlands, i.e. $\mathcal{F} = \{Woodlands\}$. The hypotheses obtained using LETHE and FAME respectively are:

$$\begin{aligned}
\mathcal{H}_L = &Pogona(Gary) \vee \exists livesIn.EucalyptForest(Gary) \\
\mathcal{H}_F = &\top \sqsubseteq \exists \nabla.(Pogona \sqcap \forall livesIn.(\neg Arid \sqcup \neg Habitat \sqcup \exists livesIn^{-}.\{Gary\}) \\
&\sqcup \exists \nabla.(\{Gary\} \sqcap \exists livesIn.EucalyptForest)
\end{aligned}$$

where livesIn^{-} denotes the inverse of the role livesIn.

Example 6.3.1 illustrates a potential drawback of utilising a more expressive forgetting approach: the hypothesis produced can be more difficult to interpret, as seen by the additional complexity of the first disjunct of $\mathcal{H}_F$. Despite this, the extra expressivity in the target language of FAME does confer a benefit in the context of abduction: producing additional explanations. Since the forgetting solution produced when utilising FAME in Step (2) is the strongest set of entailments of $\mathcal{O}, \neg\psi$ in a more expressive language than that obtained when using LETHE, it preserves additional entailments by comparison. This leads to additional disjuncts, i.e., explanations in the final hypothesis. In Example 6.3.1, if $\mathcal{F}$ is extended to $\mathcal{F} = \{Woodlands, EucalyptForest\}$, then the hypothesis produced using LETHE in the forgetting step is $\mathcal{H}_L = Pogona(Gary)$, whereas the hypothesis produced using FAME is as follows:

$$\begin{aligned}
\mathcal{H}_F = \top \sqsubseteq &\exists \nabla.(Pogona \sqcap \forall livesIn.(\neg Arid \sqcup \neg Habitat \sqcup \exists livesIn^{-}.\{Gary\}) \\
&\sqcup \exists \nabla.(\{Gary\} \sqcap \exists livesIn.\{SpringbrookPark\})
\end{aligned}$$

The second disjunct in $\mathcal{H}_F$ is equivalent to $livesIn(Gary, SpringbrookPark)$, an explanation that is absent from $\mathcal{H}_L$. As this case shows, it is possible in some cases to translate the hypotheses obtained to $\mathcal{ALC}$ with disjunctive assertions. This may be desirable in some applications to avoid unnecessary extensions to the language used to express the background knowledge. In this way, further reasoning and modelling remains unchanged by the addition of computed explanations. If C is an $\mathcal{ALCO}$ concept, then the following translations are possible:

$$\top \sqsubseteq \exists \nabla.(\neg\{a\} \sqcup C) \quad \Longleftrightarrow \quad C(a)$$
$$\top \sqsubseteq \exists \nabla.(\neg\{a\} \sqcup \exists r.\{b\}) \quad \Longleftrightarrow \quad r(a,b)$$

It is also worth noting that, in Example 6.3.1, the following relations hold: $\mathcal{O},\mathcal{H}_L \models \mathcal{H}_F$ and $\mathcal{O},\mathcal{H}_F \not\models \mathcal{H}_L$. This indicates that, in this case, the hypothesis obtained by the abduction approach when using FAME for the forgetting step was weaker than the corresponding hypothesis obtained when using LETHE for this step. This is true for both sets of abducibles. In the first case, this is due to the fact that the first explanation, i.e. the first disjunct, of $\mathcal{H}_F$ is weaker than the corresponding explanation in $\mathcal{H}_L$. In the second case, the effect is more apparent: $\mathcal{H}_F$ consists of a disjunction of two explanations, while $\mathcal{H}_L$ contains only one explanation. In the general case, this relationship between $\mathcal{H}_F$ and $\mathcal{H}_L$ is to be expected. This is due to the fact that the forgetting solution computed by FAME can be stronger than the uniform interpolant produced by LETHE, owing to the fact that FAME computes forgetting solutions in a more expressive language than the uniform interpolants computed by LETHE. Thus, $\mathcal{H}_F$ can be weaker than $\mathcal{H}_L$ under the background ontology, since these are obtained by negating the reduced forgetting solutions.

6.4 Properties of the Approach

An important aspect of FAME for this work is the fact that it is sound for forgetting in $\mathcal{ALCOI}(\nabla)$, as expressed in Theorem 3.5.2, which is a weaker form of the theorem in [ZS16]. The aim of this work is to produce hypotheses for abduction problems expressed in $\mathcal{ALC}$. In this setting, all abduction problems given as input can be expressed in $\mathcal{ALCO}(\nabla)$ as indicated in Figure 6.1. As a result, role hierarchies are unnecessary and are excluded from this setting. Role conjunctions are also excluded since they are only needed in the forgetting solution produced by FAME when the input is expressed in $\mathcal{ALCOIH}$ [Zha18]. Thus, Theorem 3.5.2 sufficient for this setting.

One limitation of using FAME to compute the forgetting result is that it is not complete for forgetting, as discussed in Section 3.5.2. As such the abduction approach in this chapter is not complete as it utilises FAME in Step (2). Despite this, the additional expressivity means that additional hypotheses are reachable in certain scenarios, as illustrated by Examples 6.2.1, 6.2.2 and 6.3.1 earlier in this chapter. Therefore there is in a sense a trade-off: while some explanations may be missed, others may be obtained.

The soundness of the filtering approach is expressed below.

Theorem 6.4.1. *Let $\mathcal{O}$ be an $\mathcal{ALCO}(\nabla)$ ontology, ψ an observation as a set of axioms, $\mathcal{F}$ a forgetting signature and ℓ an annotator concept appended disjunctively to each disjunct in $\neg\psi$, where $\ell \notin sig(\mathcal{O})$ and $\ell \notin \mathcal{F}$. For each axiom β in the forgetting result $\mathcal{V}$ obtained by forgetting all symbols in $\mathcal{F}$ via FAME [ZS16], if $\ell \notin sig(\beta)$ then β is redundant under the dual of Definition 4.2.1 condition (iii), and should be removed in the extraction of the reduced forgetting result $\mathcal{V}^*$.*

Proof: The proof is by induction over the construction of a derivation using the calculus of FAME [ZS16], and takes the same form as the proof of Theorem 5.5.2 in Chapter 5 Section 5.5. The annotation concept ℓ does not appear in the signature $\mathcal{F}$. Thus, ℓ is not eliminated and if a clause in the normal form of $(\mathcal{O}, \neg\psi)$ contains the annotation concept ℓ, then any clause derived via inferences on this clause under FAME's forgetting calculus will also contain ℓ. Therefore, any axiom β in the forgetting result $\mathcal{V}$ that does not contain ℓ was derived purely using axioms in the background ontology $\mathcal{O}$, i.e., $\mathcal{O} \models \beta$. Since under Definition 4.2.1, $\mathcal{O} \not\models \psi$, such a β will not contribute to the explanation of ψ required by abduction, and should be omitted from $\mathcal{H}_F$ to satisfy Definition 4.2.1 condition (iii).

As a direct result of Theorems 3.5.2 and 6.4.1, the full abduction approach is also sound with respect to the ABox abduction problem in Definition 6.2.3 in $\mathcal{ALC}$ with observations as sets of concept and role assertions. As in Chapter 5, the full abduction approach is assumed to follow Steps (1)–(4) of Figure 5.2, where the filtering procedure uses the annotation-based approximation prior to full entailment checking due to the infeasibility of performing entailment checks over the entire forgetting solution.

6.5 Experimental Evaluation

To perform an evaluation of the approach, a prototype was implemented in Java using the OWL-API[1]. This prototype extends the prototype used in the experimental evaluation in Chapter 5, where the necessary modifications to each step, such as the required forms of negation and the option to use FAME during the forgetting step, in Figure 5.1 are included. One of the primary aims of this investigation was to compare the use of the semantic forgetting system FAME against the uniform interpolation system LETHE in Step (2) of Figure 5.1. Therefore, the prototype can be configured to use either system during this step, where the corresponding procedure is followed for either case. The results in this section can therefore be interpreted as a direct comparison between the approaches in Chapters 5 and 6 respectively.

As discussed in previous chapters, there are few experimental evaluations of abductive reasoning over DL ontologies, particularly over large, expressive ontologies such as those uploaded to NCBO Bioportal. In addition, the approaches in this thesis solve a particular, challenging instantiation of the abduction problem: computing semantically minimal spaces of independent explanations. Therefore, the approach taken in these experiments was to utilise the experimental framework outlined in Section 5.6 as a benchmark. The observations generated by this approach do not violate the conditions in Definition 4.2.1, i.e., they are consistent with the corresponding background ontology, but are also not entailed by it. In each case, this was checked using the external reasoner HermiT [GHM$^+$14] during the generation of each observation. In addition, the approach attempts to emulate the form and complexity of observations

[1]http://owlapi.sourceforge.net/

that may be observed in real application scenarios. While it is not possible to know exactly what forms the observations may take outside of case studies, it is important to try to emulate information that may be seen in practice, rather than generating arbitrarily simple or complex cases. The framework for generating observations was also extended. Specifically, the restriction placed on the observation was lifted and generated observations could include role assertions as well as concept assertions. This was used to evaluate the performance of the abduction approach in cases where FAME was used for the forgetting step, while the observation generation procedure from Section 5.6 was utilised when comparing this to the use of LETHE for forgetting.

First Experiment: Setup

The first experiment aims to compare the performance of the abduction system when using FAME in Step (2) of Figure 5.1 as opposed to LETHE. In both cases, the *full filtering* approach which uses the annotation-based approximation, followed by entailment checking, was used. For the entailment checks, the DL reasoner HermiT was used as during the observation generation procedure. The metrics used for the comparison of the two abduction approaches were the time taken to compute a hypothesis and the characteristics of the hypotheses obtained. To ensure that the performance of the two approaches was directly comparable, the set of observations was restricted to those that can be handled by the abduction system using LETHE as in Chapter 5. These included any $\mathcal{ALC}$ concept assertion, with at least one concept symbol that is not $\top$ or $\bot$, over a single individual. For ontologies with an ABox the individuals used in the observations were existing individuals from the ABox, while for those ontologies that did not have an ABox the individual was a freshly created one. The restriction to one individual was due to the fact that disjunctive assertions over multiple individuals cannot be expressed using the OWL API, which was used to provide input to the prototype for this experiment as in Section 5.6. For each observation, the forgetting signature $\mathcal{F}$ was set to one random concept symbol in the observation ψ. In this way, the results are indicative of a single step of the abduction procedure, where the assumption is that the user has no additional information that would enable an informed reduction in the size of the abducible set $\mathcal{S}_A$. Thus, the hypothesis obtained is the weakest possible hypothesis (least assumptive) for the given observation. It is assumed that the user would proceed to further refine the hypothesis by forgetting symbols from the hypotheses obtained. The time limit in this experiment was 300 seconds for both the forgetting and filtering steps respectively.

Ontology Name	DL	TBox Size	ABox Size	Num. Concepts	Num. Roles
BFO	$\mathcal{EL}$	52	0	35	0
LUBM	$\mathcal{EL}$	87	0	44	24
HOM	$\mathcal{EL}$	83	0	66	0
DOID	$\mathcal{EL}$	7892	0	11663	15
SYN	$\mathcal{EL}$	15352	0	14462	0
ICF	$\mathcal{ALC}$	1910	6597	1597	41
Semintec	$\mathcal{ALC}$	199	65189	61	16
OBI	$\mathcal{ALC}$	28888	196	3691	67
NATPRO	$\mathcal{ALC}$	68565	42763	9464	12

Table 6.1: Characteristics of the experimental corpus.

The corpus used in experiment 1 is the same as the one used in Chapter 5, which consists of ontologies from NCBO Bioportal[2], OBO Foundry[3], the LUBM benchmark [GPH05] and the Semintec[4] financial ontology. The choice of corpus is detailed in [DS19a]. The statistics of this corpus are shown in Table 6.1.

Second Experiment: Setup

The second experiment focused on assessing the performance of the abduction approach detailed in this Chapter, using both the approximate and full filtering settings. As before, these settings correspond to the case where the resulting hypothesis is the negation of the approximately reduced forgetting solution $\mathcal{V}^*_{app}$, which is not guaranteed to fully satisfy Definition 4.2.1, and the fully reduced forgetting solution $\mathcal{V}^*$, which is guaranteed to fully satisfy Definition 4.2.1, respectively. The corpus used

[2]https://bioportal.bioontology.org/
[3]http://www.obofoundry.org/
[4]http://www.cs.put.poznan.pl/alawrynowicz/semintec.htm

Number of	Mean	Median	90th Percentile	Maximum
TBox Axioms	1374	328	3830	8535
ABox Assertions	1014	26	2472	10889
Concepts	783	221	2232	6446
Roles	54	21	76	1043
Individuals	558	23	1605	8220

Table 6.2: Characteristics of the experimental corpus used in experiment 2.

in the second experiment was extracted from a snapshot of NCBO Bioportal [MP17]. The observations were generated in the same way as in experiment 1, but without the restrictions on the observations that were required for the comparison with the system in Chapter 5. The forgetting signature in each case included at least one symbol from the observation, including role symbols. As before the choice of forgetting signature for this experiment is based upon the fact that, while forgetting aims to restrict a background ontology to a portion of the original, the aim of the abduction problem is instead to produce a space of independent explanations that does not make too many assumptions about the new observations without sufficient prior knowledge. As a result, the assumption is that the forgetting signature sizes used in practice for the abduction problem are likely to be small, particularly when little is known about the observations in question. For the second experiment, the timeout for the method was set to 1000 seconds in total. The success rates reported include cases for which FAME failed to forget at least one symbol due to the incompleteness of the forgetting calculus utilised by the algorithm and cases for which the abduction approach, including both forgetting and filtering, exceeded the time limit.

For the second experiment, a larger corpus was used to evaluate the performance of the abduction approach in this chapter. The requirements of this second corpus were as follows. (1) For each ontology in the corpus, it must be possible to parse the ontology using OWL API, FAME and the reasoner HermiT. If there was an error in loading an ontology into any of these systems, it was excluded from the corpus. (2) The maximum size of any given ontology was 100,000 axioms. A restriction such as this is required for practical purposes. The specific upper bound on the size is based upon prior experimental evaluations performed for the forgetting system FAME [Zha18], since this is utilised as part of Step (2) and therefore similar guidelines are applicable here. (3) When generating sets of observations, an upper limit of 2,000 attempts was permitted before the process was terminated. If it was not possible to generate the required number of observations within this limit, then the corresponding ontology was excluded from the corpus. In this way, ontologies for which it was not possible to generate a sufficient number of non-entailed, consistent observations were excluded. (4) Every ontology in the corpus must have a non-empty ABox. One of the main benefits of the abduction approach in this Chapter is the fact that additional hypotheses can be reached. The ability to handle observations and hypotheses that contain role assertions enables the use of information about existing individuals in the ABox and their relationships to one another. Therefore, the most appropriate setting to evaluate the approach is in the presence of an ABox. Statistics for the final corpus, which contained 50 ontologies in total, are provided in Table 6.2.

All ontologies were preprocessed into their $\mathcal{ALC}$ fragments, since this is the setting of this work. Axioms not representable in $\mathcal{ALC}$ were removed, while those that are representable in $\mathcal{ALC}$ were translated using simple conversions.

Both experiments were performed on a machine using a 2.8GHz Intel Core i7-7700HQ CPU and 12GB of RAM.

6.5.1 Results

Since fixpoint operators are not utilised in the implementation of FAME, these were not present in the results. Thus, cases requiring fixpoints are deemed to be a failure case in the first experiment when comparing to the approach in Chapter 5, and count against the reported success rates for computing $\mathcal{H}_F$. However, these are unlikely to have a significant impact as they are rare in practice, as illustrated by the experimental results in Chapter 5 Section 5.6.

Ont.	Mean Time /s		Max Time/s		Mean Disjuncts		$\mathcal{O},\mathcal{H}_L \equiv$	Success %	
Name	$\mathcal{H}_L$	$\mathcal{H}_F$	$\mathcal{H}_L$	$\mathcal{H}_F$	$\mathcal{H}_L$	$\mathcal{H}_F$	$\mathcal{O},\mathcal{H}_F\%$	$\mathcal{H}_L$	$\mathcal{H}_F$
BFO	0.05	0.04	0.64	0.26	1.73	1.73	100.0	100.0	100.0
LUBM	0.08	0.06	0.67	0.30	2.53	2.96	60.8	100.0	86.7
HOM	0.06	0.05	0.65	0.26	2.5	2.5	100.0	100.0	100.0
DOID	3.35	3.07	9.97	10.26	4.77	4.77	100.0	100.0	100.0
SYN	6.18	2.84	16.12	13.92	5.6	5.6	100.0	100.0	100.0
ICF	0.96	0.67	3.56	2.16	1.93	1.93	100.0	100.0	100.0
Sem.	2.89	3.09	6.70	6.39	1.10	1.63	58.3	96.7	100.0
OBI	34.47	32.97	120.05	108.85	43.45	42.2	91.3	96.7	100.0
NAT.	46.04	138.24	301.27	688.87	10.61	4.17	62.5	76.7	76.7

Table 6.3: Results for the first experiment. $\mathcal{H}_L$ ($\mathcal{H}_F$) indicates results for the abduction approach using LETHE (FAME) in the forgetting step. The time limit for forgetting and filtering was 300 seconds each. For the equivalence check, only cases where both LETHE and FAME computed a hypothesis were compared. For the success rate, failures took into account times exceeding the timeout and, in the case of FAME, results for which the concept could not be forgotten and results containing definer symbols.

First Experiment: Results

The results for the first experiment are shown in Table 6.3. Over most of the ontologies in the corpus the mean runtime was shorter for the abduction approach in this chapter, which utilises FAME in Step (2). Two exceptions to this general trend were the Semintec and NATPRO ontologies: the runtime using the abduction approach in this chapter was longer in a few cases, particularly over the NATPRO ontology, for which the mean runtime was over double that of the approach in Chapter 5. These differences could be due to the computation of additional explanations resulting from the expressivity of the forgetting solution obtained using FAME in Step (2), which would necessitate additional entailment checks during the filtering process in Step (3). Also, for ontologies with large ABoxes, a significant number of axioms need to be transformed to TBox axioms using nominals, which may increase the time taken. In most cases, the success rate when computing $\mathcal{H}_L$ was 100%. The same is true for $\mathcal{H}_F$. In the former case, using the abduction approach of Chapter 5, failures occurred over the larger and more expressive ontologies, Semintec, OBI and NATPRO. These are due to timeouts during Step (2), indicating that LETHE took longer than 300 seconds to produce a solution. When utilising FAME in Step (2), failures can occur due to the incompleteness of FAME's calculus: all of the failures over the LUBM ontology when computing $\mathcal{H}_F$ were due to this characteristic, while for the NATPRO ontology all of the failure cases were instead due to timeouts. In most cases, the hypotheses $\mathcal{H}_L$ and $\mathcal{H}_F$ were equivalent under the corresponding ontology. This indicates that it should often be possible to express $\mathcal{H}_F$ in $\mathcal{ALC}$, which may help to improve the readability issue discussed in Example 6.3.1 in these cases. Over the LUBM, Semintec, OBI and NATPRO ontologies, a number of the $\mathcal{H}_F$ hypotheses were weaker than the corresponding $\mathcal{H}_L$. This is expected, due to the fact that the forgetting result obtained in Step (2) when utilising FAME may be stronger than the corresponding forgetting solution produced by LETHE. In some cases there may be hypotheses that cannot be expressed without the extra expressivity of the forgetting solution computed using FAME. Example 6.5.1 is taken from the LUBM experiments and demonstrates the benefit in practice.

Example 6.5.1. *For the observation $\psi = \neg Organization(a)$, where a is a fresh individual, the key axioms in the LUBM ontology were:*

$$Person \sqcap \exists worksFor.Organization \sqsubseteq Employee \quad College \sqsubseteq Organization$$
$$Employee \sqsubseteq Person \sqcap \exists worksFor.Organization$$

For the forgetting signature $\mathcal{F} = \{Organization\}$, the hypothesis was:

$$\mathcal{H}_F = \top \sqsubseteq \forall\nabla.(\neg\{a\} \sqcup \exists worksFor^-.(\neg Employee \sqcap Person))$$

Other explanations, such as those equivalent to $\neg College(a)$, are redundant with respect to Definition 1(iii) and are removed by the filtering step. Using the approach in Chapter 5, no hypothesis was produced as the above hypothesis requires the use of the inverse role $worksFor^-$, which cannot be produced using LETHE in Step (2).

Second Experiment: Results

The results for experiment 2 are shown in Table 6.4. As expected, the approximate filtering took less time than the full filtering across all cases, as it does not perform

$\mathcal{F}$	Forgetting Time		Approx. Filter Time		Full Filter Time		Successes %	
Size	Mean	Max	Mean	Max	Mean	Max	Approx.	Full
1	0.05	1.02	0.74	869.63	7.40	880.11	90.3	89.3
5%	0.13	11.15	0.09	28.25	8.29	878.05	81.7	80.9
10%	1.04	75.09	0.06	5.52	6.45	975.24	70.9	70.6

Table 6.4: Results for experiment 2. Percentages for $\mathcal{F}$ are relative to $sig(\mathcal{O}, \psi)$. All times are in seconds.

the additional, expensive entailment checks. The maximum time for the approximate filtering for an $\mathcal{F}$ size of 1 is particularly high. It is likely that for this single case the forgetting solution was particularly large, indicating that the forgotten symbol occurred frequently in the given ontology. The mean number of redundant axioms removed from the forgetting results by the approximate filtering was 2444.6, 2510.4 and 2873.3 for $\mathcal{F}$ sizes of 1, 5% and 10% respectively. The mean additional redundancies removed by the full filtering setting was 11.7, 11.3 and 9.7 axioms respectively. This indicates that in many cases the approximate filtering may be sufficient to obtain a space of explanations that is largely free of redundancies. The success rates indicate that the full filtering setting caused a number of additional timeouts for each size of $\mathcal{F}$. However, the majority of failures were the result of FAME failing to forget at least one symbol in $\mathcal{F}$. For the approximate filtering cases, 100%, 100% and 99.5% of failures occurred due to the forgetting step for $\mathcal{F}$ sizes 1, 5% and 10% respectively. For the full filtering cases, the corresponding values were 88.8%, 94.8% and 94.8% respectively. FAME's failure rates for these abduction experiments are higher than those reported for forgetting experiments [ZS16, Zha18]. This may be due to the frequency of role symbols occurring in ABox observations for abduction, many of which included role assertions or complex concepts involving roles.

Chapter 7

TBox and Knowledge Base Abduction

In this Chapter, forgetting-based abduction is investigated for the problem of TBox abduction, for which the observations and hypotheses can be TBox axioms, i.e., universally quantified axioms rather than ground axioms as in the ABox abduction problem. First, the general problem of computing least assumptive spaces of independent explanations is investigated in the context of TBox abduction from two perspectives. The first adheres to standard considerations of TBoxes in DLs, examining disjunctive redundancy occurring inside a single TBox axiom. The second aims to more fully capture the notion of independent explanations by extending the setting to include disjunctions of TBoxes. The need to extend the forgetting calculus used in Chapter 5, to overcome both shortcomings of the approach and the TBox abduction problem, are then examined. Following this, the knowledge base (KB) abduction problem is defined, which generalises the problems of ABox and TBox abduction. To solve this problem, extensions that have been made to the calculus used by the forgetting system LETHE, as well as the techniques to avoid redundant inferences with respect to abduction, are presented. The result is the first approach for KB abduction over $\mathcal{ALC}$ ontologies, that also extends the space of explanations that can be reached when compared to previous chapters. The resulting system is then evaluated experimentally over a corpus of ontologies, focusing on the effect of increasing the number of non-abducibles and the size of the observation with respect to ABox, TBox and KB observations. The aim is to avoid, where possible, strong assumptions on the forms of observations that may be encountered in practice.

A portion of the work in this chapter is based on a collaboration with Patrick Koopmann, Sophie Tourret and Renate A. Schmidt that was published in [KDTS20], particularly Sections 7.3, 7.4 and the Appendix A.1. The experimental setup and results presented in Section 7.6 are different to those in the aforementioned work.

7.1 Independent Explanations and TBox Abduction

As discussed in previous chapters, forgetting-based approaches to abduction lend themselves well to abduction problems where the aim is to produce semantically minimal hypotheses due to the duality between strongest necessary and weakest sufficient conditions. However, unlike most considerations of semantic minimality in DLs [KES11, HB12, HBK14] the setting of the problem considered in this thesis does not exclude disjunctions in the hypothesis. Since the approach relies on both contraposition and forgetting, and the forgetting solution is a conjunction of entailments of the input, the resulting hypothesis will be a disjunction. As a result, an additional requirement must be met: inter-disjunct redundancy as specified in Definition 4.2.1 condition (iii). As argued in Chapter 4, this ensures that the semantically minimal hypothesis computed does not consist mostly of redundant explanations, i.e., that each disjunct is an independent explanation for the new observation. This requirement is conceptually the same in the TBox abduction setting. However, the notion of disjunctive redundancy is less clear in the case of TBox abduction and fully solving the problem requires further extension.

Consider the setting of the abduction problem in Chapter 5: abduction in the DL $\mathcal{ALC}$. An example of TBox abduction in this setting is as follows:

Example 7.1.1. *Consider the following instance of the abduction problem:*

$$\mathcal{O} = \{B \sqsubseteq C,$$

$$\begin{aligned}
&\forall r.A \sqsubseteq C, \\
&E \sqsubseteq C, \\
&E \sqcap D \sqsubseteq \bot\} \\
\psi =& D \sqsubseteq C \\
\mathcal{S}_A =& \{r, A, B, D\}
\end{aligned}$$

A possible hypothesis is as follows:

$$\mathcal{H}_1 = D \sqsubseteq (\forall r.A) \sqcup B$$

In order to check whether or not a candidate TBox abduction hypothesis satisfies the conditions in Definition 4.2.1, it is first necessary to establish the notion of a space of explanations in the TBox abduction setting. This requires an appropriate way to account for disjunctions of explanations in a TBox hypothesis. In this section, two possible characterisations of disjunctive redundancy will be explored: *intra-axiom* and *inter-axiom*. The former examines disjunctions in hypotheses that take the form of TBox axioms, as encountered commonly in DLs, while the latter case requires an extension of the standard consideration of TBoxes to the notion of a disjunction of TBoxes.

7.1.1 Intra-Axiom Disjunctive Redundancy

For Example 7.1.1, only one axiom is required in the hypothesis $\mathcal{H}_1$: $D \sqsubseteq (\forall r.A) \sqcup B$. However, the notion of disjunctions of explanations is still relevant here. Consider the following hypotheses as alternative solutions to Example 7.1.1:

$$\begin{aligned}
\mathcal{H}_2 &= D \sqsubseteq \forall r.A \\
\mathcal{H}_3 &= D \sqsubseteq B \\
\mathcal{H}_4 &= D \sqsubseteq (\forall r.A) \sqcup B \sqcup E
\end{aligned}$$

All three of these hypotheses are consistent explanations for ψ under $\mathcal{O}$. Thus, they satisfy the notions in Definition 4.2.1 conditions (i) and (ii). It is worth noting that the superclass of both $\mathcal{H}_2$ and $\mathcal{H}_3$ consists of one of the disjuncts from the superclass in $\mathcal{H}_1$. Consequently, both $\mathcal{H}_2$ and $\mathcal{H}_3$ are less semantically minimal than $\mathcal{H}_1$, since $\mathcal{O},\mathcal{H}_2 \models \mathcal{H}_1$ and $\mathcal{O},\mathcal{H}_3 \models \mathcal{H}_1$ but $\mathcal{O},\mathcal{H}_1 \not\models \mathcal{H}_2$ and $\mathcal{O},\mathcal{H}_1 \not\models \mathcal{H}_3$. Meanwhile, without considering redundancy of individual disjuncts, $\mathcal{H}_4$ is as semantically minimal as $\mathcal{H}_1$ since both entail one another under the background ontology $\mathcal{O}$.

Capturing the notion of inter-disjunct redundancy in Definition 4.2.1 condition (iii) requires an appropriate consideration of disjunction in the TBox abduction setting. Neither $\mathcal{H}_2$ nor $\mathcal{H}_3$ are interesting cases here, since neither contain a disjunction and thus trivially satisfy the notion of condition (iii). This leaves $\mathcal{H}_4$. If the hypothesis is still seen as being a disjunction of the following form:

$$\mathcal{H} = \alpha_1 \vee ... \vee \alpha_n$$

where each α_i is a conjunction of DL axioms, then $\mathcal{H}_4$ also trivially passes condition (iii) since it is a single axiom.

However, it is clear that $\mathcal{H}_4$ intuitively contains a form of disjunctive redundancy: the disjunct E in the superclass is redundant, since $D \sqsubseteq E$ is contradictory under $\mathcal{O}$ since D and E are disjoint. Consider the following general concept inclusions (GCIs):

$$D \sqsubseteq \forall r.A$$
$$D \sqsubseteq B$$
$$D \sqsubseteq E$$

consisting of D included in each of the disjunctions in the superclass of $\mathcal{H}_4$. If each of these is seen as a candidate explanation for ψ, then the inter-disjunct redundancy condition can be checked as follows:

$$\mathcal{O}, D \sqsubseteq \forall r.A \models D \sqsubseteq B \sqcup E$$

$$\mathcal{O}, D \sqsubseteq B \models D \sqsubseteq (\forall r.A) \sqcup E$$
$$\mathcal{O}, D \sqsubseteq E \models D \sqsubseteq (\forall r.A) \sqcup B$$

where each "disjunct" being checked is now a GCI. The first two entailment checks return false. Therefore, the explanations $D \sqsubseteq \forall r.A$ and $D \sqsubseteq B$ are not redundant with respect to one of the other explanations for the observation ψ. The third axiom $D \sqsubseteq E$ is inconsistent with the background ontology, i.e., $\mathcal{O}, D \sqsubseteq E \models \bot$, and as such the third entailment check trivially holds. Thus, the disjunct "E" in the superclass can be seen as redundant: it does not provide a consistent (and independent) explanation for the observation ψ. Removing this disjunct from the superclass leaves the hypothesis originally suggested in Example 7.1.1:

$$\mathcal{H} = D \sqsubseteq (\forall r.A) \sqcup B$$

as the preferred solution to the abduction problem. This solution comes closer to satisfying the notion of a set of independent explanations, since it excludes the redundant explanation $D \sqsubseteq E$.

Note that disjunctions, and thus inter-disjunct redundancy, can occur in both the superclass and subclass of TBox axioms, as illustrated by the following example.

Example 7.1.2. *Consider the following abduction problem:*

$$\begin{aligned}
\mathcal{O} =& \{D \sqsubseteq A \\
& D \sqsubseteq B \\
& D \sqsubseteq E \\
& C(a) \\
& \neg E(a)\} \\
\psi =& D \sqsubseteq C \\
\mathcal{S}_A =& \{A, B, C, E\}
\end{aligned}$$

The following is a hypothesis for the above problem:

$$\mathcal{H}_1 = A \sqcap B \sqsubseteq C$$

In the above hypothesis $\mathcal{H}_1$, the conjunction in the subclass should be treated as a disjunction in the same way as an explicit disjunction in the superclass, i.e., both $A \sqsubseteq C$ and $B \sqsubseteq C$ are stronger explanations for ψ than $\mathcal{H}_1$. For example, the following candidate hypothesis for Example 7.1.2 contains an intra-axiom disjunctive redundancy:

$$\mathcal{H}_2 = A \sqcap B \sqcap E \sqsubseteq C$$

in the form of the "E" in the subclass, since $E \sqsubseteq C$ is inconsistent with the background $\mathcal{O}$ given the information about the individual a.

Given this characterisation, one possible realisation of the abduction problem in Definition 4.2.1 in the context of TBox abduction in $\mathcal{ALC}$ is as follows:

Definition 7.1.1. *Let $\mathcal{O}$ be an $\mathcal{ALC}$ ontology and ψ be an observation as set of $\mathcal{ALC}$ TBox axioms, where $\mathcal{O}, \psi \not\models \perp$ and $\mathcal{O} \not\models \psi$. Let $\mathcal{S}_A$ be a set of symbols, called abducible symbols, where $\mathcal{S}_A \subseteq sig(\mathcal{O}, \psi)$. The TBox abduction problem is to find an $\mathcal{ALC}$ hypothesis $\mathcal{H}$ as a set of TBox axioms such that:*

(i) $\mathcal{O}, \mathcal{H} \not\models \perp$

(ii) $\mathcal{O}, \mathcal{H} \models \psi$

(iii) Let $\mathcal{H} = \gamma_1 \sqcap \ldots \sqcap \gamma_m \sqsubseteq \alpha_1 \sqcup \ldots \sqcup \alpha_n$. The following conditions must hold where C and D represent the sub and superclass of $\mathcal{H}$ respectively:

(a) *For each $\alpha_i \in D$, it is not the case that $\mathcal{O}, C \sqsubseteq \alpha_i \models C \sqsubseteq \bigsqcup_{j \neq i}^{n} \alpha_j$.*

(b) *For each $\gamma_i \in C$, it is not the case that $\mathcal{O}, \gamma_i \sqsubseteq D \models \bigsqcap_{j \neq i}^{m} \gamma_j \sqsubseteq D$.*

(iv) If there exists a $\mathcal{H}'$ satisfying (i)–(iii), where $sig(\mathcal{H}') \subseteq \mathcal{S}_A$, then $\mathcal{O}, \mathcal{H}' \models \mathcal{H}$.

This definition assumes that the hypothesis $\mathcal{H}$ takes the form of a single TBox axiom, as permitted in DLs. Note that a conjunction of GCIs can also be represented in the form of a single TBox axiom. The aim is then to reduce redundancy between disjunctions that can occur within the axiom. However, this definition is not sufficient to fully capture the notion of a hypothesis in Definition 4.2.1. Particularly, it assumes that the semantically minimal hypothesis $\mathcal{H}_T$ can be represented as a single GCI.

7.1.2 Inter-Axiom Disjunctive Redundancy

The main assumption made in Definition 7.1.1 is limiting for the abduction setting considered in this thesis, as illustrated by the following example:

Example 7.1.3. *Consider the following TBox abduction problem:*

$$\mathcal{O} = \{\emptyset\}$$
$$\psi = \exists r.A \sqcap A \sqsubseteq \exists r.B \sqcup D$$

Two possible explanations for ψ under $\mathcal{O}$ are:

$$A \sqsubseteq B$$
$$A \sqsubseteq D$$

To obtain the semantically minimal space of independent explanations, the optimal hypothesis would be:

$$\mathcal{H} = A \sqsubseteq B \vee A \sqsubseteq D$$

The hypothesis $\mathcal{H}$ in Example 7.1.3 satisfies the properties of absence of inter-disjunct redundancy and semantic minimality. However, the required disjunction cannot be expressed using a single GCI, and is thus not directly representable in DLs such as $\mathcal{ALC}$. This is clearer when considering the first-order translations of DL statements as expressed in Figure 7.1 [HSG04].

DL Concepts:
$\pi(C,x) = C(x)$
$\pi(\nabla,x,y) = \top$
$\pi(C \sqcap D,x) = \pi(C,x) \wedge \pi(D,x)$
$\pi(C \sqcup D,x) = \pi(C,x) \vee \pi(D,x)$
$\pi(\forall r.C,x) = \forall y(\pi(r,x,y) \rightarrow \pi(C,y))$

General Concept Inclusion Axiom:
$\Pi(C \sqsubseteq D) = \forall x(\pi(C,x) \rightarrow \pi(D,x))$

Figure 7.1: Standard translations from description logics to first-order logics. [HSG04]

Consider the following two statements with respect to Example 7.1.3:

$$(1)\, A \sqsubseteq B \sqcup D$$
$$(2)\, A \sqsubseteq B \vee A \sqsubseteq D$$

where the first statement attempts to capture the required explanations in a single TBox axiom, while the second statement is the suggested optimal hypothesis for the abduction problem given in Example 7.1.3. The first-order translations of these two statements proceeds as follows:

$$\Pi(A \sqsubseteq B \sqcup D) = \forall x(\pi(A,x) \longrightarrow \pi(B,x) \vee \pi(D,x))$$
$$\Pi(A \sqsubseteq B \vee A \sqsubseteq D) = \forall x_1(\pi(A,x_1) \longrightarrow \pi(B,x_1)) \vee \forall x_2(\pi(A,x_2) \longrightarrow \pi(D,x_2))$$

which then becomes:

$$(1)\, \forall x(A(x) \longrightarrow B(x) \vee D(x))$$
$$(2)\, \forall x_1(A(x_1) \longrightarrow B(x_1)) \vee \forall x_2(A(x_2) \vee D(x_2))$$

note that, since universal quantification does not distribute over disjunction, these two are not equivalent. In fact, statement (1) is not a valid explanation for ψ in Example 7.1.3. It is in fact too weak: if statement (1) is taken to be a hypothesis $\mathcal{H}$ then when it is added to the background ontology $\mathcal{O}$, which in this case is empty, the requirement $\mathcal{O},\mathcal{H} \models \psi$ is not satisfied. This is clearer when considering possible interpretations satisfying each statement: an interpretation where some elements of A are elements of B but not D, while the rest are are elements of D but not B, satisfies statement (1) but not statement (2) nor the observation ψ. For example, the following ABox:

$$
\begin{array}{rr}
A(ind1) & A(ind2) \\
B(ind1) & \neg B(ind2) \\
\neg D(ind1) & D(ind2) \\
r(ind1, ind2) &
\end{array}
$$

is a model of statement (1): $A \sqsubseteq B \sqcup D$. However, the question remains: does this statement provide an explanation for the TBox observation ψ in Example 7.1.3. Examining the information regarding the individual $ind1$: it is an element of A and it has an r-successor that is an element of A, namely $ind2$. Thus, under ψ it should be the case that $ind1$ either has an r-successor that is an element of B, or $ind1$ is an element of D. Neither of these hold, and so $\mathcal{O},\mathcal{H} \not\models \psi$ where $\mathcal{O} = \{\emptyset\}$, i.e., statement (1) cannot be a hypothesis for ψ.

The solution to the abduction problem in Example 7.1.3 is instead statement (2): a disjunction of GCIs. Examining the two disjuncts in statement (2), both statements would be consistent explanations for the observation ψ under $\mathcal{O}$:

$$\mathcal{O}, A \sqsubseteq B \models \exists r.A \sqcap A \sqsubseteq \exists r.B \sqcup D$$
$$\mathcal{O}, A \sqsubseteq D \models \exists r.A \sqcap A \sqsubseteq \exists r.B \sqcup D$$

It is also the case that neither $\mathcal{O}, A \sqsubseteq B \models A \sqsubseteq D$ nor $\mathcal{O}, A \sqsubseteq D \models A \sqsubseteq B$ hold, meaning that neither disjunct is stronger than the other under the background ontology. As a result, the hypothesis satisfies the inter-disjunct redundancy notion in Definition 4.2.1 condition (iii). Generalising from this leads to a second definition for the problem of computing the weakest space of $\mathcal{ALC}$ TBox hypotheses in the DL $\mathcal{ALC}$, as follows.

Definition 7.1.2. ***TBox Abduction in $\mathcal{ALC}$ Ontologies.*** *Let $\mathcal{O}$ be an ontology and ψ be a set of TBox axioms, both expressed in $\mathcal{ALC}$, such that $\mathcal{O}, \psi \not\models \bot$ and $\mathcal{O} \not\models \psi$. Let $\mathcal{S}_A$ be a set of symbols called abducibles. The* TBox abduction problem *is to compute a hypothesis $\mathcal{H} = \alpha_1 \vee ... \vee \alpha_n$ as a disjunction of TBox axioms, where each α_i is equivalent to a GCI of the form $C_i \sqsubseteq D_i$ where C_i, D_i are arbitrary $\mathcal{ALC}$ concepts. The solution $\mathcal{H}$ must contain only those symbols specified in $\mathcal{S}_A$ and satisfy the following conditions:*

(i) $\mathcal{O}, \mathcal{H} \not\models \bot$

(ii) $\mathcal{O}, \mathcal{H} \models \psi$,

(iii) $\mathcal{H}$ does not contain inter-disjunct *redundancy i.e., there is no disjunct α_i in $\mathcal{H}$ such that $\mathcal{O}, \alpha_i \models \alpha_1 \vee ... \vee \alpha_{i-1} \vee \alpha_{i+1} \vee ... \vee \alpha_n$*

(iv) for any $\mathcal{H}'$ satisfying conditions (i)–(iii) where $sig(\mathcal{H}') \subseteq \mathcal{S}_A$, then $\mathcal{O}, \mathcal{H}' \models \mathcal{H}$.

By allowing disjunctions of TBox axioms, this problem takes the same form as the ABox abduction problem presented in Definition 5.1.1: assuming that each disjunct α_i is a TBox axiom rather than a concept assertion, the conditions (i) – (iv) can be treated the same way. Note that in $\mathcal{ALC}$, a conjunctive set of TBox axioms (a TBox) can be represented as a single TBox axiom and thus each α_i is not restricted in this sense.

As discussed, it is not possible to represent solutions as disjunctions of GCIs directly in $\mathcal{ALC}$. However, by utilising the universal role ∇, it is possible to represent a disjunction of GCIs in DLs such as those utilised in Chapter 6. For example, the hypothesis in Example 7.1.3 can be represented equivalently as follows:

$$\top \sqsubseteq \forall\nabla.(\neg A \sqcup B) \sqcup \forall\nabla.(\neg A \sqcup D)$$

Generalising from this, any given hypothesis as a disjunction of GCIs can be represented using the above form, as formalised in Lemma 7.1.1.

Lemma 7.1.1. *A hypothesis $\mathcal{H}$ that takes the form of a disjunction of GCI axioms $\mathcal{H} = C_1 \sqsubseteq D_1 \vee ... \vee C_n \sqsubseteq D_n$, where each C_i, D_i is an $\mathcal{ALC}$ concept, can be equivalently represented as a single axiom of the following form:*

$$\mathcal{H} = \top \sqsubseteq \forall\nabla.(\alpha_1) \sqcup ... \sqcup \forall\nabla.(\alpha_n)$$

where each α_i is an $\mathcal{ALC}$ concept.

Proof: Given a disjunction of arbitrary GCIs of the form $\mathcal{H} = C_1 \sqsubseteq D_1 \vee ... \vee C_n \sqsubseteq D_n$ where each C_i, D_i is an $\mathcal{ALC}$ concept, the suggested equivalent representation is:

$$\mathcal{H} = \top \sqsubseteq \forall\nabla.(\neg C_1 \sqcup D_1) \sqcup ... \sqcup \forall\nabla.(\neg C_n \sqcup D_n)$$

i.e., each α_i in the lemma is a clause of the form $\neg C_i \sqcup D_i$. The first-order translation of the above statement, using the notation in Figure 7.1 [HSG04] proceeds as follows:

$$\begin{aligned}
&\Pi(\mathcal{H}) = \Pi[\top \sqsubseteq \forall\nabla.(\neg C_1 \sqcup D_1) \sqcup ... \sqcup \forall\nabla.(\neg C_n \sqcup D_n)] \\
&= \forall x[\pi(\top, x) \rightarrow \pi(\forall\nabla.(\neg C_1 \sqcup D_1) \sqcup ... \sqcup \forall\nabla.(\neg C_n \sqcup D_n), x)] \\
&= \forall x[\pi(\top, x) \rightarrow \pi(\forall\nabla.(\neg C_1 \sqcup D_1)) \vee ... \vee \pi(\forall\nabla.(\neg C_n \sqcup D_n))] \\
&= \forall x[\top \rightarrow \forall y_1(\pi(\nabla, x, y_1) \rightarrow \pi(\neg C_1 \sqcup D_1, y_1)) \vee ... \vee \forall y_n(\pi(\nabla, x, y_n) \rightarrow \pi(\neg C_n \sqcup D_n, y_n))] \\
&= \forall x[\top \rightarrow \forall y_1(\top \rightarrow \pi(\neg C_1, y_1 \vee \pi(D_1, y_1)) \vee ... \vee \forall y_n(\top \rightarrow \pi(\neg C_n, y_n) \vee \pi(D_n, y_n))] \\
&= \forall x[\forall y_1(\neg C_1(y_1) \vee D_1(y_1)) \vee ... \vee \forall y_n(\neg C_n(y_n) \vee D_n(y_n))]
\end{aligned}$$

eliminating the superfluous universal quantifier then leaves:

$$\Pi(\mathcal{H}) = \forall y_1(\neg C_1(y_1) \vee D_1(y_1)) \vee ... \vee \forall y_n(\neg C_n(y_n) \vee D_n(y_n))$$

Now it remains to show that this is equivalent to a disjunction of GCIs. Each disjunct above corresponds to the FOL translation of a single GCI axiom as follows:

$$\begin{aligned}
&\forall y_i(\neg C_i(y_i) \vee D_i(y_i)) \\
=&\forall y_i(C_i(y_i) \rightarrow D_i(y_i)) \\
=&\Pi(C_i \sqsubseteq D_i)
\end{aligned}$$

by the reverse of the translation in Figure 7.1. Following this, by extending the translation to include an arbitrary number of disjuncts, it can be seen that the hypothesis $\mathcal{H}$ is equivalent to:

$$C_1 \sqsubseteq D_1 \vee ... \vee C_n \sqsubseteq D_n$$

which is a disjunction of GCIs where each C_i, D_i is an $\mathcal{ALC}$ concept as required.

Thus, to solve the TBox abduction problem presented in Definition 7.1.2 via a forgetting-based approach, it is necessary to utilise a calculus that can compute forgetting results in DLs that include the top role ∇.

It is reasonable to expect that the two most promising avenues of investigation are: (1) extending the forgetting calculus used by the system LETHE to handle TBox abduction and (2) utilising the calculus used by FAME directly to solve the TBox abduction problem in the same, or a similar, fashion to the ABox abduction approach in Chapter 6.

For the first option, tackling the TBox abduction problem in Definition 7.1.2 would indeed require the extension of the forgetting calculus used in the forgetting system LETHE. The limitation regarding computing negated role assertions, outlined in Chapter 5 is not problematic in the TBox abduction case since role assertions will not be present in the observation, and it is not possible for a role assertion to be provided as an explanation of a set of TBox axioms. However, the incompleteness of role forgetting with respect to the TBox abduction problem illustrated in Example 5.4.8 must be addressed, which is discussed in Section 7.2.

For the second option, the calculus utilised by the system FAME can already perform forgetting in DLs that utilise the top role ∇, unlike the calculus used by LETHE. As a result, it should be the case that the abduction problem in Example 5.4.8 can be solved using an abduction system that utilises FAME's calculus in Step (2) of Figure 5.1. Further, it is clear that the representation of $\mathcal{H}$ suggested in this section should not be problematic. In fact, it is already similar to the representation used in Chapter 6. The same can be said of $\neg\psi$, for example given a TBox observation $\psi = \{A \sqsubseteq B, C \sqsubseteq D\}$, the negation $\neg\psi$ is equivalent to:

$$\neg\psi = \top \sqsubseteq \exists\nabla.(A \sqcap \neg B) \sqcup \exists\nabla.(C \sqcap \neg D)$$

while in Chapter 6 for an ABox observation such as $\psi = \{A(c), B(d)\}$, $\neg\psi$ takes the form:

$$\neg\psi = \top \sqsubseteq \forall\nabla.(\neg\{c\} \sqcup \neg A) \sqcup \forall\nabla.(\neg\{d\} \sqcup \neg B)$$

Note that the quantifier applied to ∇ differs here compared to Chapter 6, since in the TBox abduction case the original observation ψ is universally quantified and the negation is existentially quantified while the opposite is true for ABox abduction. Aside from this, the two representations of $\neg\psi$ take similar forms, with the main difference being that no nominals are required in the TBox abduction setting if we assume that the background ontology $\mathcal{O}$ and the observation ψ are in $\mathcal{ALC}$. The same parallel exists between the representations of $\mathcal{H}$ in both cases.

The focus of the abduction approach in the following section will be on the first case: extending the forgetting calculus of LETHE, for which there are several motivations. First, the forgetting solutions produced via LETHE's calculus are in general syntactically simpler than those produced via FAME's Ackermann-based approach, as discussed in Chapter 6. Second, unlike FAME, LETHE utilises a calculus that is complete for the forgetting problem in $\mathcal{ALC}$. Thus, assuming that the incompleteness of role forgetting with respect to TBox abduction can be overcome, it is reasonable to assume that the extension could be complete with respect to TBox abduction.

7.2 Motivating Extensions to the $Int_{\mathcal{ALC}}$ Calculus

In this section, specific cases that require extensions to the forgetting calculus $Int_{\mathcal{ALC}}$, utilised in Step (2) of Figure 5.1 as discussed in Chapter 5, will be examined. These will be used to motivate directions for extending the calculus specifically to solve these problematic cases, primarily to enable the computation of solutions to TBox abduction problems and to overcome limitations in the ABox abduction approach of Chapter 5. Here the focus is on examining motivating cases based on the prior approach and identifying initial extensions that can be made to solve them. The final extensions made to the forgetting calculus, which are the subject of the collaboration in [KDTS20], will be discussed in Section 7.4.

For TBox abduction, Example 5.4.8 in Chapter 5 demonstrates a case requiring an extension of the $Int_{\mathcal{ALC}}$ calculus in [KS15b]:

Example 7.2.1. *Recall the following abduction problem:*

$$\begin{aligned} \mathcal{O} =& \{\emptyset\} \\ \psi =& \{\exists r.C \sqsubseteq \exists r.D\} \\ \mathcal{S}_A =& \{C, D\} \end{aligned}$$

Here the aim is to solve this problem using the $Int_{\mathcal{ALC}}$ calculus used in Step (2) of Figure 5.1, as discussed in Chapter 5. For a GCI axiom $C \sqsubseteq D$, the clausal form is $\neg C \sqcup D$, while the negation of this would therefore be $C \sqcap \neg D$. As a result, during Step (1) in the abduction approach in Figure 5.1, a possible representation for the negated observation is:

$$\neg\psi = (\exists r.C \sqcap \forall r.\neg D)(a^*)$$

where a^* is a fresh individual. The application of $Int_{\mathcal{ALC}}$ calculus to eliminate $\mathcal{F} = \{r\}$

proceeds as follows, where $\mathcal{O}, \neg\psi$ has been converted to the required clausal normal form:

1) $\exists r.D_1(a^*)$

2) $\neg D_1(x) \vee C(x)$

3) $\forall r.D_2(a^*)$

4) $\neg D_2(x) \vee \neg D(x)$

while the role propagation rule could be applied on clauses (1) and (3), this would not yield any further inferences on symbols in $\mathcal{F}$. Even if the role propagation rule was applied regardless, the result would be as follows:

5) $\exists r.D_{12}(a^*)$	*Role Propagation (1, 3)*
6) $\neg D_{12}(x) \vee D_1(x)$	
7) $\neg D_{12}(x) \vee D_2(x)$	
8) $\neg D_{12}(x) \vee C(x)$	*Resolution (2, 6)*
9) $\neg D_{12}(x) \vee \neg D(x)$	*Resolution (4, 7)*

At this point, following the elimination of definer symbols and clauses containing symbols in $\mathcal{F}$, the result would still be $\mathcal{V} = \{\emptyset\}$. Thus, in Step (3) of the abduction approach in Figure 5.1 the reduced uniform interpolant $\mathcal{V}^*$ is also empty and no final hypothesis is returned from the abduction approach. However, there is clearly a valid hypothesis within the signature $\mathcal{S}_A$ under Definition 7.1.2: $\mathcal{H} = C \sqsubseteq D$.

In the above example, the issue lies in the fact that it is not possible to retain the information regarding the definer D_{12}, which is discarded at the end of the forgetting process. For the forgetting problem in $\mathcal{ALC}$ [KS15b], this is sufficient: there is no entailment of $\mathcal{O}, \neg\psi$ in the signature $\mathcal{S}_A$ that is representable in $\mathcal{ALC}$ with disjunctive assertions. However, for the abduction problem the hypothesis $\mathcal{H} = C \sqsubseteq D$ is a valid solution. Thus, it is necessary to extend $Int_{\mathcal{ALC}}$ to account for this, i.e., retain the information regarding D_{12}. Utilising the top role ∇, this can be achieved as follows:

$$10)\neg D_1 \sqcup \exists \nabla . D_{12}$$

by effectively replacing the occurrence of r, which occurs in the forgetting signature $\mathcal{F}$, with ∇, the required entailment in $\mathcal{S}_A$ is retained. Substituting the definition for D_{12} back into (10), eliminating the definers and then extracting the reduced forgetting solution $\mathcal{V}^*$ gives:

$$\mathcal{V}^* = \{\exists \nabla . (C \sqcap \neg D)\}$$

which when negated results in the following hypothesis:

$$\mathcal{H} = \top \sqsubseteq \forall \nabla r . (\neg C \sqcup D)$$

which is equivalent to the expected hypothesis $\mathcal{H} = C \sqsubseteq D$.

Extending this to the general case, an initial suggestion for extending the calculus $Int_{\mathcal{ALC}}$ is to include a rule of the following form:

$$\frac{(C_1 \vee \forall r.D_1)(t_1) \qquad (C_2 \vee \exists r.D_2)(t_2)}{(C_1 \vee C_2)\sigma \vee \exists \nabla . D_{12}(t_1\sigma)}$$

where $Q \in \{\exists, \forall\}$ and $r \in \mathcal{F}$. This is a special case of the role propagation rule to retain the entailment under r when r is a symbol to be forgotten by utilising ∇. Applying this to Example 7.2.1, the derivation of clause (6) instead becomes:

$$6)\exists \nabla . D_{12}$$

as required. Since r is no longer present in this clause, the clause will be retained.

The final realisation of this notion will be discussed later in this Chapter in Section 7.3, by the introduction of additional rules to the $Int_{\mathcal{ALC}}$ calculus.

For ABox abduction, the main limitation discussed in Chapter 5 is the inability to support observations and hypotheses that contain role assertions of the form $r(a,b)$.

For example, a hypothesis containing a role assertion as an explanation occurs in the following problem:

Example 7.2.2. *Consider the following:*

$$\begin{aligned}\mathcal{O} =& \{\exists r.C \sqsubseteq A,\\ & B \sqsubseteq A,\\ & C(b)\}\\ \psi =& A(a)\\ \mathcal{S}_A =& \{r, B\}\end{aligned}$$

A possible hypothesis, as a weakest set of explanations, is:

$$\mathcal{H} = r(a,b) \vee B(a)$$

For problems of this nature, it is necessary to be able to perform inferences on, and produce as the result of an inference, negated role assertions of the form $\neg r(a,b)$. This would require the extension of the forgetting calculus $Int_{\mathcal{ALC}}$, which does not support negated role assertions. Alternatively, the calculus would need to be extended to allow nominals in both the input and output as is the case for the calculus of FAME utilised in Chapter 6. This would enable the representation of negated role assertions as follows: $\neg r(a,b)$ could be represented as $\forall r.\neg\{b\} \sqcup \neg\{a\}$.

In addition, the language of the input and output also needs to be extended further. In Chapter 5, disjunctions of concept assertions of the form $C(a_1) \vee ... \vee C(a_n)$ were utilised. These enabled the representation of the negated observation, where the observation takes the form of a conjunction of $\mathcal{ALC}$ concept assertions. Similarly, the final hypothesis could be represented as a disjunction of conjunctions of $\mathcal{ALC}$ concept assertions. Here, it is necessary to further extend the notion of disjunctive assertions to disjunctions of both concept and role assertions as in Example 7.2.2. It is also necessary to extend the TBox with a similar notion of disjunction, to enable the representation of disjunctions of TBoxes as identified in the previous section.

In this section several directions for extending the capabilities of forgetting-based abduction, including both TBox and ABox abduction, have been identified and discussed. In the next section, a generalisation of these two separate problems will be discussed: knowledge-base (KB) abduction [EKS06]. Since the result of forgetting-based TBox and ABox abduction takes the same form, i.e., a disjunction of possible explanations, it is natural to extend this problem to consider the more general case of accepting mixed observations containing both TBox and ABox axioms. First, the conditions specified in Chapter 4 Definition 4.2.1 must be extended to this more general case. Following this, in Section 7.4 the extensions made to the $Int_{\mathcal{ALC}}$ calculus to solve this problem in a more expressive language will be discussed.

7.3 Knowledge Base Abduction Problem

The characterisations of the ABox and TBox abduction problems in Definitions 5.1.1 and 7.1.2 both seek the same form of hypotheses: the semantically minimal space of independent explanations for the given observation. In both cases this takes the form of a disjunction, either of ABox assertions or GCI axioms respectively. Therefore, by extending the calculus used in Chapter 5 for both ABox and TBox abduction it is possible to tackle an instance of a more general abduction problem: KB abduction. As discussed in Chapter 3, KB abduction is a generalisation of ABox and TBox abduction, allowing observations and hypotheses that contain a mixture of both TBox and ABox axioms. In this setting, the specific KB abduction problem can be defined as follows:

Definition 7.3.1. *Let $\mathcal{K}$ be an $\mathcal{ALC}$ knowledge base, Ψ a set of TBox and ABox axioms in $\mathcal{ALC}$, and $\mathcal{S}_A$ be a set of abducible symbols such that $\mathcal{S}_A \subseteq sig(\mathcal{K},\Psi)$. The* KB abduction problem *$\langle \mathcal{K},\Psi,\mathcal{S}_A \rangle$ is then to compute a hypothesis $\mathcal{H} = \bigvee_{i=1}^{n} \mathcal{K}_i$, where each $\mathcal{K}_i$ is a KB expressed in $\mathcal{ALCOI}\mu$, such that $sig(\mathcal{H}) \subseteq \mathcal{S}_A$ and $\mathcal{H}$ satisfies the following conditions:*

(**i**) $\mathcal{K}, \mathcal{H} \not\models \bot$,

(**ii**) $\mathcal{K}, \mathcal{H} \models \Psi$,

(**iii**) *There is no disjunct* $\mathcal{K}_i$ *in* $\mathcal{H}$ *such that* $\mathcal{K}, \mathcal{K}_i \models \mathcal{K}_1 \vee ... \vee \mathcal{K}_{i-1} \vee \mathcal{K}_{i+1} \vee ... \vee \mathcal{K}_n$

(**iv**) *For any* $\mathcal{ALC}$ *KB* $\mathcal{H}'$ *satisfying conditions (i) – (iii)* $\mathcal{K}, \mathcal{H}' \models \mathcal{H}$.

Here, the hypotheses and abductive constraints take the same form conceptually as in Definition 4.2.1. The aim is still to compute the semantically minimal space of explanations as a disjunction, where here the disjuncts are now KBs. A KB is a conjunction of both TBox and ABox axioms, rather than ABox axioms. Here the term KB is used to distinguish from DL ontologies. Compared to the previous chapters the notion of a (Boolean) KB, defined in Section 2.5, extends the setting to include negation and disjunction of ontologies, introducing the notion of disjunctive TBoxes as motivated in previous sections. KBs will be referred to using the notation $\mathcal{K}$ and the set of observations using the notation Ψ to differentiate between KB abduction and the problem of ABox abduction discussed in previous chapters.

7.3.1 Language Extensions

The abduction problem setting is still assumed to be a background knowledge $\mathcal{K}$ and observations Ψ expressed in $\mathcal{ALC}$, together with a set of abducible symbols $\mathcal{S}_A$. Therefore, the conditions in Definition 7.3.1 are specified with respect to $\mathcal{ALC}$. However, the language used to express solutions extends $\mathcal{ALC}$, similarly to the extensions used in Chapter 6. The purpose of these extensions is to capture all possible $\mathcal{ALC}$ explanations for a given observation in a finite, compact form.

The most expressive language required to represent the hypothesis $\mathcal{H}$ is $\mathcal{ALCOI}\mu$, which is $\mathcal{ALC}$ extended with nominals, inverse roles and fixpoint expressions, the semantics of which are defined in Chapter 2. Each of these extensions has the aim of capturing all possible $\mathcal{ALC}$ explanations, thereby satisfying the intuitive notion behind condition (iv) of Definition 7.3.1: the hypothesis should be the semantically minimal one, i.e., a disjunction covering all possible $\mathcal{ALC}$ explanations for the given observation. The aim of the language extensions here is to increase the space of explanations that can be covered by the KB abduction approach, while ensuring that the hypotheses are as informative and compact as possible.

To illustrate, consider the following example:

Example 7.3.1. *Given the following KB abduction problem:*

$$\begin{aligned}
\mathcal{K} = \{&EbolaPatient \equiv Patient \sqcap \exists infectedWith.Ebola, \\
&\exists contactWith.EbolaCarrierBat \sqsubseteq EbolaPatient, \\
&EbolaPatient \sqsubseteq \forall infected.EbolaPatient, \\
&EbolaPatient(p_1), \\
&\forall contactWith.\neg EbolaCarrierBat(p_2)\} \\
\psi =&EbolaPatient(p_2) \\
\mathcal{S}_A = \{&contactWith, infected, EbolaCarrierBat\}
\end{aligned}$$

then a possible consistent, explanatory hypothesis is:

$$\exists contactWith.EbolaCarrierBat(p_1) \sqcap infected(p_1, p_2)$$

assuming that p_1 is the only individual in the ontology. However, assuming that there are other individuals, the following are also possible $\mathcal{ALC}$ explanations for the observation:

$$\begin{aligned}
&\exists contactWith.EbolaCarrierBat(a_1) \sqcap infected(a_1, p_2) \\
&\exists contactWith.EbolaCarrierBat(a_2) \sqcap infected(a_2, p_2) \\
&\exists contactWith.EbolaCarrierBat(a_3) \sqcap infected(a_3, p_2) \\
&\ldots
\end{aligned}$$

for each individual $a_i \in N_I$ for which neither of the two assertions above is explicitly *excluded. An alternative is to use inverse roles, resulting in the following hypothesis:*

$$\mathcal{H} = \exists infected^{-}.\exists contactWith.EbolaCarrierBat(p_2)$$

The main use of inverse roles in the proposed approach is to effectively, compactly represent sets of $\mathcal{ALC}$ explanations that use different individuals, as illustrated in Example 7.3.1. In cases where the number of individuals is large, there may be a large number of $\mathcal{ALC}$ explanations that account for the observation in the same way but use different individuals. In practice, this may result in a large number of repetitive explanations.

Another scenario involves a chain of relations connecting two individuals, such as in Example 7.3.2.

Example 7.3.2. *Consider the abduction problem from Example 7.3.1, but let the signature of abducibles be $\mathcal{S}_A = \{infected\}$. For each individual $a \in N_I$, the following is a possible explanation for the observation ψ:*

$$infected(p_1, a) \sqcap infected(a, p_2)$$

Alternatively, using nominals, the following hypothesis can be used to compactly capture this set of explanations:

$$\mathcal{H} = \exists infected.\exists infected.\{p_2\}(p_1)$$

Effectively, the use of a nominal in Example 7.3.2 bridges the gap between two individuals from the observation and background knowledge without the need to consider all known individuals. This also conforms well to the notion of semantic minimality, condition (iv) in Definition 7.3.1, under the open-world assumption (OWA) in OWL: it is not necessary to directly identify the connecting individuals to provide an explanation, thereby obtaining an explanation that makes fewer assumptions.

Finally, the motivation for using fixpoints is the same as in Chapter 5: to provide an option allowing cycles to be represented in the hypothesis. This is illustrated in Example 7.3.3.

Example 7.3.3. *Consider the abduction problem from Example 7.3.1. Due to the third axiom, it is possible to explain the observation via an infinite chain of infections from a given individual to the observed individual p_2. If fixpoints are allowed in the solutions, alongside inverse roles and nominals, the following hypothesis can be used to express this:*

$$\mathcal{H} = \mu X.(\exists contactWith.EbolaCarrierBat \sqcup \exists infected^{-}.\{p_1\} \sqcup \exists infected^{-}.X)(p_2)$$

As in Chapter 5, it is assumed that least, but not greatest, fixpoints may be required in the hypothesis.

The aim of the above extensions is to ensure that all possible $\mathcal{ALC}$ explanations can be contained within the hypothesis $\mathcal{H}$, extending the setting of Chapter 5. Since $\mathcal{H}$ should be the semantically minimal space of independent explanations, then the least restrictive solution to the abduction problem should cover all independent explanations that can be expressed in $\mathcal{ALC}$. As illustrated by the above cases, sometimes a concise solution requires the use of the above extensions to the expressivity of $\mathcal{ALC}$.

Now that the KB abduction problem and the language needed to express ideal solutions has been identified, a method for tackling the problem can be presented.

7.4 Knowledge Base Abduction Approach

To solve the problem in Definition 7.3.1 via a forgetting-based approach, the same general steps can be followed as in Figure 5.1:

> **Step 1: Contraposition.** Negate the observation ψ and add this to the background $\mathcal{K}$.
>
> **Step 2: Forgetting.** Obtain the strongest set of entailments of $(\mathcal{K}, \neg\psi)$ in the signature of abducibles $\mathcal{S}_A$ by computing the forgetting solution $\mathcal{V}$ of $(\mathcal{K}, \neg\psi)$

by eliminating the symbols that are not in $\mathcal{S}_A$.

Step 3: Filtering. Extract the reduced forgetting solution $\mathcal{V}^*$, i.e., the set $\mathcal{V}^* \subseteq \mathcal{V}$ that excludes all axioms in $\mathcal{V}$ that are redundant under the dual of Definition 7.3.1 condition (iii).

Step 4: Return hypothesis. Negate the set $\mathcal{V}^*$ to obtain the hypothesis $\mathcal{H}$ as a disjunction of KBs.

As the setting is more general, several extensions must be made to solve this problem compared to the approach used in Chapter 5. These include extensions to the forgetting calculus $Int_{\mathcal{ALC}}$ and the strategy with which this calculus is applied.

As before, Step (3) can be performed in two ways: *approximately*, resulting in the *approximate reduced forgetting solution* $\mathcal{V}^*_{app}$, or *fully*, resulting in the *reduced forgetting solution* $\mathcal{V}^*$. Negating the former returns an approximate hypothesis, satisfying Definition 7.3.1 conditions (i) and (ii), as well as condition (iv) without considering inter-disjunct redundancy. Negating $\mathcal{V}^*$ in Step (4) results in the hypothesis fully satisfying Definition 7.3.1: the space of independent explanations for Ψ.

In the ABox abduction approach of Chapter 5, the approximation step was done via annotation-based filtering, i.e., using the forgetting calculus $Int_{\mathcal{ALC}}$ as a black-box then filtering the result to remove redundant axioms. For the KB abduction approach in this chapter, approximation is instead performed by restricting the set of inferences that can be made during forgetting based upon the negated observation. As a result, the approximate filtering is effectively a part of Step (2), resulting in $\mathcal{V}^*_{app}$, while Step (3) focuses on performing entailment checking to eliminate any remaining redundancies resulting in $\mathcal{V}^*$. This difference is reflected in Figure 7.2, and will be described in subsequent sections.

Sections 7.4.1 – 7.4.2 describe the application of the extended forgetting approach, including the restriction of inferences to those concerning $\neg\Psi$, in Steps (1) and (2) of Figure 7.2. Section 7.5 then describes the full filtering procedure to eliminate all redundant explanations from the hypothesis, and examines differences between the

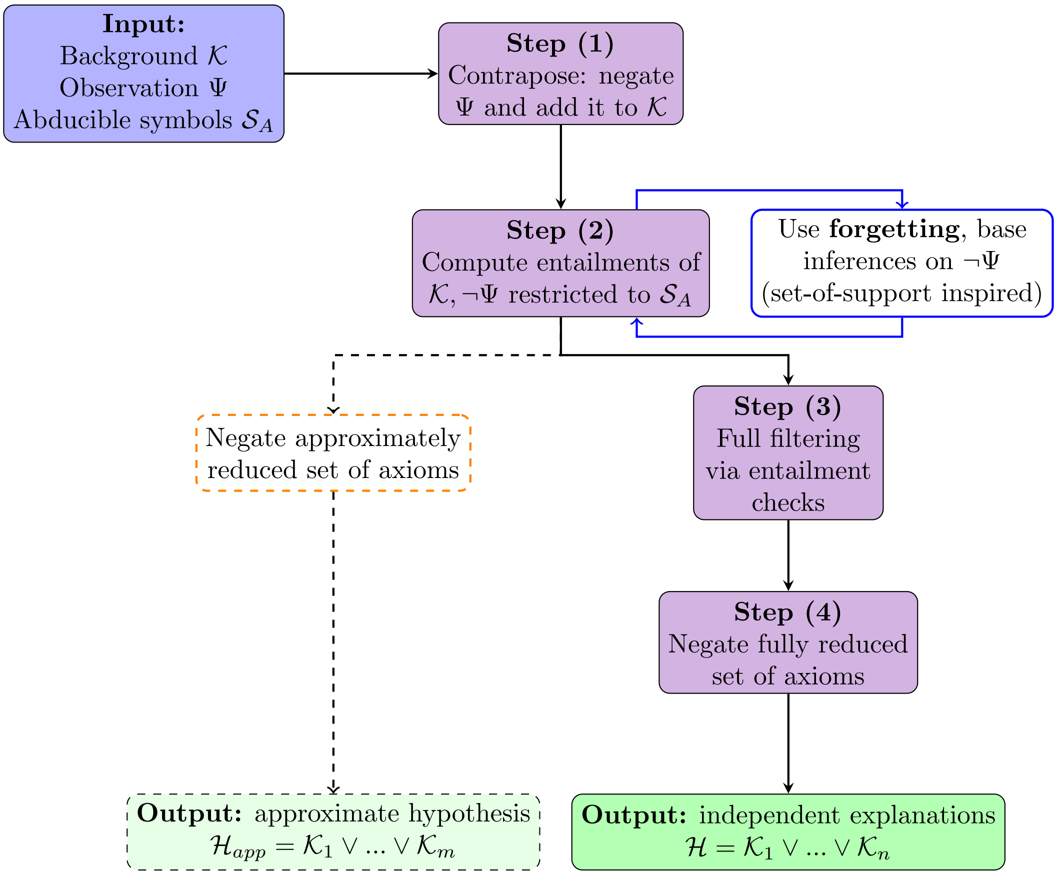

Figure 7.2: The forgetting-based KB abduction algorithm for computing hypotheses as spaces of independent explanations.

annotation-based approach to filtering and restricting inferences during forgetting.

7.4.1 Contraposition and Normalisation

As in Chapter 5 it is necessary to transform the input, a background ontology and a set of observations, into the appropriate form. The normalisation process is extended compared to the one utilised when applying $Int_{\mathcal{ALC}}$, as both the problem to be solved and the language used are more expressive.

First, the observation Ψ is a conjunction of both TBox axioms (GCIs) and ABox assertions. For the negated observation $\neg\psi$, the negation of a GCI $\neg(C \sqsubseteq D)$ takes the form $\exists\nabla.(C \sqcap \neg D)(a^*)$. For simplicity, the individual a^* is assumed here to be a fresh individual. This form is used for convenience sake so that the negated observation $\neg\Psi$ can be represented as a disjunction of assertions, since both negated TBox and negated ABox axioms in Ψ will be ABox axioms in $\neg\Psi$. Note that the notion of disjunctive assertion is slightly extended here. Each disjunct in $\neg\Psi$ may not necessarily be a concept assertion but can also be a negated role assertion $\neg r(a,b)$, while the hypothesis $\mathcal{H}$ may include role assertions $r(a,b)$ as explanations (disjuncts). The following example illustrates the process of transforming the input to the required normal form:

Example 7.4.1. *Consider the following observation.*

$$\begin{aligned}\Psi = \{&\exists s.F \sqsubseteq \exists s.E,\\ &C(a)\}\end{aligned}$$

The corresponding negated observation is represented as:

$$\neg\Psi = \exists\nabla.(\exists s.F \sqcap \forall s.\neg E)(a^*) \vee \neg C(a)$$

The corresponding clausal normal form for $\neg\Psi$ is as follows:

$$\exists\nabla.D_1(a^*) \vee \neg C(a)$$

$$\neg D_1(x) \vee \exists s.D_2(x)$$

$$\neg D_1(x) \vee \forall s.D_3(x)$$

$$\neg D_2(x) \vee F(x)$$

$$\neg D_3(x) \vee \neg E(x)$$

The general definition of the normal form utilised in this approach is provided in Definition 7.4.1 [KDTS20].

Definition 7.4.1. *Let N_D be a set of definers, which take the form of fresh concept names. Let $N_T = N_I \cup \{x\}$ be the set of terms, consisting of all the individual names N_I in the given domain together with a universally quantified variable x. A clause ϕ is a disjunction of literals $L_1 \vee ... \vee L_n$, where each literal L_i can take one of the following forms:*

$$A(t) \mid \neg A(t) \mid Qr.D(t) \mid r(a,b) \mid \neg r(a,b)$$

where A is a concept name, r is a role name, a and b are individual names, t is a term, $Q \in \{\exists, \forall\}$ and D is a definer symbol. At most one literal in a given clause ϕ may take the form $\neg D(x)$ where $D \in N_D$. It is assumed that there are no duplicate clauses and that a clause can be treated as a set of literals where the order of literals in the clause can be ignored.

The normal form above takes a similar form to the one in [KS15b], but here the language permits the extensions described in the previous section. Note, a clause $L_1(x) \vee ... \vee L_n(x)$ can be expressed in DLs as $\top \sqsubseteq L_1 \sqcup ... \sqcup L_n$, while ground clauses can be expressed as a disjunction of ABox assertions where each literal takes the form $C(a)$, $r(a,b)$ or $\neg r(a,b)$. Clauses that mix variables and ground terms are not introduced at any point. It is also assumed that sets of clauses can be added to KBs. Definer symbols are fresh concept symbols, as utilised in previous chapters. As before, the notation D_{12} is used to refer to the definer symbol representing $D_1 \sqcap D_2$ and previously introduced definer names are reused where possible.

As detailed above, the transformation of a negated observation $\neg\Psi$ will result in a disjunction of negated assertions (a clause): the negation of each GCI $\neg(C \sqsubseteq D)$ results in $\exists\nabla.(C \sqcap \neg D)(a)$ while the negation of each ABox assertion is simply a negated assertion. The full input to the method consists of a background KB $\mathcal{K}$ and the negated observation $\neg\Psi$: $\mathcal{K}, \Psi$, where $\mathcal{K}$ is a conjunction of TBox and ABox axioms. The GCIs $C_i \sqsubseteq D_i$ in $\mathcal{K}$ can each be represented equivalently as $\top \sqsubseteq \neg C_i \sqcup D_i$, which can be transformed to a clause $\neg C_i(x) \vee D_i(x)$. For each concept C, D, concepts occurring under role restrictions are replaced with definer symbols, similarly to the approach described for $Int_{\mathcal{ALC}}$ in Chapter 5, resulting in additional GCIs of the form $D_i \sqsubseteq C$. These can be treated in the same way as other GCIs. As a result, using standard CNF transformations, the input $\mathcal{K}, \neg\Psi$ is transformed into a set of clauses, where $\neg\Psi$ is guaranteed to be a ground clause.

7.4.2 Forgetting around $\neg\Psi$

The first extension that must be made to tackle the problem in Definition 7.3.1 is the need for additional rules in the calculus $Int_{\mathcal{ALC}}$, as motivated in previous sections.

The extended calculus is shown in Figure 7.3. Compared to the $Int_{\mathcal{ALC}}$ calculus in Figure 3.2 [KS15b], the *Resolution*, *Role Propagation* and *Role Instantiation* rules are the same. The $\exists$-*Role Restriction Elimination* rule in the $Int_{\mathcal{ALC}}$ calculus is replaced by two rules: the R$\exists$ rule, which is new, and the R∇ rule which fills a similar purpose. The R$\exists$ rule effectively captures the notion discussed in Example 7.2.1: retaining information during role forgetting via the use of the universal (top) role (∇). The rules Rr and R$\forall$-2 are also new, enabling inferences with and producing negated role assertions.

The forgetting calculus in Figure 7.3 can be utilised to compute a forgetting solution, which is the strongest set of entailments of the input $\mathcal{K}, \neg\Psi$ as in Theorem 3.6.1. As in Figure 5.1, a reduced forgetting solution $\mathcal{V}^*$ should then be extracted from this to ensure that entailments that are redundant with respect to the abduction problem are eliminated, i.e., explanations that do not satisfy Definition 7.3.1 condition (iii) are removed prior to returning the hypothesis.

Resolution:

$$\frac{\phi_1 \vee A(t_1) \qquad\qquad \phi_2 \vee \neg A(t_2)}{(\phi_1 \vee \phi_2)(\sigma)}$$

Role Propagation:

$$\frac{\phi_1 \vee (\forall r.D_1)(t_1) \qquad\qquad \phi_2 \vee Qr.D_2(t_2)}{(\phi_1 \vee \phi_2 \vee Qr.D_{12}(t_1))\sigma}$$

Role Instantiation:

$$\frac{\phi_1 \vee r(t_1, b) \qquad\qquad \phi_2 \vee (\forall r.D)(t_2)}{(\phi_1 \vee \phi_2 \vee D(b))\sigma}$$

Rr:

$$\frac{\phi_1 \vee r(a,b) \qquad\qquad \phi_2 \vee \neg r(a,b)}{\phi_1 \vee \phi_2}$$

R$\forall$-2

$$\frac{\phi_1 \vee \neg D(a) \qquad\qquad \phi_2 \vee (\forall r.D)(b)}{\phi_1 \vee \phi_2 \vee \neg r(b,a)}$$

R$\exists$

$$\frac{\phi_1 \vee \exists r.D(t)}{\phi_1 \vee (\exists \nabla.D)(t)}$$

R∇

$$\frac{\phi \vee (\exists \nabla.D)(t) \qquad \neg D(x)}{\phi}$$

where ϕ_1 and ϕ_2 are clauses, D_1 and D_2 are definer symbols, Q $\in \{\forall, \exists\}$, σ is the most general unifier of t_1 and t_2 if it exists, D_{12} is a new definer symbol for $D_1 \sqcap D_2$ and no clause contains more than one negative definer literal of the form $\neg D_i(x)$.

Figure 7.3: Extended forgetting calculus for the KB abduction problem [KDTS20].

Avoiding Unnecessary Inferences

As before, it is necessary to devise an efficient method for computing the reduced forgetting solution $\mathcal{V}^*$. Computing all possible entailments of $\mathcal{K}, \neg\Psi$ then performing the entailment check in Definition 7.3.1 condition (iii) directly is computationally infeasible, as demonstrated by the experimental results in Chapter 5 Section 5.6. In the setting of this chapter, this is even more pronounced: both the forgetting calculus and the expressivity of the language in which the problem is set are extended. As a result, the range of possible inferences that can be made using the calculus in Figure 7.3 is wider in scope.

Therefore, for the proposed KB abduction approach, the focus is instead on avoiding unnecessary inferences in the first place. This differs to the approach used in Chapters 5 and 6, for which the aim was to eliminate the unnecessary entailments in $\mathcal{V}$ as a post-processing step. Effectively, this means that the result of the forgetting step, Step (2) of Figure 7.2, is an approximation of the reduced forgetting solution $\mathcal{V}^*$. Note, however, that this approximation does not necessarily have the same properties as the annotation-based approximation $\mathcal{V}^*_{app}$ discussed in Chapter 5. As a result, the approximation obtained in this chapter will be referred to as $\mathcal{V}^s_{app}$ to differentiate from $\mathcal{V}^*_{app}$. The fully reduced forgetting solution will be referred to as $\mathcal{V}^*$ as before, since this is the same in both cases. Differences between the two approaches to avoiding or eliminating redundancy will be discussed in Section 7.5.

Inferences made using the calculus in Figure 7.3 are restricted using a set-of-support [Pla94] inspired strategy similar to the one suggested in [KS14a]. Clauses in the normalised form of $\mathcal{K}, \neg\Psi$ are split into two sets: the *background set* Φ_B, which contains all clauses in the normalised form of $\mathcal{K}$, and the *support set* Φ_S, which contains all clauses in the normalised form of $\neg\Psi$. For each symbol in the forgetting signature $\mathcal{F}$, the following steps are repeated making use of the calculus in Figure 7.3 until all symbols have been covered, where each inference is restricted so that at least one of the premise (parent) clauses comes from the support set Φ_S:

1. Perform all inferences on symbols in the forgetting signature $\mathcal{F}$, as well as all

possible inferences via the rules *RA* and *R*$\forall$-2 on definer symbols. As with $Int_{\mathcal{ALC}}$, all inferences using the role propagation and role instantiation rules that enable further inferences on symbols in $\mathcal{F}$ are also performed. All inferred clauses are added to the support set Φ_S, provided that they have not already been derived previously.

2. Remove clauses containing symbols in $\mathcal{F}$ from Φ_S.

3. If a clause containing a definer is derived and thus added to Φ_S, move all clauses containing this definer from Φ_B to Φ_S.

To ensure that this approach terminates, it is necessary to store any clauses that have been derived and added to the support set previously in a separate set so that they are not derived more than once. This is due to the fact that previously forgotten symbols could potentially be reintroduced to Φ_S during the process, so repeat derivations must be avoided. The need to add all clauses containing definers, in (3) above, is an extension of the set of support style approach. This ensures that all connections between clauses and definers are retained in the support set Φ_S, enabling further inferences on definer symbols.

By restricting inferences performed using the calculus in Figure 7.3 to cases where one of the premises is not contained in the background knowledge, i.e. clauses from the support set, the number of consequences that do not depend on the negated observation $\neg\psi$ that are computed using the calculus in Figure 7.3 is reduced. As shown in Lemma 5.4.1, consequences that are not dependent on $\neg\psi$ are guaranteed to be redundant with respect to the abduction problem.

Note that any TBox axioms in $\mathcal{V}^s_{app}$ will not be dependent on the negated observation $\neg\Psi$, since any inference on the representation of $\neg\Psi$ will instead result in an ABox assertion over an existing individual or over the fresh individual a^* used to represent the negation of TBox axioms in Ψ. The only other TBox axioms will be introduced for the sole purpose of eliminating definer symbols, as described in the next section. Thus, all TBox axioms in $\mathcal{V}^*_{app}$ can be discarded prior to performing the entailment checks for the dual of Definition 7.3.1 condition (iii) to obtain $\mathcal{V}^*$.

In this way, the set-of-support inspired strategy provides an in-built way to compute an approximation, $\mathcal{V}^s_{app}$, of the reduced forgetting solution $\mathcal{V}^*$. This fulfils a similar role as the annotation-based filtering utilised in Chapter 5. However, it is first necessary to denormalise the set of clauses returned by the set-of-support inspired approach taken above, in order to eliminate definers and return $\mathcal{V}^s_{app}$ as a KB.

Denormalisation

Once all inferences using the calculus in Figure 7.3 have been computed using the set-of-support inspired approach, it is necessary to transform the resulting set of clauses into a KB that does not contain any definer symbols. The set of clauses returned by the set-of-support inspired approach, i.e., the final saturated state of the support set Φ_S. Definer elimination is performed so that all $\mathcal{ALC}$ entailments of the input are preserved, aside from those which use definer names. As a result, the set of entailments required to preserve all $\mathcal{ALC}$ explanations in the final hypothesis are preserved.

As in the forgetting procedure of $Int_{\mathcal{ALC}}$ [KS15b], definers are eliminated via the introduction of concept inclusions (CIs) of the form $D_i \sqsubseteq C$ where D_i is a definer and C is a concept. However, previously clauses could not contain concepts of the form $\neg D_i(a)$. For the extended calculus, it is possible to produce clauses of the form $\neg D_i(a) \vee \phi$ where ϕ is an arbitrary clause. Thus, the definition of a definer may in fact refer to an individual a, for which nominals are required. This is also relevant when there are clauses of the form $\phi' \vee \forall r.D(t)$, since it is then necessary to utilise inverse roles and substitute $\neg D(a)$ with $\forall r^-.C(a)$, where C is a concept corresponding to ϕ'.

To introduce the required CIs, it is necessary to first introduce a representation for negative definer literals $\neg D_i$. Note that each time a new definer D_i is introduced, a clause of the form $\neg D_i(x) \vee C(x)$ is added. Therefore, if we assume that all possible inferences using the *Resolution* rule in Figure 7.3 have been applied on positive definers, it is not necessary to introduce a representation for clauses containing positive definer literals of the form $D(t) \vee \phi$. This is due to the fact that all positive definer literals will be resolved with a corresponding negative definer literal.

For each definer D_i in the set of clauses derived from $\mathcal{K}, \neg\Psi$ during the forgetting step, the negation $\neg D_i$ is replaced by a fresh definer $\overline{D}_i$. For a concept C, the concept obtained by replacing all occurrences of $\neg D_i$ in C with $\overline{D}_i$ is denoted as C^{-}. For a clause $\phi = L_1 \vee ... \vee L_n$, a concept $C^{\phi} = L_1^c \vee ... L_n^c$ is introduced. Each L^c is defined as follows:

$$
\begin{array}{lll}
C^{-} & \text{if} & L = C(x) \\
\exists\nabla.(\{a\} \sqcap C^{-}) & \text{if} & L = C(a) \\
\exists\nabla.(\{a\} \sqcap \exists r.\{b\}) & \text{if} & L = r(a,b) \\
\exists\nabla.(\{a\} \sqcap \forall r.\neg\{b\}) & \text{if} & L = \neg r(a,b)
\end{array}
$$

where in each case, the new axioms are added before the corresponding clause is removed after replacement. As mentioned earlier, no clause mixes both variables and individual names. Thus, the two cases can be treated separately: clauses containing only variables are handled by applying the first translation rule above to all disjuncts in the clause, while the latter three are sufficient for clauses containing only individuals.

It is then possible to introduce a concept inclusion to give meaning to each definer symbol. For each clause containing a definer symbol, a corresponding concept inclusion is introduced. For a clause containing a definer D_i, the following case is handled similarly as in [KS15b]:

$$
\begin{array}{lll}
\neg D_i(x) \vee \phi & \text{introduce} & D_i \sqsubseteq C^{\phi}
\end{array}
$$

while the following cases require the introduction of nominals and inverse roles:

$$
\begin{array}{lll}
\phi \vee \forall r.D_i(x) & \text{introduce} & \overline{D}_i \sqsubseteq \forall r^{-}.C^{\phi} \\
\phi \vee \forall r.D_i(a) & \text{introduce} & \overline{D}_i \sqsubseteq \forall r^{-}.(\neg\{a\} \sqcup C^{\phi}) \\
\neg D_i(a) \vee \phi & \text{introduce} & D_i \sqsubseteq \neg\{a\} \sqcup C^{\phi}
\end{array}
$$

Any remaining occurrences of literals of the form $\neg D_i(a)$ are replaced by $\overline{D}_i$ and all clauses that are not disjunctions of ABox assertions are replaced with $\top \sqsubseteq C^{\phi}$. Once this is done every definer D_i, including each corresponding $\overline{D}_i$, has an associated axiom of the form $D_i \sqsubseteq C$. As a result, the meaning of each definer symbol is represented via a concept inclusion.

The definer elimination technique shown in Figure 3.3 [KS15b] can then be applied to obtain the forgetting solution not containing definer symbols. Since the introduced axioms are only needed for the purpose of definer elimination, they can be discarded once the elimination process is complete. This leaves the definer-free set of clauses capturing all the required entailments of $\mathcal{K}, \neg\Psi$, with respect to the abduction problem, of $\mathcal{K}, \neg\Psi$ within the signature of abducibles $\mathcal{S}_A$.

7.4.3 Eliminating Remaining Redundant Explanations

Once the set-of-support inspired approximation of the reduced forgetting solution $\mathcal{V}^S_{app}$ is returned, the full filtering procedure as in Step (3) of Figure 7.2 can be performed.

If the aim is to eliminate all inter-disjunct redundancies, including those nested inside concepts, then disjunctions must be pulled out. The steps for this are standard rules to transform the result into disjunctive normal form (DNF) as in Chapter 5, where a disjunction of KBs is in DNF if every disjunctive concept $C \sqcup D$ occurs only in a concept inclusion of the form $E \sqsubseteq C \sqcup D$ or in an assertion under a universally quantified role restriction. In this way, disjuncts are checked with respect to the inter-axiom redundancy notion discussed in Section 7.1, where here each disjunct is a $\mathcal{K}$. The intra-axiom redundancy notion is not addressed by this approach, since disjunctions may occur within CIs.

As with the ABox abduction approaches, filtering can be performed either *approximately* or *fully*. The approximate or *initial hypothesis* is obtained by directly negating the result of the set-of-support inspired approach after denormalisation, and satisfies the notions of consistency, explanation and semantic minimality without the inter-disjunct redundancy condition. For the approximate case, it is necessary to perform an additional check $\mathcal{K} \wedge \mathcal{H} \not\models \bot$ to ensure that the initial hypothesis is not inconsistent.

Once disjunctions have been pulled out, the full filtering procedure can be performed to obtain a hypothesis satisfying Definition 7.3.1, i.e., a set of *independent explanations*. The following entailment check is performed for each $\mathcal{K}_i$ in the hypothesis.

$$\mathcal{K} \wedge \mathcal{K}_i \models \mathcal{K}_1 \vee \ldots \vee \mathcal{K}_{i-1} \vee \mathcal{K}_{i+1} \vee \ldots \vee \mathcal{K}_n$$

for each disjunct in the hypothesis. If the entailment check returns true, then the disjunct $\mathcal{K}_i$ is redundant with respect to Definition 7.3.1 and is removed. As in Chapter 5, in practice it is not possible to determine if a disjunct containing a fixpoint operator is redundant with respect to Definition 7.3.1 condition (iii). This is due to the fact that the above check is performed using an external reasoner, and there are at present no DL reasoners that can handle fixpoint operators. Therefore, the entailment check proceeds as follows in practice:

$$\mathcal{K} \wedge \mathcal{K}_1 \wedge \ldots \wedge \mathcal{K}_{i-1} \wedge \mathcal{K}_{i+1} \wedge \ldots \wedge \mathcal{K}_n \wedge \neg\mathcal{K}_i \models \bot$$

where for $\mathcal{K}_j$, $j \neq i$, greatest fixpoints $\nu X.C[X]$ are simulated by replacement with $D \sqsubseteq C[D \to X]$ where D is a fresh concept name for the purpose of eliminating any redundant disjuncts KB_i in $\mathcal{H}$ that do not contain fixpoints.

7.4.4 Negating the Forgetting Solution

To obtain the final hypothesis, it is necessary to negate the KB obtained either immediately after the set-of-support inspired strategy ($\mathcal{V}^S_{app}$), or after the full filtering procedure has been applied to this result ($\mathcal{V}^*$).

The concept inclusions introduced to eliminate definers can be removed prior to filtering, since these are no longer needed once the denormalisation step is complete. The result is a conjunction of disjunctions of assertions, since the negated observation in Step (1) is represented as $\exists\nabla.(C \sqcap \neg D)(a^*)$ where a^* is a fresh individual.

At this point, occurrences of the top role ∇ can be removed. When introduced, these only occur under existential quantifiers: during the negation of Ψ, by inference via the $R\exists$ rule in the calculus in Figure 7.3 during Step (2) and during the denormalisation phase of the forgetting procedure. Upon negating the result, once all negations have been pushed inwards the top role will only occur under universal quantification. The following equivalences, as well as standard DNF transformations, are used to pull out these occurrences:

$$
\begin{aligned}
\exists r.(C_1 \sqcap \forall\nabla.C_2) \quad &\iff \quad \exists r.C_1 \sqcap \forall\nabla.C_2 \\
\forall r.(C_1 \sqcup \forall\nabla.C_2) \quad &\iff \quad \forall r.C_1 \sqcup \forall\nabla.C_2 \\
(\forall\nabla.C)(a) \quad &\iff \quad \top \sqsubseteq C
\end{aligned}
$$

For fixpoint operators, pushing the negation inwards ensures that all occurrences of fixpoints in the hypothesis obtained will be least fixpoints. This is due to the fact that during the denormalisation phase of the forgetting process in Step (2), the definer elimination procedure only introduces greatest fixpoints.

To show that the presented abduction approach computes hypotheses satisfying Definition 7.3.1, up to possible inter-disjunct redundancy of disjuncts containing fixpoint operators, it is necessary to show that the set-of-support inspired approach to applying the calculus in Figure 7.3 during Step (2) computes the set of all *relevant* entailments of $\mathcal{K}, \neg\Psi$, i.e., those with an ancestor in $\neg\Psi$. Additionally, it is necessary to show that all of these relevant entailments that do not involve definer symbols are retained during the denormalisation phase. Once the set of relevant entailments is obtained in Step (2), the soundness of the abduction approach can be shown similarly as in Chapter 5. The aforementioned proofs, originally presented in the extended version of [KDTS20], can be found in the appendix.

The following abduction problem is used to illustrate the full approach:

Example 7.4.2. *Consider the following background KB:*

$$\begin{aligned}\mathcal{K} = \{&A \sqsubseteq C,\\ &\exists r.B \sqsubseteq C,\\ &F \sqsubseteq G,\\ &C \sqsubseteq I,\\ &A \sqcap H \sqsubseteq \bot,\\ &H(a)\}\end{aligned}$$

and the observation Ψ *from Example 7.4.1:*

$$\begin{aligned}\Psi = \{&\exists s.F \sqsubseteq \exists s.E,\\ &C(a)\}\end{aligned}$$

Let the set of abducible symbols be $\mathcal{S}_A = \{A, B, E, G, H, I, r\}$*. In Step (2), forgetting non-abducibles proceeds as follows. First,* $\mathcal{K}$ *and* $\neg\Psi$ *are transformed into normal form, resulting in the background* Φ_B *and supported* Φ_S *clause sets:*

Φ_B	Φ_S
$bg1)\neg A(x) \vee C(x)$	$1)\exists\nabla.D_1(a^*) \vee \neg C(a)$
$bg2)\forall r.D_4(x) \vee C(x)$	$2)\neg D_1(x) \vee \exists s.D_2(x)$
$bg3)\neg D_4(x) \vee \neg B(x)$	$3)\neg D_1(x) \vee \forall s.D_3(x)$
$bg4)\neg F(x) \vee G(x)$	$4)\neg D_2(x) \vee F(x)$
$bg5)\neg C(x) \vee I(x)$	$5)\neg D_3(x) \vee \neg E(x)$
$bg6)\neg A(x) \vee \neg H(x)$	
$bg7)H(a)$	

Inferences under the calculus in Figure 7.3 are used to eliminate the non-abducible *symbols $\mathcal{F} = \{s, C, F\}$, which proceeds as follows. Starting with C:*

$$6)\exists\nabla.D_1(a^*) \vee \neg A(a) \qquad \textit{Resolution(bg1, 1)}$$

$$7)\exists\nabla.D_1(a^*) \vee \forall r.D_4(a) \qquad \textit{Resolution(bg2, 1)}$$

where, for example, the resolution between bg1 and bg5 on C is avoided. Following this, clause (1) is removed from Φ_S, since it contains C, while bg2 and bg3 are added to Φ_S due to the definer D_4. Next, inferences proceed as follows, where the symbol F can also be eliminated:

$$8)\forall r.D_4(x) \vee I(x) \qquad \textit{Resolution(bg3, bg5)}$$

$$9)\neg D_2(x) \vee G(x) \qquad \textit{Resolution(bg4, 4)}$$

where clause (4) is then removed from Φ_S. Now, s must be eliminated:

$$10)\neg D_1(x) \vee \exists s.D_{23}(x) \qquad \textit{Role propagation(2, 3)}$$

$$11)\neg D_{23}(x) \vee D_2(x)$$

$$12)\neg D_{23}(x) \vee D_3(x)$$

$$13)\neg D_{23}(x) \vee G(x) \qquad \textit{Resolution(9, 11)}$$

$$14)\neg D_{23}(x) \vee \neg E(x) \qquad \textit{Resolution(5, 12)}$$

$$15)\neg D_1(x) \vee \exists\nabla.D_{23}(x) \qquad \textit{R}\exists\textit{(10)}$$

The final state of the support set Φ_S contains clauses (5–9), clauses (11–15) as well as the introduced background clauses bg2 and bg3. Following the denormalisation phase, the set-of-support based approximation of the reduced forgetting solution, $\mathcal{V}^S_{app}$, is:

$$\{\exists\nabla.(\neg E \sqcap G)(a) \vee \forall r.\neg B(a),$$
$$\exists\nabla.(\neg E \sqcap G)(a) \vee \neg A(a)\}$$

noting that any remaining TBox axioms can be discarded. Negating $\mathcal{V}^S_{app}$, as in the *approximate filtering approach, results in the initial hypothesis:*

$$\mathcal{H} = \{G \sqsubseteq E, A(a)\} \vee \{G \sqsubseteq E, \exists r.B(a)\}$$

while applying the full filtering procedure eliminates the redundant disjunct, resulting in the final hypothesis (independent explanations) as follows:

$$\mathcal{H} = \{G \sqsubseteq E, \exists r.B(a)\}$$

7.5 Comparing Redundancy Elimination Strategies

The set-of-support inspired approach has the advantage that it limits the number of inferences made using the extended calculus, unlike the annotation-based approach which is applied as a form of post-processing once all inferences have been made. This is more impactful in this setting, since the calculus has been extended and thus the number of possible inferences if the calculus was applied without restriction is likely to be larger than for the calculus in Figure 3.2.

In [KDTS20], redundancy elimination focuses on two notions: the inter-axiom redundancy between disjunctions of GCIs as discussed earlier in this chapter and the redundancy between disjuncts as concept and role assertions analogous to the inter-disjunct redundancy notion used in Chapter 5. To illustrate:

Example 7.5.1. *For the following abduction problem:*

$$\begin{aligned}\mathcal{O} = \{&A \sqsubseteq C,\\ &B \sqsubseteq C,\\ &B \sqcap F \sqsubseteq \bot,\\ &\exists r.C \sqsubseteq D,\\ &D \sqsubseteq E,\\ &C(b)\}\end{aligned}$$

$$\begin{aligned}\psi =& \{F \sqsubseteq C, \\ & E(a)\} \\ \mathcal{S}_A =& \{A, B, D, r\}\end{aligned}$$

Consider the following three candidate hypotheses:

$$\begin{aligned}\mathcal{H}_1 =& \{F \sqsubseteq A, D(a)\} \vee \{F \sqsubseteq A, r(a,b)\} \\ \mathcal{H}_2 =& \{F \sqsubseteq A \sqcup B, D(a)\} \\ \mathcal{H}_3 =& \{F \sqsubseteq A, D(a)\}\end{aligned}$$

under Definition 7.3.1, $\mathcal{H}_1$ is not an acceptable solution since the second disjunct is redundant: $\mathcal{O}, r(a,b) \models D(a)$ and thus the second disjunct is stronger than the first under $\mathcal{O}$. Both $\mathcal{H}_2$ and $\mathcal{H}_3$ are acceptable hypotheses under the definition, despite the fact that the B in the superclass of the GCI in $\mathcal{H}_2$ is somewhat redundant: since B and F are disjoint, it cannot be the case that $B \sqsubseteq F$.

In fact, for the above example the system described in this section computes the hypothesis $\mathcal{H}_2$. It is worth noting that, as discussed earlier, applying the intra-axiom redundancy notion would further reduce unnecessary complexity and redundancy in the hypothesis. However, the consideration of whether or not it is worthwhile is a practical one. In this setting, disjunctions may be more common than in the TBox abduction setting due to the presence of ABox axioms in the observation, a possibility that is examined in the experimental results in Section 7.6. This, combined with the increased complexity of the result in terms of the language used means that the intra-axiom redundancy notion may be too expensive to apply in practice, given that entailment checks are already used to eliminate redundant disjuncts occurring between GCIs and within disjunctive ABox assertions.

However, there are some redundancies that are not captured by the set-of-support approach that would otherwise be detected via annotations as demonstrated below:

Example 7.5.2. *Consider the following abduction problem:*

$$\begin{aligned}
\mathcal{O} = &\{G \sqsubseteq A,\\
&\exists s.A \sqsubseteq C,\\
&C \sqsubseteq E,\\
&F \sqsubseteq \forall r.A\\
&\exists r.\neg A(a)\}\\
\psi = &C(a)\\
\mathcal{S}_A = &\{s, E, F, G\}
\end{aligned}$$

Under the set of support inspired approach, the following are the background and support sets:

Φ_B	Φ_S
$bg1)\forall s.D_1(x) \vee C(x)$	$1)\neg C(a)$
$bg2)\neg D_1(x) \vee \neg A(x)$	
$bg3)\neg C(x) \vee E(x)$	
$bg4)\neg F(x) \vee \forall r.D_2(x)$	
$bg5)\neg D_2(x) \vee A(x)$	
$bg6)\exists r.D_1(a)$	
$bg7)\neg G(x) \vee A(x)$	

the following inferences are then performed using the calculus in Figure 7.3, applied under the strategy described in Section 7.4.2:

$$2)\forall s.D_1(a) \qquad\qquad \mathit{Resolution}(1, bg1)$$

due to this inference, all clauses that contain the definer D_1 *are added to the support set as described in point (3) of the forgetting step in Section 7.4.2. This includes clauses bg1, bg2 and bg6. As such, the inferences resulting in clauses 3 and 9 below are now permitted as at least one parent is present in the support set.*

$3)\neg F \vee \exists r.D_{12}(a)$	$Role\ propagation(bg4, bg6)$
$4)\neg D_{12}(x) \vee A(x)$	
$5)\neg D_{12}(x) \vee \neg A(x)$	
$6)\neg D_{12}(x)$	$Resolution(4,5)$
$7)\neg F \vee \exists\nabla.D_{12}(a)$	$R\exists(3)$
$8)\neg F(a)$	$R\nabla(6,7)$
$9)\neg D_1(x) \vee \neg G(x)$	$Resolution(bg2, bg7)$

this results in the approximately reduced forgetting solution $\mathcal{V}^s_{App}$*:*

$$\mathcal{V}^s_{App} = \{\forall s.\neg G(a),\\ \neg F(a)\}$$

which if directly negated gives the candidate hypothesis:

$$\mathcal{H}_s = \exists s.G(a) \vee F(a)$$

However, if instead the annotation-based filtering approach in Chapter 5 is used to extract $\mathcal{V}^*_{app}$*, the following result is obtained:*

$$\mathcal{H} = \exists s.G(a)$$

Thus, the redundancy $F(a)$ *is retained via the set of support based approach, unless annotations are also used during the process. In practice, this requires an additional entailment check during the strict filtering procedure when using the set-of-support* *inspired approach alone, which would likely increase the cost.*

The redundancy retained by the set of support based approach in Example 7.5.2 is due to the fact that once a clause containing a definer symbol has been inferred and added to the support set, it is also necessary to move all background clauses containing this definer to the support set.

Combining the annotation-based approach with the set-of-support inspired approach may be promising. Each entailment in $\mathcal{V}^*_{app}$ must be checked against the dual of Definition 7.3.1 condition (iii) to ensure that the reduced forgetting solution $\mathcal{V}^*$ is extracted. As a result, eliminating any given redundancy in $\mathcal{V}$ reduces the number of entailment checks that must be made. Therefore, combining these approaches would likely reduce the time taken to perform the full filtering required to ensure that each explanation in $\mathcal{H}$ is both as informative as possible and independent from the other explanations.

7.6 Experimental Evaluation

The aim of the experiments in this section was to evaluate the performance of the KB abduction approach across a variety of scenarios, split primarily into abduction problems for which the observation was a set of ABox axioms, a set of TBox axioms or a KB including both types of axioms.

To run these experiments, a prototype was implemented in Java (and Scala). The prototype makes use of the OWL-API. The latest implementation of LETHE, at the time of writing, was extended with the set-of-support inspired forgetting approach and the filtering steps required to eliminate redundant disjuncts from the hypothesis.

7.6.1 Corpus

The experiments were carried out over a corpus of ontologies extracted from the 2017 snapshot of NCBO Bioportal [MP17]. The ontologies in the corpus were restricted to those satisfying a set of requirements, the aim of which was to find a balance between the practicality of running the experiments on the available resources and emphasising interesting and non-trivial abduction problems. These requirements are as follows:

- They must contain between 100 and 40,000 axioms. This was done to ensure that there was sufficient background information to provide a non-trivial abduction problem and to increase the number of non-empty hypotheses. The upper limit was imposed for the purpose of practicality, both in generating a sufficient number of satisfactory abduction problems for the experiments as well as performing the experiments within a reasonable time limit.

- The ontology must contain both a TBox and an ABox. By providing both of these components, KB, TBox and ABox observations must all be explained with respect to existing knowledge. If both of these components are present, the hypothesis must be situated appropriately with respect to the schema defined by the TBox, but also cover the data on existing individuals. For example, an ABox observation may be explained with respect to the relationship between observed individuals and existing individuals. Without existing data on individuals, these cases would be excluded. Similarly, without an upper schema, the abduction problem becomes significantly less interesting: either there will be no explanation for an observation or the explanation could be trivial since there are no existing constraints.

- They must be consistent and parsable using the OWL-API, LETHE and the reasoner HermiT. Otherwise, the process of explanation is trivial since all observations will already be entailed by the ontology and any arbitrary "explanation" is permissible. Thus, the abduction problem is separated from the problem of explanation under inconsistent knowledge. If an error was encountered when checking the consistency of the ontology, then it was excluded from the corpus.

Ontologies satisfying these criteria were then restricted to their $\mathcal{ALC}$ fragment. The final experimental corpus consisted of 115 ontologies, the statistics for which are summarised in Table 7.1.

	Min	Mdn.	Mean	90^{th} Percentile	Max
TBox Axioms	27	948	2411	5841	29770
ABox Axioms	1	60	1564	2829	30887
Concept Symbols	5	432	1435	3554	10939
Role Symbols	0	44	97	308	1390
Individuals	0	39	609	1323	12153

Table 7.1: Characteristics of the experimental corpus.

7.6.2 Generating Abduction Problems

A primary goal behind these experiments was to avoid making simplifying assumptions regarding the abduction problem. Few computationally efficient abduction approaches exist for large ontologies, particularly in the case of KB abduction for which, to the best of the author's knowledge, no implemented systems currently exist. In the context of DLs, no works address the form of the abduction problem solved in this thesis: generating spaces of independent explanations. Therefore, the exact nature of abduction problems in real application scenarios is not well understood, meaning no guidelines or benchmarks exist for the abduction problems that form the subject of these experiments. Therefore, strong assumptions on the structure, size and interrelatedness of statements in the observations were avoided. For example it is not assumed that, for a given observation, each axiom in the set is somehow related to each other axiom in the set. In effect, this means that the user is not expected to know exactly how each new statement in a given observation relates to one another.

For each experiment, the algorithm was tested across three types of abduction problems, based on the type of observation: ABox observations, TBox observations and KB observations. For the first two, only ABox or TBox axioms respectively were present in the set of axioms that made up a given observation. For the KB observations, both TBox and ABox axioms were permitted in the observation, where each observation contained at least one TBox axiom and at least one ABox axiom to ensure that the problem was a KB abduction problem. The ratio of TBox to ABox axioms was decided at random up to the total number of axioms specified for the observation.

The observation generation procedure is similar to the ones used in [DS19a, DS19b, KDTS20]. However, since there are several fundamental differences, it is described here for clarity. For each abduction problem, the observation was first generated using the procedure described below and was then passed along with the corresponding background ontology and signature of abducibles to the KB abduction algorithm.

The ABox axioms in the observations were generated randomly using the signature of concept and role symbols in the given background ontology. Concept assertions of the form $C(a)$ were combined at random with $\mathcal{ALC}$ operators: disjunction ($\sqcup$), existential restrictions ($\exists$), universal restrictions ($\forall$) and negation ($\neg$) to produce $\mathcal{ALC}$ concept expressions. Conjunction was excluded since the observations are already provided as sets, interpreted conjunctively, of new statements and thus the use of conjunction is based on the number of axioms in each observation. For role assertions, the object property was used to construct a statement of the form $r(a,b)$. The individuals to which each axiom were applied were chosen from the existing individuals in the background ontologies. If too few individuals (< 5) were present in the ontology, then fresh individual names were used to supplement the set from which individuals were drawn. The reasoning behind this strategy was to provide diverse sets of observations that conformed to the language requirements of the abduction problem being investigated, without being too simple. Drawing the observations from the ABox of each ontology was considered, but the generation approach was preferred to ensure that the statements occurring in ABox observations covered a range of different complexities, making use of each of the operators in $\mathcal{ALC}$.

The TBox axioms in the observations were selected at random from the given background ontology. These axioms were then removed from the background ontology to ensure that the resulting observation was not entailed by it. The aim was to ensure that a reasonable number of non-empty hypotheses were obtained for observations containing TBox axioms. Therefore, the selection approach was preferred over generating inclusion axioms, in a similar way to the ABox observation case above, to ensure that the TBox axioms conformed to the schema of each given ontology. The selection approach also confers the advantage that it is easier to generate suitable, i.e., consistent observations for the KB and TBox cases.

For each abduction problem, 20 attempts were made to generate an observation satisfying the abduction problem, i.e., a consistent and non-entailed observation under the given background ontology. If no such observation was generated in 20 attempts, then an inconsistent observation was permitted and the algorithm was applied as usual. This methodology strikes a balance between the practicality of running the experiments and the need for interesting abduction problems. Despite the fact that these observations were inconsistent, contrary to the assumptions in Definition 7.3.1, this still provided data on the runtime of the algorithm and the effectiveness of the filtering procedure, since each step must be completed as normal. The difference is that for these cases, the hypothesis was guaranteed to be empty since no explanations satisfying Definition 7.3.1 would exist for an inconsistent observation.

Signatures of abducible symbols were chosen randomly based on the frequency of occurrence in the given background ontology: the chance of a symbol being specified as an abducible was proportional to how often it occurred in the ontology. This way, symbols used extensively in an ontology were less likely to be excluded from the explanations produced. For each abduction problem, a random percentage of the background ontology was specified as abducible between the limits specified for each experiment.

An additional requirement for the signature of abducibles was that, for each abduction problem, at least one symbol in each axiom in the observation must be non-abducible. While this is a strong requirement, this ensures that the problem that is solved is the most challenging and most interesting case: each new axiom provided must be explained in a new way. If the signature of an axiom in the observation does not contain a non-abducible symbol, then in the semantically minimal hypothesis returned, the axiom will simply be returned as it is which is not a particularly interesting case. One exception to this requirement is that role symbols in the observation were not forcibly specified as additional non-abducibles in this way. Preliminary testing showed that the number of empty hypotheses increased significantly when role symbols were forcibly specified as additional non-abducibles. This is likely because the ontologies in the corpus often contain far fewer role symbols, but the role symbols that are used are used frequently and play a key part in many statements, as indicated in Table 7.1. For example, attempting to produce explanations that exclude the “hasPart” role symbol in an ontology about anatomy was found to result in a large number of empty hypotheses.

7.6.3 Experiments Performed

For the ABox, TBox and KB observation problems described above, three experiments were performed as follows:

- **Experiment 1:** Observations containing between 1 and 10 axioms, with 1–30% of the total symbols being specified as non-abducibles.
- **Experiment 2:** Observations containing between 1 and 10 axioms, with 31–60% of the total symbols being specified as non-abducibles.
- **Experiment 3:** Observations containing between 11 and 20 axioms, with 1–30% of the total symbols being specified as non-abducibles.

Each experiment was run over three sets of abduction problems: 30 ABox, 30 TBox and 30 KB observations for a total of 90 abduction problems per experiment.

These three sets of experiments are intended to investigate the effect of varying two of the main inputs to the abduction system, under the conditions described in the previous section. These are the size of the observation in terms of the number of axioms it contains, and the total number of symbols specified as non-abducibles. In this sense, experiment 1 can be viewed as a baseline against which experiments 2 and 3 are compared.

The proportion of non-abducibles and the number of axioms in the observation are expected to influence the time taken to compute a hypothesis. As the number of axioms in the observation increases it is likely that further inferences will be performed, assuming that each axiom in the observation must be explained, i.e., at least one symbol in each axiom is specified as non-abducible. As the proportion of non-abducible symbols increases, the number of symbols to be forgotten is increased, which is also expected to impact the total number of inferences required to compute the hypothesis.

The effect of these changes are investigated with respect to the runtime statistics and statistics regarding the structure of the hypotheses obtained. Both of these sets of statistics are presented for two hypotheses:

- **Initial Hypothesis**: refers to the hypothesis obtained by simplifying then negating the result of performing all inferences under the calculus in 7.3, restricted using the set-of-support inspired strategy to avoid unnecessary inferences. This hypothesis obtained at this point can be seen as an approximation of the hypothesis satisfying Definition 7.3.1, since it is not guaranteed to satisfy conditions (i) or condition (iii): consistency and independence of the explanations (disjuncts).

- **Independent Explanations**: refers to the hypothesis obtained by checking each disjunct in the initial hypothesis for the independence constraint in Definition 7.3.1 condition (iii). This check is performed using an external reasoner, in this case HermiT [GHM+14]. The result is a hypothesis that is a space of independent explanations for the given observation.

For these experiments, a separate consistency check $\mathcal{K}, \mathcal{H} \not\models \bot$ was not performed over the initial hypothesis. Therefore, the consistency of the initial hypothesis is not guaranteed. However, the aim during the experiments is for filtering to be applied to check the independence of each disjunct. Since this also eliminates inconsistent hypotheses, the separate check for the initial hypothesis was deemed unnecessary. This also enables an assessment of the computational cost and effectiveness, in eliminating redundant explanations, of the filtering step as well as the initial forgetting procedure.

As a result of the approach taken to generating observations, inconsistent observations were permitted only in a minority of cases. For the TBox observations, no inconsistent observations were permitted. For the KB observations, only 0–1% of observations were inconsistent. Finally, for the ABox observations 4%–21% of the observations were inconsistent, where the latter bound corresponds to experiment 3:

observations consisting of 11–20 axioms. The differences in the proportions of inconsistent observations is a consequence of the way in which observations were generated: selection from the background ontology for TBox axioms, and random generation for ABox axioms.

Note that the steps to compute the initial hypothesis and to filter to obtain independent explanations are performed sequentially. The time limit for each experiment was set to a total of 10 minutes for each abduction problem. All of the experiments were performed on a machine using a 3.10GHz Intel Core i5-2400 CPU and 8GB of RAM.

Evaluation of the results will be divided into two perspectives: the runtime and the structure of the hypotheses obtained.

7.6.4 Runtime

In this section, statistics for the times taken to compute solutions to the generated abduction problems are presented. The statistics regarding runtimes were calculated over the set of completed abduction problems, i.e., excluding timeout cases. The success rate field indicates the proportion of cases that succeeded before the 10 minute limit was reached and without any memory issues.

It is worth noting that the filtering time in each case depends heavily on the size of the initial hypotheses: the more disjuncts there are in the initial hypothesis, then the more entailment checks are made during the filtering step. Therefore, the runtime statistics for computing *independent explanations* will be discussed with reference to the number of explanations (disjuncts) in the hypotheses produced.

Table 7.2 shows the runtime results for experiment 1. The results for computing the initial hypothesis indicate that, in the majority of cases for ABox, TBox and KB observations, the forgetting step took a matter of seconds to complete as indicated by the median values. The mean and 90^{th} percentile run times indicate that, for a smaller proportion of difficult cases, the forgetting step took far longer to complete. This may be due to the presence of commonly occurring symbols in the set of non-abducibles, which may lead to a higher number of inferences. The success rates for computing the

Computation of Initial Hypothesis (sec)

Observation	Min.	Median	Mean	90^{th} Percent.	Max.	Success Rate (%)
ABox	0.2	4.4	54.4	192.1	591.9	92.7
TBox	0.3	5.8	53.2	177.6	593.3	81.9
KB	0.3	6.5	58.8	197.2	596.3	86.1

Filtering for Independent Explanations (sec)

Observation	Min.	Median	Mean	90^{th} Percent.	Max.	Success Rate (%)
ABox	0.0	0.1	5.3	4.3	533.9	91.0
TBox	0.0	0.7	21.1	59.9	553.2	74.7
KB	0.0	0.2	11.2	9.1	584.4	83

Table 7.2: Runtime statistics for abduction over observations of size 1–10 axioms, with 1–30% of background symbols non-abducible. Time limit was set to 10 minutes.

Computation of Initial Hypothesis (sec)

Observation	Min.	Median	Mean	90^{th} Percent.	Max.	Success Rate (%)
ABox	0.2	5.1	57.4	198.1	594.7	87.4
TBox	0.3	8.4	62.3	219.3	579.9	70.2
KB	0.3	9.2	68.8	242.4	596.3	77.1

Filtering for Independent Explanations (sec)

Observation	Min.	Median	Mean	90^{th} Percent.	Max.	Success Rate (%)
ABox	0.0	0.0	4.7	1.6	562.1	86.6
TBox	0.0	0.4	14.8	20.6	579.9	66.1
KB	0.0	0.1	7.4	4.4	590.5	74.9

Table 7.3: Runtime statistics for abduction over observations of size 1–10 axioms, with 31–60% of background symbols non-abducible. Time limit was set to 10 minutes.

Computation of Initial Hypothesis (sec)

Observation	Min.	Median	Mean	90^{th} Percent.	Max.	Success Rate (%)
ABox	0.3	23.4	96.1	331.1	596.3	64.9
TBox	0.7	19.9	74.6	243.8	587.3	58.1
KB	0.4	30.0	99.5	324.9	596.5	57.4

Filtering for Independent Explanations (sec)

Observation	Min.	Median	Mean	90^{th} Percent.	Max.	Success Rate (%)
ABox	0.0	0.0	0.8	0.1	211.3	64.3
TBox	0.0	0.4	9.0	7.0	580.3	53.3
KB	0.0	0.0	3.0	1.1	472.9	56

Table 7.4: Runtime statistics for abduction over observations of size 11–20 axioms, with 1–30% of background symbols non-abducible. Time limit was set to 10 minutes.

initial hypothesis indicate that abduction problems for which the observation contained only TBox axioms were the most difficult to complete in the allotted time, followed by KB observations and then ABox observations. This may indicate that the presence of TBox axioms in the observation increases the time taken for the forgetting step, compared to ABox axioms.

In most cases, the filtering step took under a second to complete for each type of observation. The proportion of non-empty initial hypotheses, shown in column 1 of Table 7.5, likely explains the easiest cases. For TBox observations, a higher proportion of the initial hypotheses were non-empty: 94.3% as opposed to 43.0% for ABox and 58.1% for KB observations respectively. This may explain the median filtering time for TBox observations, which is the highest of the three. The number of disjuncts in the non-empty hypotheses also impacts the filtering time, since each additional disjunct in the initial hypothesis is another entailment check that must be made. As shown in Table 7.6, in most cases ABox observations resulted in hypotheses with more explanations. ABox observations also resulted in the largest hypotheses overall, as indicated by the 90$^{\text{th}}$ percentile and maximum values. Despite this, the time taken to filter for independent explanations is lowest for ABox observations, both in most cases and in the longest cases, followed by the KB observations and finally the TBox observations. This may indicate that the presence of TBox axioms in the explanations has a larger effect on the filtering time taken than the number of disjuncts. This may be due to the difficulty of entailment checking over TBoxes as opposed to ABoxes.

The success rates within the 10 minute time limit are also lower when the filtering step is performed in addition to forgetting, as to be expected, particularly if the forgetting step takes the majority of the 600 second time limit. This difference in success rates is most noticeable for abduction problems with TBox observations: a decrease of 7.2% compared to 1.7% and 3.1% for the ABox and KB observations respectively. This likely reflects the effect noted above.

Table 7.3 shows the results for experiment 2, in which the proportion of non-abducibles was increased from 1–30% to 31–60% over the previous experiment. The results show that the time taken to compute an initial hypothesis increased across all observation types, both in terms of the most common run times and the most difficult cases. This is also reflected in the success rates, which decreased by 5.3%, 11.7% and 9% compared to the results in experiment 1 for abduction problems with ABox, TBox and KB observations respectively. This is likely due to an increase in the number of inferences required to eliminate the additional non-abducibles using the calculus in Figure 7.3. The drop in success rates compared to experiment 1 was most significant over the TBox observations, followed by KB observations, indicating that the effect of increasing the number of non-abducibles is more costly proportionally for observations containing TBox axioms.

The reduction in success rates when additionally performing the filtering step was 0.8%, 4.1% and 2.2% respectively. The reduction in 10 minute success rates when filtering, compared to the corresponding results in experiment 1, are similar: the cost of filtering is most pronounced for TBox observations, then KB and finally ABox observations. This is also reflected in the comparative proportion of non-empty initial hypotheses for each type of observation, shown in column 2 of Table 7.5, which is similar to experiment 1. However, as before, the hypotheses produced for TBox observations generally contained fewer explanations than those produced for KB or ABox observations, indicating that the presence of TBox axioms has a more significant impact on the cost of the filtering step.

The magnitude of the decrease in success rate when filtering for independent explanations is lower across all three cases for experiment 2, though this is likely due to the additional cost incurred during the forgetting step. Since the success rates are already lower when computing the initial hypothesis, the difference in success rate when performing filtering in addition may be less significant.

Table 7.4 shows the results for experiment 3, which examines the effect of increasing the number of axioms in the observations from 1–10 to 11–20, with 1–30% of symbols specified as non-abducible as in experiment 1. The results indicate that increasing the observation size increased the time taken to compute a solution across all three abduction types, as expected. Compared to increasing the proportion of non-abducibles as in experiment 2, increasing the size of the observation had a larger impact on the time taken to compute a solution. This is true for the majority of cases, as indicated by the median values which are higher than the corresponding results for both of the previous experiments. The most difficult problems follow a similar trend. The mean and 90$^{\text{th}}$ percentile values across ABox, TBox and KB cases all increased compared to the previous two experiments, while the success rates decreased by 27.8%, 23.8% and 28.7% across ABox, TBox and KB observations respectively compared to experiment 1. This is likely due to the assumption that each axiom in the set of observations should be explained, i.e., there is at least one non-abducible symbol in the signature of each axiom. As a result, the number of inferences under the set-of-support inspired approach will likely increase as the number of axioms in the observations increases, since inferences are focused on axioms in the observation.

The times taken for the filtering step did not increase as significantly. In addition, the decrease in success rate when filtering for independent explanations was similar to experiments 1 and 2: a decrease of 0.6%, 4.8% and 1.4% for the ABox, TBox and KB observations respectively. However, across all three observation types the 90$^{\text{th}}$ percentile times are lower, i.e., there were fewer cases for which the filtering step took a significantly longer time. This indicates that the effect of increasing observation size was not as pronounced with respect to the time taken to filter the initial hypothesis as it was to compute it. In the ABox abduction case, this may be due to the fact that only 7.4% of the initial hypotheses were non-empty, as shown in column 3 of Table 7.5. In other cases, this may be due to the lower success rates in the computation of the initial hypothesis. It is possible that computing the initial hypothesis for the more difficult abduction problems had mostly already exceeded ten minutes, leaving fewer difficult cases during the filtering step.

Cases with Non-Empty Initial Hypothesis %

Observation Type	Ψ axioms: 1–10, $sig(\mathcal{O}) \setminus \mathcal{S}_A$: 1–30%	Ψ axioms: 1–10 $sig(\mathcal{O}) \setminus \mathcal{S}_A$: 31–60%	Ψ axioms: 11–20 $sig(\mathcal{O}) \setminus \mathcal{S}_A$: 1–30%
ABox	43.0	28.2	7.4
TBox	94.3	83.2	80.8
KB	58.1	45.3	59.1

Cases with At Least One Independent Explanation %

Observation Type	Ψ axioms: 1–10, $sig(\mathcal{O}) \setminus \mathcal{S}_A$: 1–30%	Ψ axioms: 1–10 $sig(\mathcal{O}) \setminus \mathcal{S}_A$: 31–60%	Ψ axioms: 11–20 $sig(\mathcal{O}) \setminus \mathcal{S}_A$: 1–30%
ABox	36.1	23.2	4.2
TBox	67.4	41.3	39.1
KB	43.8	32.0	43.0

Table 7.5: Percentage of hypotheses that were non-empty. Ψ axioms: total axioms in the observation. $sig(\mathcal{O}) \setminus \mathcal{S}_A$: percentage of symbols that were non-abducible.

7.6.5 Hypothesis Structure

This section discusses the results regarding the structure of the hypotheses obtained for experiments 1, 2 and 3 above. Since the observations and signature of abducibles are selected randomly, it is expected that in many cases there may be no suitable hypothesis satisfying Definition 7.3.1 in the given signature. Therefore, only results regarding the non-empty hypotheses, the proportion of which are shown in Table 7.5, are used to calculate statistics regarding hypothesis structure.

The results in Table 7.5 indicate that the filtering process eliminated a significant number of "false" initial hypotheses, i.e., those that did not contain an explanation for the observation. These cases are abduction problems for which no valid explanation existed within the given signature of abducibles. This can be seen by the decreases in the proportion of non-empty hypotheses post filtering for each set of abduction problems. This was expected, since the set-of-support inspired approach to restricting inferences during the forgetting step is not guaranteed to avoid all inferences that are redundant with respect to the abduction problem. The number of false candidate hypotheses eliminated by the filtering procedure was highest for the TBox observations, for which the proportion of non-empty hypotheses decreased by 26.9%, 41.9%

Number of Explanations (disjuncts)

	Initial Hypothesis					Independent Explanations				
Obs.	Min.	Mdn	Mean	P90	Max.	Min.	Mdn	Mean	P90	Max.
ABox	1	2	11.0	18	576	1	1	5.1	8	576
TBox	1	1	1.6	2	113	1	1	1.1	1	18
KB	1	1	4.4	8	432	1	1	2.8	5	160

Table 7.6: Number of explanations in hypotheses for observations of size 1–10 axioms, with 1–30% of background symbols non-abducible.

Number of Explanations (disjuncts)

	Initial Hypothesis					Independent Explanations				
Obs.	Min.	Mdn	Mean	P90	Max.	Min.	Mdn	Mean	P90	Max.
ABox	1	1	5.1	9	320	1	1	2.4	5	70
TBox	1	1	2.4	3	101	1	1	1.2	1	25
KB	1	1	4.4	8	216	1	1	2.5	4	105

Table 7.7: Number of explanations in hypotheses for observations of size 1–10 axioms, with 31–60% of background symbols non-abducible.

Number of Explanations (disjuncts)

	Initial Hypothesis					Independent Explanations				
Obs.	Min.	Mdn	Mean	P90	Max.	Min.	Mdn	Mean	P90	Max.
ABox	1	12	59.0	141	1000	1	8	20.4	48	220
TBox	1	1	3.3	4	162	1	1	1.6	2	120
KB	1	2	8.8	18	440	1	2	6.4	15	80

Table 7.8: Number of explanations in hypotheses for observations of size 11–20 axioms, with 1–30% of background symbols non-abducible.

and 41.7% for experiments 1, 2 and 3 respectively. This indicates that the number of redundancies remaining after the set-of-support inspired forgetting strategy may be higher in general when there are TBox axioms in the observation.

Table 7.6 provides results regarding the number of explanations (disjuncts) in non-empty hypotheses for experiment 1. The hypotheses produced to explain ABox observations contained the most disjuncts, both before and after performing filtering for independent explanations. This is particularly true for the largest hypotheses, as illustrated by the 90$^{\text{th}}$ percentile. In addition, the hypotheses for KB observations generally contained more explanations than those for purely TBox observations. These results indicate that it is more likely to obtain a disjunctive hypothesis, i.e., multiple explanations for an ABox observation than a TBox observation. Across all three observation types, the filtering process removed a number of redundant explanations. This is particularly true for the larger hypotheses in each category, as indicated by the decrease in the means and 90$^{\text{th}}$ percentiles in each case. The effect of filtering on the number of explanations was most pronounced over the ABox observation cases, indicating a higher number of redundant explanations in some of the larger hypotheses. Of all non-empty initial hypotheses, 54.7%, 10.7% and 41.4% were disjunctive for the ABox, TBox and KB observations respectively. For the independent explanations, the corresponding values were 36.2%, 3.3% and 23.4%. This further supports the notion that there are fewer disjunctive hypotheses for TBox observations, and that filtering removes a significant number of redundant explanations from the initial hypotheses.

The corresponding results for experiment 2 are shown in Table 7.7. The results show a similar trend to experiment 1: the largest hypotheses for ABox observations generally contained more disjuncts than the largest hypotheses for TBox and KB observations, supporting the possibility that disjunctive hypotheses are less common for TBox observations. However, the number of explanations in the hypotheses for ABox observations decreased with the increase in the proportion of non-abducible symbols. This may be due to the fact that ABox axioms are weaker than TBox axioms. Thus, with a less restrictive signature, more avenues of explanation may exist for ABox observations initially, resulting in more avenues of explanation being eliminated as the signature of symbols that can be used in explanations is restricted. Of all non-empty initial hypotheses, 47.7%, 25.3% and 45.3% were disjunctive for the ABox, TBox and KB observations respectively. For the independent explanations, the corresponding values were 28.1%, 6.6% and 22.8%. This also indicates a similar trend to the results observed in the previous experiment.

Table 7.8 shows the corresponding results for experiment 3. The results indicate that increasing the number of axioms in the observation increased the number of disjuncts in the hypotheses across all three observation types. This effect was most significant for ABox observations, followed by KB and finally TBox observations. For ABox observations, this effect was observed across most of the abduction problems, as indicated by the increase in the median value, while more hypotheses with a large number of explanations were also produced as indicated by the 90^{th} percentile and maximum results. A significant portion of these additional explanations were removed as redundant during filtering, as indicated by the corresponding values for the independent explanations columns. However, the number of independent explanations was still significantly higher for ABox observations than in experiments 1 and 2. This may be due to there being multiple ways of explaining a number of the individual ABox axioms in each observation, resulting in a large number of possible combinations of explanations in the final hypothesis for the full set of observations. Meanwhile for KB and TBox abduction the effect was less apparent in most cases, being significant more so in the cases with the largest number of explanations. This likely reflects the same effect noted above: it may be that it is less likely for there to be multiple explanations, i.e. a disjunctive hypothesis, for TBox axioms as opposed to ABox axioms. Of all non-empty initial hypotheses, 91.4%, 25.9% and 61.4% were disjunctive for the ABox, TBox and KB observations respectively. For the independent explanations, the corresponding values were 50.3%, 6.0% and 26.2%. Again, this follows a similar trend to the previous experiments. However, the proportion of disjunctive hypotheses decreased more significantly across all three types of observation, indicating a higher proportion of redundant explanations as the number of axioms in the observations increased.

Table 7.9 summarises the proportion of the initial hypotheses, and hence forgetting solutions obtained in Step (2), that contained each of the additional language features: fixpoints, nominals and inverse roles. Inverse roles were not utilised in any of the cases observed across all three experiments. Generally, nominals were the most commonly used of the three additional language features, while fixpoints occurred more

Experiment		Proportion of Initial Hypotheses Containing (%)		
		Fixpoints	Nominals	Inverse Roles
1	ABox	2.2	10.8	0.0
	TBox	1.9	13.3	0.0
	KB	5.8	15.3	0.0
2	ABox	3.5	8.1	0.0
	TBox	3.2	13.8	0.0
	KB	9.0	16.1	0.0
3	ABox	14.1	20.9	0.0
	TBox	6.3	23.1	0.0
	KB	20.3	30.6	0.0

Table 7.9: Proportion of initial hypotheses containing at least one occurrence of fixpoints, nominals and inverse roles.

frequently for KB observations. The proportions of fixpoints and nominals were higher across experiment 3, for which the number of observations was increased, though this may be due to the smaller sample size of non-empty hypotheses for this experiment.

The hypothesis characteristics for experiments 1, 2 and 3 are shown in Tables 7.10, 7.11 and 7.12 respectively. The results indicate that for ABox observations, there exist a minority of cases for which it is possible to have a TBox hypothesis. In most cases, these TBox axioms were redundant as illustrated by the decrease in the mean, P90 and maximum values for the number of TBox axioms across all three ABox observation experiments. This is to be expected, since the semantic minimality constraint ensures that TBox hypotheses are only produced for ABox observations when absolutely necessary, meaning that a non-redundant TBox explanation is unlikely in this scenario. A similar result can be observed for TBox observations: there are a minority of cases for which ABox assertions are produced as part of an explanation and in most, though not all, cases these are redundant in the final set of independent explanations. As expected, KB observations generally resulted in hypotheses containing a split of both TBox and ABox axioms. In general, increasing the size of the observation had a larger impact on the number of axioms present in the hypotheses than increasing the proportion of non-abducible symbols.

Initial Hypothesis

	TBox Axioms					ABox Axioms				
Obs.	Min.	Mdn	Mean	P90	Max.	Min.	Mdn	Mean	P90	Max.
ABox	0	0	1.3	0	360	0	6	63.9	108	4620
TBox	0	5	6.1	10	104	0	0	0.5	0	208
KB	0	3	9.8	16	1408	0	2	14.4	21	2440

Independent Explanations

	TBox Axioms					ABox Axioms				
Obs.	Min.	Mdn	Mean	P90	Max.	Min.	Mdn	Mean	P90	Max.
ABox	0	0	0.2	0	72	0	4	28.9	40	4608
TBox	0	4	4.7	9	96	0	0	0.0	0	18
KB	0	2	6.1	12	336	0	2	8.7	14	684

Table 7.10: Number of axioms in hypotheses for observations of size 1–10 axioms, with 1–30% of background symbols non-abducible.

Initial Hypothesis

	TBox Axioms					ABox Axioms				
Obs.	Min.	Mdn	Mean	P90	Max.	Min.	Mdn	Mean	P90	Max.
ABox	0	0	1.8	2	640	0	4	24.8	36	3104
TBox	0	4	10.3	16	827	0	0	0.5	0	101
KB	0	2	10.4	20	504	0	2	10.9	16	1164

Independent Explanations

	TBox Axioms					ABox Axioms				
Obs.	Min.	Mdn	Mean	P90	Max.	Min.	Mdn	Mean	P90	Max.
ABox	0	0	0.3	1	12	0	3	9.4	17	449
TBox	0	3	4.7	8	158	0	0	0.0	0	12
KB	0	2	6.1	10	420	0	2	7.0	10	710

Table 7.11: Number of axioms in hypotheses for observations of size 1–10 axioms, with 31–60% of background symbols non-abducible.

Initial Hypothesis

	TBox Axioms					ABox Axioms				
Obs.	Min.	Mdn	Mean	P90	Max.	Min.	Mdn	Mean	P90	Max.
ABox	0	0	6.0	4.8	448	0	114	755.8	1997	15960
TBox	0	15	31.9	40	2641	0	0	1.2	0	150
KB	0	18	58.8	122	1848	0	6	43.3	83	1672

Independent Explanations

	TBox Axioms					ABox Axioms				
Obs.	Min.	Mdn	Mean	P90	Max.	Min.	Mdn	Mean	P90	Max.
ABox	0	0	0.8	0	20	0	88	261.4	590	2920
TBox	0	14	21.2	20	2024	0	0	0.2	0	13
KB	0	17	50.6	116	1024	0	6	33.6	81.5	864

Table 7.12: Number of axioms in hypotheses for observations of size 11–20 axioms, with 1–30% of background symbols non-abducible.

Chapter 8

Utilising the Abduction Approaches in Practice

The focus in this chapter is on discussing the use of the hypotheses produced by the signature-based abduction approaches in this thesis and the connection between these approaches and related problems such as induction and use of data. The use of abductive reasoning to produce disjunctive hypotheses, in this case spaces of independent explanations, has not been well investigated in terms of applications. The effect of these hypotheses, as opposed to computing individual explanations as conjunctions without the presence of disjunction, is not yet well understood.

Here, the aim is to propose promising directions for utilising these disjunctive hypotheses in practice by examining and discussing how they may be integrated into a knowledge base and by examining examples from related reasoning problems including concept learning and explanatory induction.

8.1 Disjunctive Hypotheses

The approaches in this thesis utilise forgetting with efficient filtering methods to produce hypotheses in the form of a space, i.e., disjunction of independent explanations.

This perspective, in which the aim is to compute a disjunction of explanations satisfying given constraints, is generally less common than the perspective of abduction producing a collection of individual (conjunctive) explanations.

However, several works on abduction in A.I. do focus on the aim of computing disjunctive hypotheses. In Lin [Lin01], abduction in propositional logic is viewed from the perspective of weakest sufficient conditions, which are noted to be equivalent to a disjunction of explanations, leading to the aim of capturing all possible explanations rather than just an individual explanation. The connection between second-order quantifier elimination (SOQE) and abduction [DLS01, GSS08] can also be seen as a form of disjunctive abduction (with redundancies), since SOQE can be used to compute weakest sufficient conditions for abduction. As discussed in Chapter 4, forgetting, and thus the abduction approach in this thesis, is related to the problem of SOQE. Therefore, the result of forgetting can be viewed as a strongest necessary condition as in Theorem 3.6.1. As a result, the hypotheses in this thesis obtained by contraposition on the forgetting solution also share the aim of capturing all possible explanations, where the additional need to impose an independence criterion on each of the explanations obtained by forgetting has been motivated and addressed.

Konolige [Kon92] defines the notion of a cautious explanation as a disjunction of the subset-minimal abductive explanations for an observation, and discusses the notion of independence between the individual explanations that is also specified in this work.

In the DL setting, however, the problem of computing disjunctive hypotheses has received little attention and, to the best of the author's knowledge, there are as yet no other methods for computing disjunctions of independent explanations over DL ontologies. In fact, in many cases disjunction is excluded from the hypotheses produced [KES11, HB12, HBK14]. This may be due to the difficulty of computing these disjunctive hypotheses in the first place, particularly in the presence of the criteria of semantic minimality and independence of explanations, which is problematic in the DL setting due to the emphasis on tractability of reasoning even over large ontologies. Additionally, there is a need for further investigation into the effects and utility of disjunctive hypotheses in practice, as this has yet to receive significant formal investigation in the existing literature on abduction [IS19] particularly in the DL setting.

Therefore, identifying promising directions for utilising the hypotheses produced by the approaches proposed in this work is important, particularly in the DL setting. This also relates to issues such as the selection of an appropriate forgetting signature, for which there is a need to devise automated approaches. The focus here is on providing suggestions on how these issues may be addressed in practice and on how the disjunctive hypotheses relate and provide utility to several tasks in the setting of DLs.

8.2 Forgetting and Hypothesis Refinement

When presented with a disjunctive hypothesis to explain a new observation, it is unlikely that the disjunction itself will be added to the background knowledge, since there is a degree of uncertainty associated with which of the given explanations is the most useful or "correct" in the given circumstance. Therefore, it is necessary to perform refinement in order to select one, or several, of the individual explanations that compose the entire disjunctive hypothesis. Given that the disjunctive hypothesis satisfies the independence constraint, it can be assumed that any of the available explanations would at least be consistent and in some sense unique.

When presented with a semantically minimal, i.e., least assumptive hypothesis, it may be useful to seek stronger hypotheses, using the initial hypothesis as a starting point. This may be desirable if the initial weaker hypothesis is too vague to provide useful insight. For example, in tasks such as repair, the aim may be to add a hypothesis that is as informative as possible [LWKDI13].

Another consideration is the choice of abducibles, which determines the hypothesis obtained. For the approaches presented in this thesis, the non-abducibles are specified as part of the forgetting signature, i.e., the symbols to be eliminated in Step (2) of Figures 5.1 and 7.2. The task of choosing the set of abducible symbols, and thus the forgetting signature, can be difficult in practice. In several forgetting tasks, it is often assumed that the user may, to an extent, already know the signature of symbols that they wish to exclude from the given knowledge base. For example, when computing the logical difference [ZAS+19] using forgetting the aim is to compute entailments within the common signature of two ontologies, which provides a natural way to specify the forgetting signature as the symbols outside of this common signature.

In the abduction setting, this choice may often not be as clear. Observations may describe previously unseen phenomena or data, and there may be no existing expectation of the signature of symbols that should be used to explain them. Even in tasks such as diagnosis, where the signature of abducibles might be restricted to causes, it may be desirable to find ways to guide further restrictions to the set of abducibles based upon finding increasingly stronger explanations for the given observation.

Therefore, it is important to devise strategies to guide the selection of the forgetting signature. Additionally, if these strategies can be automated or partially automated, then this may be beneficial in providing a default framework with which to apply abduction approaches that utilise forgetting in practice. The perspective taken here is that one natural approach to the selection of the forgetting signature ties directly to hypothesis refinement. Hypothesis refinement is particularly important in this setting due to two characteristics of the hypotheses produced: they are a set of independent explanations and they are semantically minimal.

8.2.1 Iterative Abduction

The hypothesis refinement task can be viewed as a search problem through the space of possible hypotheses. An approach for refining hypotheses in forgetting-based abduction suggested here will be referred to as *iterative abduction*, and can be seen as a form of tree search over the space of possible explanations. Here the approach in Chapter 5, specifically the approach in Figure 5.1, will be used to illustrate this approach.

The following notation will be used during this section for each step of the suggested iterative approach, viewing the approach as a tree search:

- d will be used to denote the depth of the search.

- N^d will be used to denote the number of nodes at depth d.
- $\mathcal{F}_i^d$ denotes the i^{th} forgetting signature used to progress from depth $d-1$ to depth d, where $1 \leq i \leq N^d$.
- A forgetting solution $\mathcal{V}_i^d$ is the result of forgetting the signature $\mathcal{F}_i^d$ from a previous forgetting solution $\mathcal{V}_j^{d-1}$, where $1 \leq j \leq N^{d-l}$.
- The hypothesis $\mathcal{H}_i^d$ denotes the corresponding hypothesis extracted from $\mathcal{V}_i^d$ during Steps (3) and (4) of Figure 5.1.

Effectively, each node in the search can be viewed as the pair $\{\mathcal{V}_i^d, \mathcal{H}_i^d\}$. The root node can be viewed as the pair $\{\mathcal{O}, \psi\}$, which would be the corresponding results with an empty forgetting signature. A key aspect of forgetting that lends itself well to an iterative approach such as this is the following:

Corollary 8.2.1. *Let $\mathcal{V}$ be the result of forgetting a signature $\mathcal{F}$ from an ontology $\mathcal{O}$, $\mathcal{V}'$ be the result of forgetting a signature $\mathcal{F}'$ from $\mathcal{O}$ and $\mathcal{V}''$ be the result of forgetting $\mathcal{F} \cup \mathcal{F}'$ from $\mathcal{O}$. (1) The result of forgetting $\mathcal{F}'$ from $\mathcal{V}$ is equivalent to $\mathcal{V}''$. (2) The result of forgetting $\mathcal{F}$ from $\mathcal{V}'$ is equivalent to $\mathcal{V}''$.*

The notion behind Corollary 8.2.1 is that forgetting can be applied iteratively. By retaining the forgetting solution of a previous step, the computation of inferences need only be performed when necessary, and only once. By remembering previously used forgetting signatures, the fact that forgetting leads to the same result irrespective of order given the same signatures provides a way to limit the search space: provided that the total signature of symbols forgotten has already been covered elsewhere in the search, there is no need to perform the step.

As before, the process starts with $(\mathcal{O}, \neg\psi)$. From here, the search space is initialised by selecting a starting signature of non-abducibles to obtain an initial hypothesis via the approach in Figure 5.1. For the most general case, it is assumed that there is no prior knowledge that enables the selection of this starting signature. In this case, a natural starting point for specifying the set of abducibles for the abduction approaches in this work is based upon the signature of symbols in the observation ψ itself. This is due to the fact that the abduction approaches presented in this thesis are goal-oriented, i.e., during Step (2), inferences are based around eliminating symbols in the set of non-abducibles and the set of relevant inferences is based around the negated observation. It should be the case that at least one symbol in the observation ψ is specified as non-abducible. Otherwise, the semantically minimal hypothesis required under Definition 4.2.1 will simply be the observation ψ itself, as discussed in Chapter 5.

Therefore, the suggested first step is to compute a set of forgetting solutions: one for each unique forgetting signature $\mathcal{F}_i^1$ containing a single symbol from $sig(\psi)$. This results in the set of strongest possible forgetting solutions $\mathcal{V}_i^1$, i.e. those that preserve the most entailments of $(\mathcal{O}, \neg\psi)$. By repeating Steps (3) and (4) of Figure 5.1 for these forgetting solutions, a set of corresponding semantically minimal hypotheses $\mathcal{H}_i^1$ are obtained. Each pair $\{\mathcal{V}_i^1, \mathcal{H}_i^1\}$ can be seen as a node of the tree, where $1 \leq i \leq N^l$ and N^l is the number of nodes at depth l. To illustrate, consider the following scenario concerning a simplified view of a reptile habitat domain:

Example 8.2.1. *Let the background knowledge be the following ontology:*

$$
\begin{aligned}
\mathcal{O} = \{ & PineWoods \sqsubseteq Woodlands, \\
& Woodlands \sqsubseteq Habitat, \\
& Arid \sqcap Tropical \sqsubseteq \bot \\
& Pogona \sqcap Iguana \sqsubseteq \bot, \\
& Iguana \sqsubseteq \exists livesIn.(Woodlands \sqcap Tropical), \\
& GreenIguana \sqsubseteq Iguana, \\
& Pogona \sqsubseteq \exists livesIn.(Woodlands \sqcap Arid), \\
& PogonaMinor \sqsubseteq Pogona, \\
& PogonaMajor \sqsubseteq Pogona, \\
& PogonaMinorMinima \sqsubseteq PogonaMinor\}
\end{aligned}
$$

for the following observation concerning an unknown specimen:

$$\psi = \exists livesIn.Woodlands(specimen1)$$

if the signature of non-abducibles is $\mathcal{F} = \{\emptyset\}$*, then no inferences are made and the initial hypothesis is simply* ψ*. If instead the set of non-abducibles is* $\mathcal{F}_1^1 = \{Woodlands\}$*, the following forgetting solution* $\mathcal{V}_1^1$ *is obtained in Step (2) of Figure 5.1:*

$$\begin{aligned}
\mathcal{V}_1^1 = \{&Arid \sqcap Tropical \sqsubseteq \bot,\\
&Pogona \sqcap Iguana \sqsubseteq \bot,\\
&PineWoods \sqsubseteq Habitat,\\
&Iguana \sqsubseteq \exists livesIn.(Habitat \sqcap Tropical),\\
&GreenIguana \sqsubseteq Iguana,\\
&Pogona \sqsubseteq \exists livesIn.(Habitat \sqcap Arid),\\
&PogonaMinor \sqsubseteq Pogona,\\
&PogonaMajor \sqsubseteq Pogona,\\
&PogonaMinorMinima \sqsubseteq PogonaMinor,\\
&(\neg Iguana \sqcup \ell)(specimen1),\\
&(\neg Pogona \sqcup \ell)(specimen1),\\
&\forall livesIn.\neg PineWoods(specimen1)\}
\end{aligned}$$

and the corresponding hypothesis $\mathcal{H}_1^1$ *is returned using Steps (3) and (4) of Figure 5.1:*

$$\mathcal{H}_1^1 = Pogona(specimen1) \vee \exists livesIn.PineWoods(specimen1) \vee Iguana(specimen1)$$

for $\mathcal{F}_2^1 = \{livesIn\}$*, the following hypothesis is obtained:*

$$\mathcal{H}_2^1 = Pogona(specimen1) \vee Iguana(specimen1)$$

leaving three independent explanations for the given observation.

However, when performing abduction in the presence of large background ontologies where the signature of symbols is also likely to be large, specifying only one symbol as non-abducible does not provide much of a restriction to the set of explanations produced. These explanations will be the weakest possible ones. It may be desirable to further refine these hypotheses to seek more specific ones. This could be done exhaustively using a depth-first or a breadth-first approach.

Example 8.2.1 (Continued)**.** *Here a depth-first approach is followed. Continuing with Example 8.2.1, assume that Pogona is determined to be the most promising explanation in the previous step. Therefore, the aim here is to refine this hypothesis. By forgetting* $\mathcal{F}_1^2 = \{Pogona\}$ *from* $\mathcal{V}_1^1$*, the following forgetting solution* $\mathcal{V}_1^2$ *is obtained:*

$$\begin{aligned}
\mathcal{V}_1^2 = \{&Arid \sqcap Tropical \sqsubseteq \bot,\\
&Pogona \sqcap Iguana \sqsubseteq \bot,\\
&PineWoods \sqsubseteq Habitat,\\
&Iguana \sqsubseteq \exists livesIn.(Habitat \sqcap Tropical),\\
&GreenIguana \sqsubseteq Iguana,\\
&PogonaMinor \sqsubseteq \exists livesIn.(Habitat \sqcap Arid),\\
&PogonaMajor \sqsubseteq \exists livesIn.(Habitat \sqcap Arid),\\
&PogonaMinorMinima \sqsubseteq PogonaMinor,\\
&(\neg Iguana \sqcup \ell)(specimen1),\\
&(\neg PogonaMinor \sqcup \ell)(specimen1),\\
&(\neg PogonaMajor \sqcup \ell)(specimen1),\\
&\forall livesIn.\neg PineWoods(specimen1)\}
\end{aligned}$$

for which the corresponding hypothesis is:

$$\mathcal{H}_1^2 = PogonaMinor(specimen1) \vee PogonaMajor(specimen1) \vee \ldots$$

$$...\exists livesIn.PineWoods(specimen1) \vee Iguana(specimen1)$$

Repeating this procedure, and forgetting $\mathcal{F}_1^3 = \{PogonaMinor\}$ from $\mathcal{V}_1^2$ would result in the hypothesis:

$$\mathcal{H}_1^3 = PogonaMinorMinima(specimen1) \vee PogonaMajor(specimen1) \vee ...$$
$$...\exists livesIn.PineWoods(specimen1) \vee Iguana(specimen1)$$

In Example 8.2.1, the other explanations in the hypothesis are retained while the explanation $Pogona(specimen1)$ is refined. If further information was gathered that invalidated these explanations, it would be possible to discard these explanations whilst refining the preferred one. For example, it could be the case that the specimen was confirmed to be a *Pogona*, leading to the choice to refine this explanation. This would exclude the explanation $Iguana$. Additionally, the user may decide that further refinements on the habitat would not provide an interesting explanation. In this case, the negations of these undesirable explanations could be removed from $\mathcal{V}_1^1$, meaning that the hypothesis during the second iteration would instead be $\mathcal{H}_1^1 = Pogona(specimen1)$, followed by $\mathcal{H}_1^2 = PogonaMinor(specimen1) \vee PogonaMajor(specimen1)$.

In the above instance, guidance on how to refine the hypothesis, and hence choose the forgetting signature, are driven by human involvement. This could alternatively be automatically guided, for example using external data. This is often the approach taken by approaches to the concept learning problem, which take sets of positive and negative examples over which coverage metrics are utilised to refine inductive generalisations.

8.3 Signature-Based Abduction and Concept Learning

This section explores the use of the abduction approaches in this thesis for the problem of concept learning. Concept learning is a common problem in the area of ontology learning, where the aim is to learn the definition of a concept with respect to a given background ontology. The learned definition takes the form of a potentially complex

DL concept C, which can then be added to the background ontology via a TBox axiom of the form $Target \sqsubseteq C$ or, as a stronger statement, $Target \equiv C$. The problem of concept learning in DL ontologies can be defined as follows [LH10]:

Definition 8.3.1. *Let C_T be a concept called the* Target, *$\mathcal{O}$ be background knowledge in the form of a DL ontology, E^+ and E^- be sets of positive and negative examples containing elements of the form $Target(a)$, where a is an individual $a \in N_I$. The concept learning problem is to compute a TBox axiom α of the form $C \equiv Target$, where C is a DL concept such that $Target \notin sig(C)$, such that $\mathcal{O}, \alpha \models E^+$ and $\mathcal{O}, \alpha \not\models E^-$*

Similar to the perspective taken in ILP, the problem is usually seen as a search problem through the quasi-ordered space of possible definitions for the target concept. A quasi-ordering is a reflexive and transitive relation, and a common quasi-ordering applied to the search space is subsumption ($\sqsubseteq$). As is commonly the case in ILP, refinement operators are used to traverse the quasi-ordered search space. Refinement operators can be defined as follows [LH10]:

Definition 8.3.2. *Let $(\mathcal{S}, \sqsubseteq)$ be a quasi-ordered space, where $\mathcal{S}$ is a set of concepts expressible in a language $\mathcal{L}$. A downward (w.r.t. upward) $\mathcal{L}$ refinement operator ρ is a mapping from $\mathcal{S}$ to $2^{\mathcal{S}}$ such that for all $C \in \mathcal{S}$, $C' \in \rho(C)$ implies $C' \preceq C$ ($C \preceq C'$). The concept C' is then called a generalisation (specialisation) of the concept C.*

To illustrate this problem, consider Example 8.3.1 which is based upon a modified version of an example from Ray [RBR03], translated to DLs.

Example 8.3.1. *Consider the following background knowledge, expressed as a DL ontology $\mathcal{O}$:*

$$\begin{aligned}\mathcal{O} = \{&Tired \sqcap Poor \sqsubseteq Sad,\\ &Lonely \sqsubseteq Sad,\\ &Tired(oli),\\ &Tired(ale),\end{aligned}$$

$$\begin{aligned} &Tired(kb),\\ &Lecturer(ale),\\ &Lecturer(kb),\\ &Student(oli)\} \end{aligned}$$

and the following set of positive and negative examples:

$$E^+ = \{Sad(ale), \quad\quad E^- = \{Sad(oli)\}$$
$$Sad(kb)\}$$

A possible inductive generalisation $\mathcal{H}_{Ind}$ that can be added to $\mathcal{O}$ such that $\mathcal{O}, \mathcal{H}_{Ind} \models E^+$ and $\mathcal{O}, \mathcal{H}_{Ind} \not\models E^-$ is:

$$\mathcal{H}_{Ind} = Lecturer \sqsubseteq Sad$$

The generalisation provided in Example 8.3.1 is also the shortest result (with 100% accuracy) returned using the DL-Learner system, treating the problem as a standard supervised learning problem This is also the result returned using the system DL-Learner [LH10], treating the problem as a standard supervised learning problem [BLW16].

As can be seen in the above example, the inductive generalisation provided makes little use of the background knowledge contained in $\mathcal{O}$: the concept symbols in the provided TBox axioms are not utilised in the generalisation produced. For example, rather than the relationship $Lecturer \sqsubseteq Sad$, one might expect that $Lecturer \sqsubseteq Poor$ or $Lecturer \sqsubseteq Lonely$ would also be possible generalisations since adding these to the background ontology $\mathcal{O}$ also leads to full coverage of the positive examples and none of the negative examples. However, since none of the individuals in the examples are specified to be elements of either of the concepts *Poor* or *Lonely*, these generalisations are unreachable without somehow enhancing the existing knowledge.

While abduction does not aim to produce generalisations, it has been suggested as a mechanism for making better use of background knowledge by producing a hypothesis to explain the data in E^+ and E^-. Several works have proposed that abduction can be used in this way [FK00b, RBR03, TNCKM06], enabling additional generalisations via inductive learning. A practical realisation of this notion in the setting of Inductive and Abductive Logic Programming is the system XHAIL [Ray09], which has been applied in the context of learning and revising metabolic networks represented as logic programs [RWK09]. XHAIL uses abduction over background knowledge to initialise a preliminary ground hypothesis, which is then generalised to provide an inductive hypothesis.

Such a realisation would also be useful in the context of induction and learning in DL ontologies. The abduction approaches developed in this thesis provide hypotheses that are a useful basis for the integration abduction and induction in DLs, due to the characteristics of the hypotheses produced. Using the approach in Chapter 5, Example 8.3.1 can be framed as an instance of the problem in Definition 5.1.1 as follows.

Example 8.3.1 (Continued). *Starting with the positive examples, the abduction problem can be specified as follows. Given the background ontology $\mathcal{O}$ and an observation $\psi \in E^+$, let the set of abducibles be $\mathcal{S}_A = \{Poor, Lonely\}$. The hypotheses satisfying Definition 5.1.1 obtained by applying the abduction approach in Chapter 5, for $\psi_1 = Sad(ale)$ and $\psi_2 = Sad(kb)$, are as follows:*

$$\mathcal{H}_1 = (Lonely \sqcup Poor)(ale)$$
$$\mathcal{H}_2 = (Lonely \sqcup Poor)(kb)$$

For the task of computing a generalisation $\mathcal{H}_{Ind}$ satisfying the concept learning problem, the following alternative datasets are provided based on the result of abduction:

$$E_1^+ = \{Lonely(ale), \quad Lonely(kb)\} \qquad E_1^- = \{Lonely(oli)\}$$
$$E_2^+ = \{Poor(ale), \qquad E_1^- = \{Poor(oli)\}$$

$$Poor(kb)\}$$

Using these datasets as input to subsequent rounds of induction, the following inductive generalisations can be reached:

$$\mathcal{H}_{Ind} = Lecturer \sqsubseteq Poor$$
$$\mathcal{H}_{Ind} = Lecturer \sqsubseteq Lonely$$

The benefit of the abduction approach in this context is in the form of the hypothesis produced. Each independent explanation for the given examples can be interpreted as a separate dataset over which generalisation can be performed. This enables a wider range of solutions to the concept learning problem that would not be reachable without this abductive step. In addition, the fact that each of these explanations is independent minimises the redundancy in the resulting generalisations. Without this criterion, it is possible in many cases to produce a number of generalisations that are redundant or equivalent to existing ones, unnecessarily increasing the size of the search space.

The above example shows how the ABox abduction approach in Chapter 5 can be applied to inductive problems, following the notion of abduction and induction operating in a cycle [FK00b]. An open question is how knowledge base abduction, as investigated in Chapter 7, and induction can be integrated successfully.

8.4 Knowledge-Base Abduction and Induction

Most existing work on abduction in DLs tackles the problems of TBox and ABox abduction entirely separately. Little work exists on the problem of KB abduction and the different forms this problem can take. Consequently, the effects and use of knowledge base abduction has not been investigated.

Abduction and explanatory induction have been identified as closely related forms of reasoning [Lac00, EKS06]. However, a particular characteristic of the KB abduction system in Chapter 7 provides a clear separation between the process of induction and the form of abduction considered here. For example:

Example 8.4.1. *Given the following abduction problem:*

$$\begin{aligned}\mathcal{O} &= \{B \sqsubseteq C,\\ &\quad D(a)\}\\ \psi &= \{C(a)\}\\ \mathcal{S}_A &= \{D, B\}\end{aligned}$$

In the general KB abduction case, possible consistent, explanatory hypotheses for ψ in $\mathcal{S}_A$ could include:

$$\begin{aligned}\mathcal{H}_1 &= B(a)\\ \mathcal{H}_2 &= D \sqsubseteq B\end{aligned}$$

the system discussed in this section will return $\mathcal{H}_1$, but not $\mathcal{H}_2$.

Due to the requirement of semantic minimality in Definition 7.3.1 condition (iv), the hypothesis $\mathcal{H}_2$ is not an acceptable solution since $\mathcal{O}, \mathcal{H}_2 \models \mathcal{H}_1$ but $\mathcal{O}, \mathcal{H}_1 \not\models \mathcal{H}_2$. Thus, the inability to return $\mathcal{H}_2$ is not problematic for the abduction problem discussed in Chapter 7. As given, there is no way to refine the hypothesis $\mathcal{H}_1$ to seek a stronger hypothesis of the form of $\mathcal{H}_2$: the only option to obtain a stronger hypothesis would be to further restrict the signature of abducibles $\mathcal{S}_A$, but this will not lead to such a generalisation in this case as if either B or D are removed from $\mathcal{S}_A$, $\mathcal{H}_2$ will be lost. In essence, this means that TBox hypotheses are produced only when required, i.e., a TBox hypothesis is the semantically minimal hypothesis for the given observation. As a result, the KB abduction approach in this work does not perform any explicit generalisation step from ground observations to universally quantified explanations, but can produce them when necessary. This is illustrated by the following case:

Example 8.4.2. *Given the following abduction problem:*

$$\mathcal{O} = \{C \sqsubseteq \forall r.F,$$
$$C(a)\}$$
$$\psi = \forall r.E(a)$$
$$\mathcal{S}_A = \{F, E\}$$

The KB abduction approach in Chapter 7 proceeds as follows:

Φ_B	Φ_S	
$bg1)\neg C(x) \vee \forall r.D_1(x)$	$1)\exists r.D_2(a)$	
$bg2)\neg D_1(x) \vee \neg F$	$2)\neg D_2(x) \vee \neg E(x)$	
$bg3)C(a)$	$3)\neg C(a) \vee \exists r.D_{12}(a)$	*Role Propagation(bg1, 1)*
	$4)\neg D_{12}(x) \vee D_1(x)$	
	$5)\neg D_{12}(x) \vee D_2(x)$	
	$6)\neg D_{12}(x) \vee \neg F(x)$	*Resolution(bg2, 4)*
	$7)\neg D_{12}(x) \vee \neg EF(x)$	*Resolution(2, 5)*

after the elimination of definers and symbols in $\mathcal{F}$, and the completion of filtering in Step (3) of Figure 7.2 the reduced forgetting solution is:

$$\mathcal{V}^* = \{\exists\nabla.(F \sqcap \neg E)(a)\}$$

which when negated gives the hypothesis:

$$\mathcal{H} = F \sqsubseteq E$$

In Example 8.4.2 the hypothesis computed satisfies the requirements in Definition 7.3.1 despite the fact that it is a TBox hypothesis for an ABox observation, i.e., it is still the semantically minimal explanation and requires no explicit generalisation step.

Similarly, when applied to the concept learning problem described in Section 8.3, the KB abduction approach produces the same hypotheses for the set E^+ namely:

$$\mathcal{H}_1 = \text{Poor(ale)} \vee \text{Lonely(ale)}$$
$$\mathcal{H}_2 = \text{Poor(kb)} \vee \text{Lonely(kb)}$$

demonstrating the fact that, unlike the concept learning problem, no generalisation step is performed. However, since the KB abduction approach can produce hypotheses for TBox axioms, it is possible to apply KB abduction to concept learning at another step:

Example 8.4.3. *Consider the generalisation obtained for the concept learning problem in Example 8.3.1:*

$$\mathcal{H}_{ind} = Lecturer \sqsubseteq Sad$$

Treating this generalisation as an observation for an abduction problem, it is possible to produce an abductive hypothesis for the above generalisation. Let $\mathcal{O}$ be as in Example 8.3.1, $\Psi = \mathcal{H}_{ind}$ and $\mathcal{S}_A = sig(\mathcal{O}) \setminus \{Sad\}$, then the KB abduction approach in Chapter 7 produces the following hypothesis:

$$\mathcal{H} = Lecturer \sqsubseteq (Tired \sqcap Poor) \sqcup Lonely$$

This indicates that there may be multiple ways to integrate the form of abduction in this work with induction, depending on the scope of the abduction problem.

Chapter 9

Conclusion

This thesis developed abductive reasoning capabilities in the setting of description logic ontologies. Abductive reasoning is an important tool in tasks such as diagnostics, ontology repair, expansion of ontologies, query explanation and automating parts of scientific investigation. The need for abductive reasoning in the setting of description logic ontologies has been recognised, but there was still a lack of practical systems for abductive reasoning in this domain, particularly for the task of computing hypotheses satisfying constraints such as semantic minimality. This is not surprising, given the inherent difficulty of abductive reasoning, especially in the presence of such constraints.

To develop the abduction approaches presented in this work, the understanding of the promising connection between forgetting and abductive reasoning was developed with respect to desirable abductive constraints in DLs. This led to the identification and definition of a new abductive reasoning problem in the DL setting: computing a space of independent explanations that make the fewest assumptions necessary to explain a given observation. This problem lifts a common constraint applied to semantically minimal hypotheses, in that disjunction is allowed in and forms a core part of the hypotheses produced.

An algorithm for solving the identified abduction problem was proposed, and several approaches were developed based on the above connection. In each case, the need for efficient approaches to eliminating redundant inferences from forgetting solutions was identified and solved. The result is the first set of methods that solve the ABox abduction problem of computing semantically minimal, disjunctive hypotheses for $\mathcal{ALC}$ ontologies. In addition, by extending these results, the first method for computing these hypotheses for the generalised problem of Knowledge Base abduction was developed.

The practicality of the developed approach to abduction was evaluated for ABox, TBox and KB abduction via experiments over corpora containing a range of DL ontologies from repositories under active industrial and research use. Up to now, there have been few extensive experimental evaluations of abductive reasoning in DLs, and no evaluations of an approach that produces hypotheses that are semantically minimal with disjunction. Therefore, a new framework for conducting meaningful experimental evaluations of abductive reasoning systems in DLs, including the generation of appropriate, non-trivial observations, was developed and utilised.

The use of semantically minimal, disjunctive hypotheses in DLs had not received any significant attention, likely due to a lack of systems that can produce such hypotheses and a need for further investigation into the effects of these hypotheses in practice. Thus, directions for utilising the developed abduction methods have been presented as part of this work, providing a basis for the use of these approaches in a variety of promising tasks.

To summarise, the main contributions of this thesis are as follows:

- The capabilities of forgetting-based abduction were developed with respect to abductive constraints identified in the DL setting. The need to eliminate redundant inferences, with respect to abduction, from forgetting solutions was identified with respect to DL ontologies.

- A new abduction problem for DLs was defined and motivated, where the hypothesis takes the form of a semantically minimal space of independent explanations.

- An algorithm for solving this abduction problem was proposed. An approach was developed using the forgetting tool LETHE and an efficient annotation-based approach to eliminating redundant explanations. The result is the first method for computing semantically minimal, disjunctive hypotheses in $\mathcal{ALC}$.

- The use of the semantic forgetting tool FAME was investigated for this setting, where the use of the resolution-based and semantic forgetting approaches were compared with respect to the forms of the hypotheses produced.
- The proposed abduction problem was extended to include TBox abduction, requiring the notion of disjunctive TBoxes. Motivated by the ABox abduction results, the forgetting calculus of LETHE was extended to address more expressive ABox abduction and TBox abduction. The result is the first approach that solves the generalisation of these two problems, Knowledge Base abduction, producing semantically minimal spaces of independent explanations for $\mathcal{ALC}$ ontologies.
- Each of the forgetting-based abduction approaches were evaluated over corpora consisting of ontologies used in industry and research. New approaches to conducting experiments for abduction in DLs, including problems such as generating non-trivial observations, was developed. The results demonstrated the practicality of forgetting-based approaches and the effect of observation size and forgetting signature size in the KB abduction setting.

The approaches to abduction developed in this thesis provide new, practical solutions for novel, expressive abductive reasoning problems for DL ontologies, that will enable a range of important applications that require ampliative reasoning outside the scope of existing deductive reasoning tools. The new perspective on abduction in DLs, provided by the core problem identified in this thesis where the aim is to produce a least assumptive space of independent explanations, also provides a promising basis for a number of future research directions.

While this thesis considers abduction over input ontologies and observations expressed the DL $\mathcal{ALC}$, the problem and methods developed within present a new perspective and a framework demonstrating salient findings and techniques which carry over to more expressive DLs.

9.1 Future Work

Abduction in More Expressive DLs. The approaches developed in this thesis have focused on forgetting-based abduction for $\mathcal{ALC}$. Resolution-based forgetting and semantic forgetting have been compared for this setting and an existing forgetting calculus has been extended to cover a broader range of $\mathcal{ALC}$ explanations. The semantic forgetting approaches explored in Chapter 6 of this thesis can also be used to compute forgetting solutions for extensions of $\mathcal{ALC}$ [ZS16, ZS17, Zha18] while extensions for DLs such as $\mathcal{SHQ}$ [KS14b] also exist for the resolution-based abduction calculus utilised in Chapter 5. Utilising these forgetting systems for more expressive forgetting-based abduction is a natural direction for future work. Though these approaches are suitable for solving the forgetting problem in more expressive DL languages, further research needs to be done on extending both the abduction problem identified in this thesis and the forgetting-based approach used to solve it to more expressive DL languages.

Abduction in Lightweight DLs. This thesis has focused on forgetting-based abduction in the expressive DL $\mathcal{ALC}$, where the hypotheses may be expressed in extensions of $\mathcal{ALC}$ when necessary. Many existing ontologies use fragments of the lightweight DL $\mathcal{EL}$, particularly in domains that emphasise efficient reasoning over large knowledge bases such as SNOMED CT [SPSW01]. Therefore, developing efficient methods for abductive reasoning over lightweight ontologies is an important problem. Directions for utilising the approaches developed in this thesis for lightweight DLs include investigating alternative forgetting systems such as NUI [KWW09], which computes $\mathcal{EL}$ solutions for ontologies expressed in $\mathcal{EL}$, or developing a new lightweight method based on the insights gained in this work but without the additional machinery required for $\mathcal{ALC}$.

Additional Preference Relations. The focus in this work has been on computing hypotheses satisfying the criteria of semantic minimality and independence of explanations. Consequently, other criteria such as computing explanations of minimal length, i.e., syntactic minimality, have not been addressed. As discussed briefly in Chapter 4, it is possible to prioritise eliminating certain equivalent explanations over others by applying an ordering to the process of checking each explanation during the filtering step of the algorithm. However, the use of other preference relations or abductive criteria may be worth investigating. For example, computing all explanations up to a given length may be desirable in some applications. Solutions may be as simple as removing all returned explanations exceeding a given length or, more exhaustively, refining hypotheses as proposed in Chapter 8 to return all explanations up to a given length ordered by entailment.

Enhancing Efficiency of Filtering. The filtering approaches developed in this work are essential for eliminating redundant explanations from the hypotheses produced. As demonstrated by the experimental results, both the annotation-based and set-of-support based approaches show excellent performance in practice. Improving further upon these filtering methods will be important in scaling the abduction approaches in this work to even larger knowledge bases. Promising directions include integrating the two filtering approaches and perhaps devising an approach to eliminating additional redundant explanations without the use of an external reasoner by extending the scope of the approximate filtering approaches.

Forgetting-based Abduction and Learning. The connection between abductive reasoning and induction [FK00a] has been identified as a promising direction with respect to problems such as the automation of scientific hypothesis generation [KWJ$^+$04, Ray05, Ray07, Ray09]. This connection has been investigated previously, particularly in the areas of abductive and inductive logic programming (ILP) [MB00, RBR03, Ray09, IFKN09] and has been indicated as a promising direction for future research in these areas [MDRP$^+$12]. The use of abductive reasoning in subsymbolic and statistical learning has also been recently proposed as a promising combination

[DXYZ19, Zho19]. In this work, the use of forgetting-based abduction has been proposed as a promising basis for enhancing the capabilities of ILP inspired approaches to concept learning in DLs [LH10]. This is based upon the form taken by the hypotheses produced. The fact that these hypotheses are both semantically minimal and a disjunction of independent explanations is promising for top-down, refinement operator based approaches: the abductive hypotheses may provide a constrained search space starting at the weakest explanations, leveraging existing background knowledge. However, this represents only an initial proposal and further research is required into this combination. A possible starting point would be to combine and evaluate the use of the algorithms presented in this thesis with algorithms for inductive learning in DLs [LH10, SSB15]. Further research into combining the disjunctive hypotheses in this thesis with statistical learning approaches may also be promising.

Hypothesis Refinement and Use of Data. As discussed in Chapter 8, utilising data to guide the process of hypothesis refinement – and thus the choice of forgetting signature – is a potentially promising direction. Further research should be done on optimising the process of iteratively refining hypotheses and performing experiments to determine how this process performs over larger ontologies.

Probabilistic Abduction. Abduction is a form of non-monotonic reasoning, and thus the hypotheses produced only represent possible explanations rather than guaranteed truths. It is quite natural that further investigation may yield information that invalidates one or several of the explanations produced. Therefore, probabilistic abductive reasoning is a natural direction. This is particularly true for the hypotheses produced by the approaches in this thesis: the ability to assign probabilities to the individual explanations contained within the overall hypothesis would provide a way to assign preference to a given way of explaining the observations. This may also be useful as a natural way to direct the process of iterative hypothesis refinement suggested in Chapter 8 based on collected data.

Bibliography

[ABHM03] Carlos Areces, Patrick Blackburn, Bernadette Martinez Hernandez, and Maarten Marx. Handling boolean ABoxes. In *Proceedings of the International Workshop on Description Logics, DL-03*, volume 81. CEUR Workshop Proceedings, 2003.

[Ack35] Wilhelm Ackermann. Untersuchungen über das Eliminationsproblem der mathematischen Logik. *Mathematische Annalen*, 110:330–413, 1935.

[AHB13] Jamal Atif, Céline Hudelot, and Isabelle Bloch. Explanatory reasoning for image understanding using formal concept analysis and description logics. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 44(5):552–570, 2013.

[BCM+03] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider. The description logic handbook. Cambridge University Press, 2003.

[Ben13] Mike Bennett. The financial industry business ontology: Best practice for big data. *Journal of Banking Regulation*, 14(3-4):255–268, 2013.

[BHD+11] James Blythe, Jerry R Hobbs, Pedro Domingos, Rohit Kate, and Raymond Mooney. Implementing weighted abduction in markov logic. In *Proceedings of the Ninth International Conference on Computational*

Semantics (IWCS 2011). Association for Computational Linguistics, 2011.

[Bie08] Meghyn Bienvenu. Complexity of abduction in the $\mathcal{EL}$ family of lightweight description logics. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Eleventh International Conference, KR 2008*, pages 220–230. AAAI Press, 2008.

[BLW16] Lorenz Bühmann, Jens Lehmann, and Patrick Westphal. Dl-learner—a framework for inductive learning on the semantic web. *Journal of Web Semantics*, 39:15–24, 2016.

[BMH08] Michael Bada, Chris Mungall, and Lawrence Hunter. A call for an abductive reasoning feature in OWL-reasoning tools toward ontology quality control. In *OWLED*, volume 432. CEUR Workshop Proceedings, 2008.

[CDGL99] Diego Calvanese, Giuseppe De Giacomo, and Maurizio Lenzerini. Reasoning in expressive description logics with fixpoints based on automata on finite trees. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence, IJCAI-99*, pages 84–89. AAAI Press, 1999.

[CDNDS+03] Simona Colucci, Tommaso Di Noia, Eugenio Di Sciascio, Francesco M Donini, and Marina Mongiello. Concept abduction and contraction in description logics. In *Proceedings of the International Workshop on Description Logics, DL 2003*, volume 81. CEUR Workshop Proceedings, 2003.

[CDNDS+05] Simona Colucci, Tommaso Di Noia, Eugenio Di Sciascio, Francesco M Donini, and Marina Mongiello. Concept abduction and contraction for semantic-based discovery of matches and negotiation spaces in an e-marketplace. *Electronic Commerce Research and Applications*, 4(4):345–361, 2005.

[CMP96] Marta Cialdea Mayer and Fiora Pirri. Abduction is not deduction in reverse. *Logic Journal of the IGPL*, 4:95–108, 1996.

[Con04] Gene Ontology Consortium. The gene ontology (go) database and informatics resource. *Nucleic acids research*, 32(suppl_1):D258–D261, 2004.

[COSS13] Diego Calvanese, Magdalena Ortiz, Mantas Simkus, and Giorgio Stefanoni. Reasoning about explanations for negative query answers in DL-Lite. *Journal of Artificial Intelligence Research*, 48:635–669, 2013.

[CRL10] Domenico Corapi, Alessandra Russo, and Emil Lupu. Inductive logic programming as abductive search. In *Technical Communications of the 26th International Conference on Logic Programming*, volume 7 of *LIPIcs*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2010.

[Del19] Warren Del–Pinto. Forgetting–based abductive reasoning and inductive learning in ontologies. In Carsten Fuhs, Philipp Rümmer, Renate Schmidt, and Cesare Tinelli, editors, *Deduction Beyond Satisfiability*, volume 9, pages 23–44. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2019.

[DK02] Marc Denecker and Antonis Kakas. Abduction in logic programming. In *Computational logic: Logic programming and beyond*, volume 2407 of *Lecture Notes in Computer Science*, pages 402–436. Springer, 2002.

[DKMS04] Patrick Doherty, Steve Kertes, Martin Magnusson, and Andrzej Szalas. Towards a logical analysis of biochemical pathways. In *European Workshop on Logics in Artificial Intelligence*, pages 667–679. Springer, 2004.

[DLS01] Patrick Doherty, Witold Łukaszewicz, and Andrzej Szałas. Computing strongest necessary and weakest sufficient conditions of first-order formulas. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence, IJCAI-01*, pages 145–151. AAAI Press, 2001.

[DNDSD07] Tommaso Di Noia, Eugenio Di Sciascio, and Francesco M Donini. Semantic matchmaking as non-monotonic reasoning: A description logic approach. *Journal of Artificial Intelligence Research*, 29:269–307, 2007.

[DS17] Warren Del-Pinto and Renate A Schmidt. Forgetting-based abduction in $\mathcal{ALC}$. In *SOQE*, volume 2013, pages 27–35. CEUR Workshop Proceedings, 2017.

[DS18] Warren Del-Pinto and Renate Schmidt. Iterative abduction using forgetting. In *25th Automated Reasoning Workshop*, page 36, 2018.

[DS19a] Warren Del-Pinto and Renate A. Schmidt. ABox abduction via forgetting in $\mathcal{ALC}$. In *Proceedings of the Thirty-Third Conference on Artificial Intelligence, AAAI-19*, pages 2768–2775. AAAI Press, 2019.

[DS19b] Warren Del-Pinto and Renate A. Schmidt. Extending forgetting-based abduction using nominals. In *Frontiers of Combining Systems Twelfth International Symposium, FroCoS 2019*, volume 11715 of *Lecture Notes in Computer Science*, pages 185–202. Springer, 2019.

[DSTW04] James Delgrande, Torsten Schaub, Hans Tompits, and Kewen Wang. A classification and survey of preference handling approaches in non-monotonic reasoning. *Computational Intelligence*, 20:308–334, 2004.

[DWM17] Jianfeng Du, Hai Wan, and Huaguan Ma. Practical TBox abduction based on justification patterns. In *Proceedings of the Thirty-First AAAI*

Conference on Artificial Intelligence, AAAI 2017, pages 1100–1106. AAAI Press, 2017.

[DWS14] Jianfeng Du, Kewen Wang, and Yi-Dong Shen. A tractable approach to ABox abduction over description logic ontologies. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2014*, pages 1034–1040. AAAI Press, 2014.

[DWS15] Jianfeng Du, Kewen Wang, and Yi-Dong Shen. Towards tractable and practical ABox abduction over inconsistent description logic ontologies. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, pages 1489–1495. AAAI Press, 2015.

[DXYZ19] Wang-Zhou Dai, Qiuling Xu, Yang Yu, and Zhi-Hua Zhou. Bridging machine learning and logical reasoning by abductive learning. In *Advances in Neural Information Processing Systems*, pages 2815–2826. Neural Information Processing Systems Foundation, Inc, 2019.

[EG95] Thomas Eiter and Georg Gottlob. The complexity of logic-based abduction. *Journal of the ACM (JACM)*, 42(1):3–42, 1995.

[EKS06] Corinna Elsenbroich, Oliver Kutz, and Uli Sattler. A case for abductive reasoning over ontologies. In *Proceedings of OWL: Experiences and Directions*, volume 216. CEUR Workshop Proceedings, 2006.

[EPS18] Mnacho Echenim, Nicolas Peltier, and Yanis Sellami. A generic framework for implicate generation modulo theories. In *International Joint Conference on Automated Reasoning*, pages 279–294. Springer, 2018.

[EPT17] Mnacho Echenim, Nicolas Peltier, and Sophie Tourret. Prime implicate generation in equational logic. *Journal of Artificial Intelligence Research*, 60:827–880, 2017.

[EW08] Thomas Eiter and Kewen Wang. Semantic forgetting in answer set programming. *Artificial Intelligence*, 172(14):1644–1672, 2008.

[FK00a] Peter A. Flach and Antonis C. Kakas, editors. *Abduction and Induction: Essays on their relation and integration*, volume 18 of *Applied Logic Series*. Kluwer Academic Press, 2000.

[FK00b] Peter A. Flach and Antonis C. Kakas. Abductive and inductive reasoning: background and issues. In Peter A. Flach and Antonis C. Kakas, editors, *Abduction and Induction: Essays on their relation and integration*, pages 1–27. Kluwer Academic Press, 2000.

[GHM+08] Bernardo Cuenca Grau, Ian Horrocks, Boris Motik, Bijan Parsia, Peter Patel-Schneider, and Ulrike Sattler. OWL 2: The next step for OWL. *Journal of Web Semantics*, 6(4):309–322, 2008.

[GHM+14] Birte Glimm, Ian Horrocks, Boris Motik, Giorgos Stoilos, and Zhe Wang. Hermit: an OWL 2 reasoner. *Journal of Automated Reasoning*, 53(3):245–269, 2014.

[GPH05] Yuanbo Guo, Zhengxiang Pan, and Jeff Heflin. LUBM: A benchmark for OWL knowledge base systems. *Journal of Web Semantics*, 3:158–182, 2005.

[Gru95] Thomas R Gruber. Toward principles for the design of ontologies used for knowledge sharing. *International journal of human-computer studies*, 43(5-6):907–928, 1995.

[GSS08] Dov M. Gabbay, Renate A. Schmidt, and Andrzej Szałas. Second-order quantifier elimination: Foundations, computational aspects and applications. *College Publications*, 12, 2008.

[HB12] Ken Halland and Katarina Britz. ABox abduction in $\mathcal{ALC}$ using a DL tableau. In *Proceedings of the Annual Research Conference of the*

South African Institute for Computer Scientists and Information Technologists, SAICSIT 2012, pages 51–58. ACM, 2012.

[HBK14] Ken Halland, Katarina Britz, and Szymon Klarman. TBox abduction in $\mathcal{ALC}$ using a DL tableau. In *Proceedings of the International Workshop on Description Logics, DL 2014*, volume 1193, pages 556–566. CEUR Workshop Proceedings, 2014.

[Hen63] Leon Henkin. An extension of the craig-lyndon interpolation theorem. *Journal of Symbolic Logic*, 28:201–216, 1963.

[HKP$^+$09] Pascal Hitzler, Markus Krötzsch, Bijan Parsia, et al. OWL 2 web ontology language primer. 2009.

[HKS06] Ian Horrocks, Oliver Kutz, and Ulrike Sattler. The even more irresistible $\mathcal{SROIQ}$. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Tenth International Conference, KR 2006*, pages 57–67. AAAI Press, 2006.

[Hor05] Ian Horrocks. OWL: A description logic based ontology language. In *International conference on principles and practice of constraint programming*, pages 5–8. Springer, 2005.

[HS99] Ian Horrocks and Ulrike Sattler. A description logic with transitive and inverse roles and role hierarchies. *Journal of logic and computation*, 9(3):385–410, 1999.

[HSAM93] Jerry R. Hobbs, Mark E. Stickel, Douglas E. Appelt, and Paul Martin. Interpretation as abduction. *Artificial intelligence*, 63(1-2):69–142, 1993.

[HSG04] Ullrich Hustadt, Renate A. Schmidt, and Lilia Georgieva. A survey of decidable first-order fragments and description logics. *Journal on Relational Methods in Computer Science*, 1:251–276, 2004.

[IFKN09] Katsumi Inoue, Koichi Furukawa, Ikuo Kobayashi, and Hidetomo Nabeshima. Discovering rules by meta-level abduction. In *International Conference on Inductive Logic Programming*, pages 49–64. Springer, 2009.

[IS19] Katsumi Inoue and Chiaki Sakama. Disjunctive abduction. *New Generation Computing*, 37(2):219–243, 2019.

[KC17] Patrick Koopmann and Jieying Chen. Computing $\mathcal{ALCH}$-Subsumption modules using uniform interpolation. In *Proceedings of the Workshop on Second-Order Quantifier Elimination and Related Topics, SOQE-17*, volume 2013, pages 51–66. CEUR Workshop Proceedings, 2017.

[KDTS20] Patrick Koopmann, Warren Del-Pinto, Sophie Tourret, and Renate A. Schmidt. Signature-based abduction for expressive description logics. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Seventeenth International Conference, KR 2020*. AAAI Press, 2020.

[KES11] Szymon Klarman, Ulle Endriss, and Stefan Schlobach. ABox abduction in the description logic $\mathcal{ALC}$. *Journal of Automated Reasoning*, 46:43–80, 2011.

[KKT92] Antonis C. Kakas, Robert A. Kowalski, and Francesca Toni. Abductive logic programming. *Journal of Logic and Computation*, 2 (6):719–770, 1992.

[Kon92] Kurt Konolige. Abduction versus closure in causal theories. *Artificial Intelligence*, 53:255–272, 1992.

[Koo15] Patrick Koopmann. *Practical Uniform Interpolation for Expressive Description Logics*. PhD thesis, Department of Computer Science, The University of Manchester, UK, 2015.

[KPHS07] Aditya Kalyanpur, Bijan Parsia, Matthew Horridge, and Evren Sirin. Finding all justifications of OWL DL entailments. In *The Semantic Web*, pages 267–280. Springer, 2007.

[KS13] Patrick Koopmann and Renate A. Schmidt. Uniform interpolation of $\mathcal{ALC}$ ontologies using fixpoints. In *Frontiers of Combining Systems Ninth International Symposium, FroCoS 2013*, volume 8152 of *Lecture Notes in Computer Science*, pages 87–102. Springer, 2013.

[KS14a] Patrick Koopmann and Renate A Schmidt. Computing uniform interpolants of $\mathcal{ALCH}$-ontologies with background knowledge. In *Joint Automated Reasoning Workshop and Deduktionstreffen*. IJCAR, 2014.

[KS14b] Patrick Koopmann and Renate A Schmidt. Count and forget: Uniform interpolation of $\mathcal{SHQ}$-ontologies. In *International Joint Conference on Automated Reasoning*, pages 434–448. Springer, 2014.

[KS15a] Patrick Koopmann and Renate A. Schmidt. LETHE: Saturation based reasoning for non-standard reasoning tasks. In *Informal Proceedings of the Fourth International Workshop on OWL Reasoner Evaluation, ORE-15*, volume 1387, pages 23–30. CEUR Workshop Proceedings, 2015.

[KS15b] Patrick Koopmann and Renate A. Schmidt. Uniform interpolation and forgetting for $\mathcal{ALC}$ ontologies with ABoxes. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, AAAI-15*, pages 175–181. AAAI Press, 2015.

[KS17] Yevgeny Kazakov and Peter Skočovský. Enumerating justifications using resolution. In *Proceedings of the International Workshop on Description Logics, DL 2017*, volume 1879. CEUR Workshop Proceedings, 2017.

[KSH12] Markus Krötzsch, Frantisek Simancik, and Ian Horrocks. A description logic primer. 2012.

[KWJ+04] Ross D. King, Kenneth E. Whelan, Ffion M. Jones, Philip G. K. Reiser, Christopher H. Bryant, Stephen H. Muggleton, Douglas B. Kell, and Stephen G. Oliver. Functional genomic hypothesis generation and experimentation by a robot scientist. *Nature*, 427:247–252, 2004.

[KWW08] Boris Konev, Dirk Walther, and Frank Wolter. The logical difference problem for description logic terminologies. *Automated Reasoning*, 5195:259–274, 2008.

[KWW09] Boris Konev, Dirk Walther, and Frank Wolter. Forgetting and uniform interpolation in large-scale description logic terminologies. In *Proceedings of the Twenty-First International Joint Conference on Artificial Intelligence, IJCAI-09*, pages 830–835. AAAI Press, 2009.

[Lac00] Nicolas Lachiche. Abduction and induction from a non-monotonic reasoning perspective. In *Abduction and Induction*, pages 107–116. Springer, 2000.

[LDI12] Patrick Lambrix, Zlatan Dragisic, and Valentina Ivanova. Get my pizza right: Repairing missing is-a relations in $\mathcal{ALC}$ ontologies. In *Proceedings of the Seventh Joint International Semantic Technology Conference, JIST-17*, volume 7774 of *Lecture Notes in Computer Science*, pages 17–32. Springer, 2012.

[LH10] Jens Lehmann and Pascal Hitzler. Concept learning in description logics using refinement operators. *Machine Learning*, 78(1-2):203, 2010.

[Lin01] Fangzhen Lin. On strongest necessary and weakest sufficient conditions. *Artificial Intelligence*, 128:143–159, 2001.

[LK14] Michel Ludwig and Boris Konev. Practical uniform interpolation and forgetting for $\mathcal{ALC}$ TBoxes with applications to logical difference. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Fourteenth International Conference, KR-14*, pages 318–327. AAAI Press, 2014.

[LLMW06] Hongkai Liu, Carsten Lutz, Maja Milicic, and Frank Wolter. Updating description logic ABoxes. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Tenth International Conference KR-06*, pages 46–56. AAAI Press, 2006.

[LR94] Fangzhen Lin and Ray Reiter. Forget it! In *Working Notes of the AAAI Fall Symposium on Relevance*, pages 154–159. AAAI Press, 1994.

[LW11] Carsten Lutz and Frank Wolter. Foundations for uniform interpolation and forgetting in expressive description logics. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence, IJCAI-11*, pages 989–995. AAAI Press, 2011.

[LWKDI13] Patrick Lambrix, Fang Wei-Kleiner, Zlatan Dragisic, and Valentina Ivanova. Repairing missing is-a structure in ontologies is an abductive reasoning problem. In *Second International Workshop on Debugging Ontologies and Ontology Mappings (WoDOOM 2013), Montpellier, France, May 27, 2013*, pages 33–44. Rheinisch-Westfaelische Technische Hochschule Aachen University, 2013.

[MB00] Stephen Muggleton and Christopher H. Bryant. Theory completion using inverse entailment. In *Proceedings of the Tenth International Conference on Inductive Logic Programming, ILP-00*, volume 1866 of *Lecture Notes in Computer Science*, pages 130–146. Springer, 2000.

[MDR94] Stephen Muggleton and Luc De Raedt. Inductive logic programming:

Theory and methods. *The Journal of Logic Programming*, 19:629–679, 1994.

[MDRP+12] Stephen Muggleton, Luc De Raedt, David Poole, Ivan Bratko, Peter Flach, Katsumi Inoue, and Ashwin Srinivasan. Ilp turns 20. *Machine learning*, 86(1):3–23, 2012.

[Moo00] Raymond J. Mooney. Integrating abduction and induction in machine learning. In *Abduction and Induction*, pages 181–191. Kluwer, 2000.

[MP17] Nicolas Matentzoglu and Bijan Parsia. Bioportal snapshot 30.03.2017 [data set], 2017. Zenodo, http://doi.org/10.5281/zenodo.439510.

[MPH18] Drahomír Mrózek, Júlia Pukancová, and Martin Homola. Abox abduction solver exploiting multiple dl reasoners. In *International Workshop on Description Logics, DL-18*, volume 2211. CEUR Workshop Proceedings, 2018.

[NS95] Andreas Nonnengart and Andrzej Szalas. A fixpoint approach to second-order quantifier elimination with applications to correspondence theory. 1995.

[Pei78] Charles Sanders Peirce. Deduction, induction and hypothesis. *Popular Science Monthly*, 13:470–482, 1878.

[Pei60] Charles Sanders Peirce. *Collected papers of charles sanders peirce*, volume 2. Harvard University Press, 1960.

[PH17] Júlia Pukancová and Martin Homola. Tableau-based ABox abduction for the $\mathcal{ALCHO}$ description logic. In *Proceedings of the International Workshop on Description Logics, DL-17*, volume 1879. CEUR Workshop Proceedings, 2017.

[PH18] Júlia Pukancová and Martin Homola. Abox abduction for description logics: The case of multiple observations. In *International Workshop* *on Description Logics, DL-18*, volume 2211. CEUR Workshop Proceedings, 2018.

[PKMM08] Irma Sofia Espinosa Peraldi, Atila Kaya, Sylvia Melzer, and Ralf Möller. On ontology based abduction for text interpretation. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 194–205. Springer, 2008.

[Pla94] David A. Plaisted. The search efficiency of theorem proving strategies. In *International Conference on Automated Deduction*, pages 57–71. Springer, 1994.

[PMIMS17] Rafael Penaloza, C. Mencía, Alexey Ignatiev, and Joao Marques-Silva. Lean kernels in description logics. In *Proceedings of the Semantic Web Fourteenth International Conference, ESWC-17*, volume 10249 of *Lecture Notes in Computer Science*, pages 518–533. Springer, 2017.

[Poo89] David Poole. Explanation and prediction: an architecture for default and abductive reasoning. *Computational Intelligence*, 5(2):97–110, 1989.

[Pop73] Harry E. Pople. On the mechanization of abductive logic. In *IJCAI*, volume 73, pages 147–152. Citeseer, 1973.

[Pop14] Karl Popper. *Conjectures and refutations: The growth of scientific knowledge*. routledge, 1962/2014.

[PPU03] Ram'on Pino-Peréz and Carlos Uzcátegui. Preferences and explanations. *Artificial Intelligence*, 149:1–30, 2003.

[Ray05] Oliver Ray. *Hybrid Abductive Inductive Learning*. PhD thesis, Department of Computing, Imperial College London, UK, 2005.

[Ray07] Oliver Ray. Automated abduction in scientific discovery. In *Model-Based Reasoning in Science, Technology, and Medicine*, pages 103–116. Springer, 2007.

[Ray09] Oliver Ray. Nonmonotonic abductive inductive learning. *Journal of Applied Logic*, 7:329–340, 2009.

[RB08] Alan L Rector and Sebastian Brandt. Why do it the hard way? the case for an expressive description logic for snomed. *Journal of the American Medical Informatics Association*, 15(6):744–751, 2008.

[RBR03] Oliver Ray, Krysia Broda, and Alessandra Russo. Hybrid abductive inductive learning: A generalisation of progol. In *International Conference on Inductive Logic Programming*, pages 311–328. Springer, 2003.

[Rei87] Raymond Reiter. A theory of diagnosis from first principles. *Artificial intelligence*, 32(1):57–95, 1987.

[RM10] Sindhu Raghavan and Raymond J. Mooney. Bayesian abductive logic programs. In *AAAI-10 Workshop on Statistical Relational AI*, pages 82–87. AAAI Press, 2010.

[RNM05] Rajat Raina, Andrew Y. Ng, and Christopher D. Manning. Robust textual inference via learning and abductive reasoning. In *"Proceedings of the Twentieth Conference on Artificial Intelligence, AAAI-05",*, pages 1099–1105. AAAI Press, 2005.

[RSS20] Tjitze Rienstra, Claudia Schon, and Steffen Staab. Concept contraction in the description logic el. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Seventeenth International Conference, KR 2020*, pages 723–732. AAAI Press, 2020.

[RWK09] Oliver Ray, Ken Whelan, and Ross King. A nonmonotonic logical approach for modelling and revising metabolic networks. In *3rd International Conference on Complex, Intelligent and Software Intensive Systems*, pages 825–829. IEEE, 2009.

[Sch94] Klaus Schild. Terminological cycles and the propositional μ-calculus. In *Principles of Knowledge Representation and Reasoning*, pages 509–520. Elsevier, 1994.

[SMvS$^+$18] Joerg Schoenfisch, Christian Meilicke, Janno von Stülpnagel, Jens Ortmann, and Heiner Stuckenschmidt. Root cause analysis in it infrastructures using ontologies and abduction in markov logic networks. *Information Systems*, 74:103–116, 2018.

[SPG$^+$07] Evren Sirin, Bijan Parsia, Bernardo Cuenca Grau, Aditya Kalyanpur, and Yarden Katz. Pellet: A practical OWL-DL reasoner. *Journal of Web Semantics*, 5(2):51–53, 2007.

[SPM$^+$12] Craig Schlenoff, Edson Prestes, Raj Madhavan, Paulo Goncalves, Howard Li, Stephen Balakirsky, Thomas Kramer, and Emilio Miguelanez. An IEEE standard ontology for robotics and automation. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1337–1342. IEEE, 2012.

[SPSW01] Michael Q Stearns, Colin Price, Kent A Spackman, and Amy Y Wang. Snomed clinical terms: overview of the development process and project status. In *Proceedings of the AMIA Symposium*, page 662. American Medical Informatics Association, 2001.

[SSB15] Viachaslau Sazonau, Uli Sattler, and Gavin Brown. General terminology induction in OWL. In *International Semantic Web Conference*, pages 533–550. Springer, 2015.

[SSS91] Manfred Schmidt-Schauß and Gert Smolka. Attributive concept descriptions with complements. *Artificial intelligence*, 48(1):1–26, 1991.

[Sti91] Mark Stickel. A Prolog-like inference system for computing minimum-cost abductive explanations in natural-language interpretation. *Annals of Mathematics and Artificial Intelligence*, 4:89–106, 1991.

[T+55] Alfred Tarski et al. A lattice-theoretical fixpoint theorem and its applications. *Pacific journal of Mathematics*, 5(2):285–309, 1955.

[TNCKM06] Alireza Tamaddoni-Nezhad, Raphael Chaleil, Antonis Kakas, and Stephen Muggleton. Application of abductive ilp to learning metabolic network inhibition from temporal data. *Machine Learning*, 64(1-3):209–230, 2006.

[Wer13] Christoph Wernhard. Abduction in logic programming as second-order quantifier elimination. In *Frontiers of Combining Systems Ninth International Symposium, FroCoS 2013*, volume 8152 of *Lecture Notes in Computer Science*, pages 103–119. Springer, 2013.

[WKDL14] Fang Wei-Kleiner, Zlatan Dragisic, and Patrick Lambrix. Abduction framework for repairing incomplete $\mathcal{EL}$ ontologies: Complexity results and algorithms. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2014*, pages 1120–1127. AAAI Press, 2014.

[ZAS+19] Yizheng Zhao, Ghadah Alghamdi, Renate A Schmidt, Hao Feng, Giorgos Stoilos, Damir Juric, and Mohammad Khodadadi. Tracking logical difference in large-scale ontologies: a forgetting-based approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3116–3124. AAAI Press, 2019.

[Zha18] Yizheng Zhao. *Automated semantic forgetting for expressive description logics*. PhD thesis, Department of Computer Science, The University of Manchester, UK, 2018.

[Zho19] Zhi-Hua Zhou. Abductive learning: Towards bridging machine learning and logical reasoning. *Science China Information Sciences*, 62(7):76101, 2019.

[ZS15] Yizheng Zhao and Renate A. Schmidt. Concept forgetting in $\mathcal{ALCOI}$-ontologies using an Ackermann approach. In *Proceedings of the Fourteenth International Semantic Web Conference, ISWC-15*, volume 9366 of *Lecture Notes in Computer Science*, pages 587–602. Springer, 2015.

[ZS16] Yizheng Zhao and Renate A. Schmidt. Forgetting concept and role symbols in $\mathcal{ALCOIH}\mu^+(\nabla,\sqcap)$-ontologies. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI-16*, pages 1345–1352. AAAI Press, 2016.

[ZS17] Yizheng Zhao and Renate A. Schmidt. Role forgetting for ALCOQH(universal role)-ontologies using an ackermann-based approach. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 1354–1361. AAAI Press, 2017.

[ZZ10] Yan Zhang and Yi Zhou. Forgetting revisited. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Twelfth International Conference, KR 2010*, pages 602–604. AAAI Press, 2010.

Appendix A

Appendix

A.1 KB Abduction: Properties of the Forgetting Approach

This section covers the proofs necessary to show that the abduction approach can be used to compute solutions as required by Definition 7.3.1, originally presented in the extended version of [KDTS20][1].

A.1.1 Forgetting Approach Computes All Relevant Inferences

As in Chapter 5, the forgetting step is used to compute all *relevant* inferences of the background ontology together with the negated observation. Here, the relevant inferences are those that are relevant to the abduction problem, i.e., have an ancestor in $\neg\Psi$. For the KB abduction approach, in Step (2) of Figure 7.2 the forgetting procedure is applied using the set-of-support inspired approach. Therefore, it is necessary to show that the forgetting approach in Step (2) retains all of the relevant inferences

First, some notions are defined that will be utilised in the proofs in this section. Let Φ be a set of clauses and $\mathcal{S} \in \mathcal{F}$ be a concept or role symbol to be eliminated. Let N_D^Φ be the set of definers introduced during the normalisation of Φ. Here, D always

[1]https://arxiv.org/abs/2007.00757

refers to a definer. Each subset $\mathbf{D}$ of N_D^Φ is mapped to a definer D_D, where each D_D is effectively the conjunction of each definer in the given subset e.g. $D_{D_1} = D_1$ and $D_{\{D_1 \cup D_2\}} = D_1 \sqcap D_2$. As before, the notation D_{12} will be used to represent $D_{\{D_1 \cup D_2\}}$ for simplicity. Let $\mathbf{I}$ be the set of all individuals occuring in Φ together with a set of individuals containing a single unique individual name a_D corresponding to each definer symbol in $\mathrm{Sat}_{\mathcal{S}}(\Phi)$. It is assumed also that Φ contains at least one individual name.

For every definer symbol D_i in $sig(\mathrm{Sat}_{\mathcal{S}}(\Phi))$, a new individual a_{D_i} is introduced. Let $\mathbf{I}$ be the set of all individuals occurring in Φ together with the individuals corresponding to each definer.

The set of clauses $\mathrm{Sat}_S(\Phi)$ denotes the set of clauses obtained by exhaustively applying the calculus in Figure 7.3 to eliminate $\mathcal{S}$ from Φ. The *grounding* of $\mathrm{Sat}_{\mathcal{S}}(\Phi)$, denoted $\mathrm{Sat}^g_{\mathcal{S}}(\Phi)$, is defined as follows:

$$\mathrm{Sat}^g_{\mathcal{S}}(\Phi) = \{\phi[x \to a] \mid a \in \mathbf{I}, \phi \in \mathrm{Sat}_{\mathcal{S}}(\Phi)\}$$

Let $\prec_D$ be a total ordering over introduced definer symbols such that for two subsets $\mathbf{D}_i, \mathbf{D}_j \subseteq N_D^\Phi$, for the corresponding definers D_i and D_j the relation $D_i \prec D_j$ holds if $\mathbf{D}_i \subseteq \mathbf{D}_j$. Now to define an ordering specifying the order in which inferences are made using the calculus, based on the symbols being considered.

Let $\prec_S$ be a total ordering over literals. For two literals L_i, L_j, it is the case that $L_i \prec_S L_j$ if at least one of the following holds, where D, D_1 and D_2 are definers:

1. L_i takes the form $D(t)$ or $\neg D(t)$ and L_j does not
2. L_i takes the form $D_1(t)$ or $\neg D_1(t)$ and L_j takes the form $D_2(t')$ or $\neg D_2(t')$ where $D_1 \prec_D D_2$.
3. $\mathcal{S} \notin sig(L_i)$ and $\mathcal{S} \in sig(L_j)$.
4. L_i takes one of the forms $A(t)$, $r(a,b)$ or $(\exists r.D)(t)$ and L_j takes one of the forms $\neg A'(t')$, $\neg r'(a',b')$ or $\forall r'.D'(t')$.

5. L_i takes the form $\exists r.D_1(t)$ and L_j takes the form $(\exists r.D_2)(t)$ where $D_1 \prec D_2$.

The ordering $\prec_S$ is extended to clauses ϕ using the multiset extension, where $\phi_i \prec_S \phi_j$ if there is a literal $L_j \in \phi_j$ such that for all literals $L_i \in \phi_i$, it is the case that $L_i \prec_S L_j$.

Now to the required proofs [KDTS20]. It is necessary to prove that the set-of-support inspired strategy described in Section 7.4.2 computes all required inferences for the abduction problem. As discussed, the following condition is assumed when applying the calculus to eliminate a given symbol $\mathcal{S}$ [KDTS20]:

> **(*)** For a given clause $\phi = L_1(x) \vee ... \vee L_n(x)$, inferences are performed only on literals L_i such that $\mathcal{S} \in sig(L_i)$ or they are applications of the *Resolution* rule on definer symbols in clauses of the form $\neg D_1(x) \vee D_2(x)$.

The following Lemma uses the refutational completeness of the calculus, using condition (*), to show that all required consequences are computed when eliminating a symbol $\mathcal{S}$.

Lemma A.1.1. *Given a set of clauses Φ obtained by normalising a KB $\mathcal{K}$ and some concept or role symbol $\mathcal{S}$, Φ is satisfiable if and only if $Sat_{\mathcal{S}}(\Phi)$ does not contain the empty clause.*

Proof: *Assume that $Sat_{\mathcal{S}}(\Phi)$ does not contain the empty clause. Let $\mathcal{I}$ be a model of Φ based on the grounding $Sat^g_{\mathcal{S}}(\Phi)$. For every individual $a \in \boldsymbol{I}$, the model $\mathcal{I}$ has exactly one domain element resulting in the domain $\Delta^{\mathcal{I}} = \{d_a \mid a \in \boldsymbol{I}\}$. This implies that if $\mathcal{I} \models Sat^g_{\mathcal{S}}(\Phi)$ then $\mathcal{I} \models Sat_{\mathcal{S}}(\Phi)$, since $Sat^g_{\mathcal{S}}(\Phi)$ is a grounding of $Sat_{\mathcal{S}}(\Phi)$, and also $\mathcal{I} \models \Phi$, since $Sat_{\mathcal{S}}(\Phi)$ is a set of consequences derived from Φ. The corresponding interpretation function $\cdot^{\mathcal{I}}$ is constructed by induction as follows.*

For the base case $i = 0$, the interpretation $\mathcal{I}_0 = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}}_0 \rangle$ is given by setting:

(1) *For each $a \in \boldsymbol{I}$, $a^{\mathcal{I}_0} = d_a$.*

(2) *For each $D \in N_D$, $D^{\mathcal{I}_0} = \emptyset$ if $\neg D(x) \in Sat_{\mathcal{S}}(\Phi)$*

(3) *For each concept and role symbol U, excluding definer symbols, $U^{\mathcal{I}_0} = \emptyset$*

Now to proceed stepwise for $i > 0$. While $\mathcal{I}_{i-1} \not\models Sat^g_{\mathcal{S}}(\Phi)$, the aim is to extend $\mathcal{I}_{i-1}$ by considering the smallest clause $\phi_m \in Sat^g_{\mathcal{S}}(\Phi)$ that is not yet entailed by $\mathcal{I}_{i-1}$. Let $\mathcal{I}_i = \langle \Delta^{\mathcal{I}}, \cdot_i^{\mathcal{I}} \rangle$ be the next interpretation that extends $\mathcal{I}_{i-1}$, which is constructed from $\mathcal{I}_i$ in the following way, where L is the maximal literal in ϕ_m according to the ordering $\prec_S$:

(a) *If $L = A(a)$, then set $A^{\mathcal{I}_i} = A^{\mathcal{I}_i} \cup \{d_a\}$.*

(b) *If $L = r(a,b)$, then set $r^{\mathcal{I}_i} = r^{\mathcal{I}_{i-1}} \cup \{(d_a, d_b)\}$.*

(c) *If $L = (\exists r.D)(a)$, then set $r^{\mathcal{I}_i} = r^{\mathcal{I}_{i-1}} \cup \{(d_a, d_{a_D})\}$.*

(d) *Else, set $\mathcal{I}_i = \mathcal{I}_{i-1}$.*

For (c), it is the case that $d_{a_D} \in D^{\mathcal{I}_0}$ unless $\neg D(x) \in Sat_{\mathcal{S}}(\Phi)$. In this case, application of the $R\exists$ rule in Figure 7.3 on r in ϕ_m, followed by the $R\nabla$ rule, results in a clause ϕ'_m which excludes $\exists r.D$ from ϕ_m. However, since $\phi'_m \prec_S \phi$ and given that it was assumed $\mathcal{I}_{i-1} \not\models \phi_m$, then it must be the case that $\mathcal{I}_{i-1} \not\models \phi'_m$, which is a contradiction.

The rest of the proof proceeds similarly to the proof of Theorem 2 in Koopmann and Schmidt [KS15b], where for all $\phi \in Sat(\Phi)$ such that $\phi \prec_S \phi_m$, $\mathcal{I}_i \models \phi$ and case (d) cannot apply. Since the difference here is that the normal form includes negated role assertions of the form $\neg r(a,b)$, it remains to cover this case. Assume that case (d) applies from $\mathcal{I}_{i-1}$ to $\mathcal{I}_i$ and that in this case the smallest clause ϕ_m not entailed by $\mathcal{I}_{i-1}$ contains a maximal literal L of the form $\neg r(a,b)$. Note that, since $\mathcal{I}_i = \mathcal{I}_{i-1}$ in case (d), this also means $\mathcal{I}_i \models \phi_m$. This implies $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r_i^{\mathcal{I}}$, i.e., a and b are related under r. Here, both a and b must be individuals that occurred in Φ, meaning that $b^{\mathcal{I}}$ is not an individual a_D corresponding to a definer D. This implies that case (b) must have applied in some previous step to an interpretation $\mathcal{I}_j$ where $j < i$, i.e., there must have been a clause $\phi = \phi' \vee r(a,b)$ where $r(a,b)$ is the maximal literal. As a result, it must be the case that $\phi \prec_S \phi'$. However, inference via rule Rr in Figure 7.3 should result in a clause $\phi' \vee \phi'_m$, since the clause appearing in a previous step, ϕ, contains $r(a,b)$ as a maximal literal and it is assumed that the smallest non-entailed clause ϕ_m contains

$\neg r(a,b)$. Therefore, $\phi' \vee \phi'_m \in Sat_{\mathcal{S}}(\Phi)$. Since $\mathcal{I}_i \not\models \phi'_m$ and $\phi' \vee \phi'_m \prec_S \phi_m$ there is a contradiction on the minimality of the clause ϕ_m.

Given that $Sat^g_{\mathcal{S}}(\Phi)$ is finite, a new clause in $Sat^g_{\mathcal{S}}(\Phi)$ is entailed at each step from (a) – (c) and all smaller clauses in $Sat^g_{\mathcal{S}}(\Phi)$ remain entailed, there must exist a step $i > 0$ at which the updated interpretation $\mathcal{I}_i \models Sat^g_{\mathcal{S}}(\Phi)$. At this point, setting $\mathcal{I} = \mathcal{I}_i$ and noting that $\mathcal{I} \models Sat_{\mathcal{S}}(\Phi)$ and $\mathcal{I} \models \Phi$, it is possible to conclude that Φ is satisfiable using $\mathcal{I}$ as a model.

The following two lemmata concern properties of introduced definers [KDTS20]. For the following, assume that the restriction **(*)** is not applied and that the rules in Figure 7.3 are applied exhaustively, resulting in the set of clauses Sat(Φ).

Lemma A.1.2. *Let Φ be a normalised set of clauses with definers N_D^{Φ}. For every $\boldsymbol{D}_i \subseteq N_D^{\Phi}$ and every $\boldsymbol{D}_j \subset \boldsymbol{D}_i$, for which definers D_i and D_j are introduced in Sat(Φ), a clause $\neg D_i(x) \vee D_j(x)$ is in Sat(Φ) and there exists two clauses $\phi_1 \vee Q_1.D_j$ and $\phi_1 \vee \phi_2 \vee Q_2.D_i$ in Sat(Φ) such that $Q_1 = \exists$ implies $Q_2 = \exists$.*

Lemma A.1.3. *Let Φ_1 and Φ_2 be sets of normalised clauses with $\Phi_1 \subseteq \Phi_2$ and let D, D_1 and D_2 be definers such that $D_2 \in sig(Sat(\Phi_1))$ and $\neg D(x) \vee D_1(x)$, $\neg D(x) \vee D_2(x) \in Sat(\Phi_2)$. Then there exists a definer D' such that the clauses $\neg D'(x) \vee D_1(x)$ and $\neg D'(x) \vee D_2(x)$ are in the set $Sat(\Phi_1)$ and either: $D' = D$ or there is a clause $\neg D(x) \vee D'(x)$ in the set $Sat(\Phi_2)$*

Proof: *Let the definers D_1 and D_2 be the definers corresponding to the subsets $\boldsymbol{D}_1, \boldsymbol{D}_2 \in N_D^{\Phi_1}$ respectively. From Lemma A.1.2, there exists a definer D_{12} such that $\neg D(x) \vee D_{12}(x) \in sig(\Phi_2)$. Now to show that D_{12} is introduced in $Sat(\Phi_1)$ by contradiction. Assume that in a sequence of inferences, D_{12} is the first definer introduced in $Sat(\Phi_2)$ that is not introduced in $Sat(\Phi_1)$. Let D_1 and D_2 occur in the clauses $\phi_1 \vee Q_1 r.D_1(t_1)$ and $\phi_2 \vee Q_2 r.D_2(t_2)$ in $Sat(\Phi_1)$ respectively. If the rule Role Propagation in Figure 7.3 was applied on these clauses, then this would imply that the definer D_{12} was introduced in $Sat(\Phi_1)$. Since it is assumed that it is not, then Role Propagation is not applied here. For this to be the case, it must be that ϕ_1 and ϕ_2 each contain* *negative definer literals $\neg D'_1(x)$ and $\neg D'_2(x)$ where $D'_1(x) \neq D'_2(x)$, since this would lead to a clause with more than two negative definers, which is not permitted. Since the set Sat(Φ_2) does contain D_{12}, it must be the case that clauses $\phi'_1 \vee Q_1 r.D_1(t_1)$ and $\phi'_2 \vee Q_2 r.D_2(t_2)$ occur in Sat(Φ_2), where these clauses have been inferred from $\phi_1 \vee Q_1 r.D_1(t_1)$ and $\phi_2 \vee Q_2 r.D_2(t_2)$ respectively via a sequence of inferences. It must also be the case that ϕ'_1 and ϕ'_2 do not contain negative definers, so that inference under the Role Propagation rule is possible. As a result, ϕ'_1 and ϕ'_2 must contain a definer literal D' such that the clauses $\neg D'(x) \vee D'_1(x)$ and $\neg D'(x) \vee D'_2(x)$ are in the set Sat(Φ_2), where D' is not introduced in Sat(Φ_1) nor is there a definer D'' such that $\neg D'(x) \vee D''(x)$ is in Sat(Φ_2). It must also be the case that D' was introduced prior to D_{12}, which contradicts the assumption made that D_{12} was the first definer introduced in Sat(Φ_2) that was not introduced in Sat(Φ_1).*

Now to the key proof in this section: that the saturated set of clauses obtained by applying the calculus using the set-of-support inspired strategy described in Section 7.4.2 preserves all relevant consequences of the input $\neg\mathcal{K}, \neg\Psi$. This saturated set is denoted $\text{Sat}(\Phi_B, \Phi_S, \mathcal{S}_A)$, where Φ_B is the background set of clauses, Φ_S is the supported set and $\mathcal{S}_A$ is the set of symbols to be retained, i.e., the abducibles in the abduction perspective.

Recall also the steps in applying the calculus in Figure 7.3 via the set-of-support inspired approach as follows. For each symbol in the forgetting signature $\mathcal{F}$, the steps in the final strategy are as follows:

1. Perform all inferences on symbols in the forgetting signature $\mathcal{F}$, as well as all possible inferences via the rules *RA* and *R*$\forall$-2 on definer symbols. As with $Int_{\mathcal{ALC}}$, all inferences using the role propagation and role instantiation rules that enable further inferences on symbols in $\mathcal{F}$ are also performed. All inferred clauses are added to the support set Φ_S.

2. Remove clauses containing symbols in $\mathcal{F}$ from Φ_S.

3. If a clause containing a definer is derived and thus added to Φ_S, move all clauses

containing this definer from Φ_B to Φ_S.

where all derived clauses are stored in a separate set to ensure that they are not derived repeatedly.

Theorem A.1.1. *Let Φ_B and Φ_S be normalised sets of clauses and $\mathcal{S}_A$ be a signature. Let $\boldsymbol{M} = Sat(\Phi_B, \Phi_S, \mathcal{S}_A)$. For any Boolean KB $\mathcal{K}$ expressed in $\mathcal{ALC}$ such that $sig(\mathcal{K}) \subseteq \mathcal{S}_A$, $\Phi_B \cup \Phi_S \models \mathcal{K}$ if and only if $\Phi_B \cup \boldsymbol{M} \models \mathcal{K}$.*

Proof: *The set-of-support inspired algorithm for computing $Sat(\Phi_B, \Phi_S, \mathcal{S}_A)$ loops over each non-definer symbol $\mathcal{S} \in \Phi_S$ such that $\mathcal{S} \notin \mathcal{S}_A$. For each such $\mathcal{S}$, it is necessary to show that the first two steps (1) and (2) in the set-of-support strategy above preserve all entailments that do not utilise the symbol $\mathcal{S}$. This can be shown by induction, starting with the base case: let Φ_S^0 be the initial set of supported clauses before applying the calculus to $\mathcal{S}$, Φ_S^1 and Φ_S^2 be the results of step (1) and step (2) in the set-of-support strategy above respectively.*

For both steps, $\Phi^0 \cup \Phi_S^1 \models \mathcal{K}$ and $\Phi^0 \cup \Phi_S^2 \models \mathcal{K}$ hold if and only if $\Phi^0 \cup \Phi_S^1 \cup \neg KB \models \perp$ and $\Phi^0 \cup \Phi_S^2 \cup \neg KB \models \perp$ hold respectively. The negated KB $\neg\mathcal{K}$ can be represented as a set of clauses $\Phi_{\neg\mathcal{K}}$ as described in Section 7.4.1. Now it is necessary to show that for $M_1 = \Phi^0 \cup \Phi_S^1 \cup \Phi_{\neg\mathcal{K}}$ and $M_2 = \Phi^0 \cup \Phi_S^2 \cup \Phi_{\neg\mathcal{K}}$, the set M_2 is unsatisfiable if and only if M_1 is unsatisfiable. From Lemma A.1.1, this can be shown by showing that $Sat_S(M_2)$ contains the empty clause ($\perp$) if and only if $Sat_S(M_1)$ does also.

The forward direction is straightforward: $Sat_S(M_2)$ contains only clauses that are present also in $Sat_S(M_1)$, and so contains $\perp$ only if $Sat_S(M_1)$ does also. The reverse direction can be shown by proving that clauses that are in $Sat_S(M_1)$, but not in $Sat_S(M_2)$ can be recovered. This amounts to showing that the following holds:

- *(**) For every clause $\phi \in Sat_S(M_1)$ inferred by applying the calculus on the symbol S, there exists a clause $\phi' \in Sat_S(\Phi^0) \cup \Phi_S^1$ such that: (a) $\phi = \phi'$ or (b) $\phi = \neg D_1(x) \vee \phi_r, \phi' = \neg D_2(x) \vee \phi_r$ and $\neg D_1(x) \vee D_2(x) \in Sat_S(M_2)$.*

which can be shown by induction on the possible inferences. Let ϕ be a clause in

$Sat_S(M_1)$ that was the result of an inference on the symbol S, i.e., a clause with parent ϕ_1 that contains S and possibly a second parent ϕ_2 that also contains S. Under condition (*)*, one of the following are possible:*

(I) ϕ_1 *occurs in* $\Phi_0 \cup \Phi_S^0$

(II) ϕ_1 *is the result of an inference on the symbol S*

(III) ϕ_1 *is the result of an inference on* $\neg D_1(x) \vee D_2(x) \in Sat_S(M_1)$.

If (III) *holds, it must be the case that* $\phi_1 = \neg D_1(x) \vee \phi_1'$, $\neg D_1(x) \vee D_3(x) \in Sat_S(M_1)$ *and* $\neg D_3(x) \vee \phi_1' \in Sat_S(M_1)$*, where the clause* $\neg D_3(x) \vee \phi_1'$ *falls under cases* (I) *and* (II)*. If case* (I) *applies to both premises, then the claim holds as a result. If* (I) *or* (II) *apply and* $\phi_1, \phi_2 \in Sat(\Phi^0) \cup \Phi_S^1$*, then* $\phi \in Sat(\Phi^0) \cup \Phi_S^1$ *by the construction of* Φ_S^1*. If* $\phi \notin Sat(\Phi^0) \cup \Phi_S^1$*, then this must be due to the fact that* ϕ *contains a definer symbol that is not present in* $Sat(\Phi^0) \cup \Phi_S^1$*, i.e., it must be the case that* $\phi = \neg D(x) \vee \phi_r$*. Now assume that* (II) *applies and that the claim above holds for* ϕ_1 *and* ϕ_2*. Then the set* $Sat_S(M_2)$ *contains the following clauses:* $\phi_1 = \neg D(x) \vee \phi_{r1}$, $\phi_2 = \neg D(x) \vee \phi_{r2}$, $\neg D(x) \vee D_1(x)$ *and* $\neg D(x) \vee D_2(x)$*. Meanwhile, the set* $Sat_S(\Phi^0) \cup \Phi_S^1$ *contains the following clauses:* $\neg D_1(x) \vee \phi_{r1}$ *and* $\neg D_2(x) \vee \phi_{r2}$*. From Lemma A.1.3, it must be the case that there is a definer* D' *such that the set* $Sat_S(\Phi^0) \cup \Phi_S^1$ *contains the clauses* $\neg D'(x) \vee \phi_{r1}$ *and* $\neg D'(x) \vee \phi_{r2}$ *and the set* $Sat_S(M_2)$ *contains* $\neg D(x) \vee D'(x)$ *or it is the case that* $D' = D$*. Any inference performed on the clauses* ϕ_1 *and* ϕ_2 *also applies to the aforementioned clauses, and as a result the set* $Sat_S(\Phi^0) \cup \Phi_S^1$ *contains the clause* $\neg D'(x) \vee \phi_r$*. As a result, the statement (**) above holds.*

From statement (**), all relevant consequences resulting from inferences on S are derived during the computation of $Sat_S(\Phi^0) \cup \Phi_S^1$*, and the set* Φ_S^2 *is obtained by removing all clauses that contain the symbol S. From condition (*), inferences on either the symbol S or positive definers are performed before any others when computing* $Sat_S(M_1)$*. As a result, for every clause* $\phi \in Sat_S(M_1)$ *there is a clause* ϕ' *in* $Sat_S(M_2)$ *such that either* $\phi = \phi'$ *or the set* $Sat_S(M_2)$ *contains the clauses* $\phi = \neg D_1(x) \vee \phi_r$,

$\phi' = \neg D_2(x) \vee \phi_r$ and $\neg D_1(x) \vee D_2(x)$. From this, if $Sat_S(M_1)$ contains $\perp$ then $Sat_S(M_2)$ does also.

A.1.2 Denormalisation Preserves All Relevant Inferences

The denormalisation phase aims to capture the meaning of all the definers present in the result of the set-of-support inspired forgetting approach, i.e., the set of clauses $\mathrm{Sat}(\Phi_B, \Phi_S, \mathcal{S}_A)$. This requires the introduction of concept inclusions and corresponding definers $\overline{D}_i$ referring to each negative occurrence of the existing definers D_i. The set of CIs, and the replacement of negative occurrences of definers, is constructed as described in Section 7.4 of Chapter 7 by the following steps:

(1) $\phi \vee \forall r.D_i(x)$ *introduce* $\overline{D}_i \sqsubseteq \forall r^-.C^\phi$

(2) $\phi \vee \forall r.D_i(x)$ *introduce* $\overline{D}_i \sqsubseteq \forall r^-.(\neg\{a\} \vee C^\phi)$

(3) $\phi \vee \forall r.D_i(a)$ *introduce* $\overline{D}_i \sqsubseteq \forall r^-.(\neg\{a\} \sqcup C^\phi)$

(4) $\neg D_i(a) \vee \phi$ *introduce* $D_i \sqsubseteq \neg\{a\} \sqcup C^\phi$

(5) Replace all occurrences of negative definers $\neg D(a)$ by $\overline{D}(a)$

(6) Replace every remaining clause ϕ that does not take the form of a disjunction of ABox assertions by $\top \sqsubseteq C^\phi$.

where each concept $C^\phi = L_1^c \sqcup ... \sqcup L_n^c$ corresponds to a clause $\phi = L_1 \vee ... \vee L_n$ and each literal L^c is defined as follows for each corresponding literal L:

- If $L = C(x)$ then $L^c = C^-$, where C^- is the result of replacing all negative definer occurrences $\neg D$ by $\overline{D}$.
- If $L = C(a)$ then $L^c = \exists\nabla.(\{a\} \sqcap C^-)$.
- If $L = r(a,b)$ then $L^c = \exists\nabla.(\{a\} \sqcap \exists r.\{b\})$.
- If $L = \neg r(a,b)$ then $L^c = \exists\nabla.(\{a\} \sqcap \forall r.\neg\{b\})$

Now it remains to show that replacing the definers by these concept inclusions retains all relevant consequences, i.e., that eliminating definer symbols eliminates only consequences relating using definers. Starting with Lemma A.1.4.

Lemma A.1.4. *Let $\mathcal{K}_0$ be the result of introducing the concept inclusions in (1) and (2) above to $Sat(\Phi_B, \Phi_S, \mathcal{S}_A)$. Every model of $\mathcal{K}_0$ can be transformed into a model of: $\mathcal{K}_0 \cup \{\overline{D} \equiv \neg D | D \in sig(Sat(\Phi_B, \Phi_S, \mathcal{S}_A)\}$ by changing only the interpretation of definer symbols.*

Proof: *Let $\mathcal{I}$ be a model of $\mathcal{K}_0$ such that for a definer $D \in sig(\Phi_B)$, $\mathcal{I} \not\models \overline{D} \equiv \neg D$. Both the normalisation procedure and the calculus in Figure 7.3 ensure that definers occur only under existential or universal role restrictions. With respect to (1) and (2) above, for the existential case no CI is introduced for $\overline{D}$ and a model $\mathcal{I}'$ can be obtained such that $\mathcal{I}' \models \overline{D} \equiv \neg D$ by just setting $\overline{D}^{\mathcal{I}'} = (\neg D)^{\mathcal{I}}$. For the universal case, $\mathcal{I}$ can be transformed into a model $\mathcal{I}'$ of $\overline{D} \equiv \neg D$ by setting $D^{\mathcal{I}'} = D^{\mathcal{I}} \setminus \overline{D}^{\mathcal{I}}$ and $\overline{D}^{\mathcal{I}'} = \Delta^{\mathcal{I}} \setminus D^{\mathcal{I}'}$ where $\Delta^{\mathcal{I}}$ is the domain. From this, $\mathcal{I}' \models \overline{D} \equiv \neg D$. It can also be shown that $\mathcal{I}' \models \mathcal{K}_0$. To do this, occurrences of the definers D and $\overline{D}$ must be considered.*

For D, it is the case that $D^{\mathcal{I}'} \subseteq D^{\mathcal{I}}$, and so only positive occurrences of D in $\mathcal{K}_0$ need to be considered. The only occurrences of D are in clauses of the form $\phi \vee \forall r.D(t)$, since D does not occur under existential role restrictions and all clauses containing literals $D(t)$ are eliminated during the computation of $Sat(\Phi_B, \Phi_S, \mathcal{S}_A)$. As before, no clause mixes variables and individuals as terms.

Therefore, starting with a clause $L_1(x) \vee ... \vee L_n(x) \vee \forall r.D(x) \in \mathcal{K}_0$. Assume there exists $(d,e) \in r^{\mathcal{I}}$ such that $e \in (D^{\mathcal{I}} \cap \overline{D}^{\mathcal{I}})$, where d and e are individuals. Since $e \in \overline{D}^{\mathcal{I}}$ and $\mathcal{I} \models \overline{D} \sqsubseteq \forall r^-.(L_1^c \sqcup ... \sqcup L_n^c)$, it follows that $d \in (L_1^c \sqcup ... \sqcup L_1^c)^{\mathcal{I}}$. From this, for some literal L_i with $1 \leq i \leq n$ it is the case that $d \in L_i$. As such, there is no need for e to be in $D^{\mathcal{I}}$ to satisfy the clause, and thus $\mathcal{I}' \models L_1(x) \vee ... \vee L_n(x) \vee \forall r.D(x)$.

Next, a clause without variables $\phi \vee \forall r.D(a) \in \mathcal{K}_0$ such that $\mathcal{I} \not\models \phi$ and $(a^{\mathcal{I}}, d) \in r^{\mathcal{I}}$. In this case, the CI $\overline{D} \sqsubseteq \forall r^-.(\neg\{a\} \sqcup C^{\phi})$ is introduced in $\mathcal{K}_0$. Since $\mathcal{I} \not\models \phi$, it is the case that $(C^{\phi})^{\mathcal{I}} = \emptyset$. As a result, $\mathcal{I} \models \overline{D} \sqsubseteq \forall r^-.\neg\{a\}$ and $a^{\mathcal{I}}$ cannot have a successor under r satisfying $\overline{D}$. Thus, $\mathcal{I} \models \phi \vee \forall r.D(a)$ implies $\mathcal{I}' \models \phi \vee \forall r.D(a)$.

Now $\overline{D}$ must be considered. Since $\overline{D}^{\mathcal{I}} \subseteq \overline{D}^{\mathcal{I}'}$, the negative occurrences of $\overline{D}$ in $\mathcal{K}_0$ must also be considered. These all occur in CIs of the form $\overline{D} \sqsubseteq \forall r^-.C$ for domain elements $d \in \overline{D}^{\mathcal{I}'} \setminus \overline{D}^{\mathcal{I}}$. For all of these domain elements d, it is the case that $d \notin D^{\mathcal{I}}$, since $d \notin \overline{D}^{\mathcal{I}}$ and $d \in D^{\mathcal{I}}$ would imply $d \in D^{\mathcal{I}'}$ and $d \notin \overline{D}^{\mathcal{I}'}$. Now to show that for every e such that $(e,d) \in r^{\mathcal{I}}$, it is the case that $d \in C$. Consider each of the possibilities for the CI $\overline{D} \sqsubseteq \forall r^-.C \in \mathcal{K}_0$, with the aim of showing that in each case $e \in C^{\mathcal{I}'}$.

First, where C corresponds to the concept generated for a clause $L_1(x) \vee ... \vee L_n(x) \vee \forall r.D(x) \in \mathcal{K}_0$, i.e., $\overline{D} \sqsubseteq \forall r^-.(L_1^c \sqcup ... \sqcup L_n^c) \in \mathcal{K}_0$. Since $d \notin D^{\mathcal{I}}$, it follows that $e \notin (\forall r.D)^{\mathcal{I}}$, and therefore $e \in (L_1 \sqcup ... \sqcup L_n)^{\mathcal{I}'}$.

Second, where C corresponds to the concept generated for a clause $\phi \vee \forall r.D(a)$, i.e., $\overline{D} \sqsubseteq \forall r^-.(\neg\{a\} \sqcup C^{\phi}) \in \mathcal{K}_0$. If $e \neq a^{\mathcal{I}}$, then $e \in (\neg\{a\} \sqcup C^{\phi})^{\mathcal{I}'}$. Else, if $e = a^{\mathcal{I}}$ then $\mathcal{I} \not\models \forall r.D(a)$ since $d \notin D^{\mathcal{I}}$. As a result, $\mathcal{I} \models \phi$, implying that $e(C^{\phi})^{\mathcal{I}} = \Delta^{\mathcal{I}}$ and $e \in (\neg\{a\} \sqcup C^{\phi})^{\mathcal{I}'}$.

From the above, it follows that $\mathcal{I}' \models \overline{D} \sqsubseteq \forall r^-.C$ for every $\overline{D} \sqsubseteq \forall r^-.C \in \mathcal{K}_0$. This process can be repeated for every definer $D \in sig(Sat(\Phi_B, \Phi_S, \mathcal{S}_A))$, resulting in a model $\mathcal{I}^*$ of $\mathcal{K}_0 \cup \{\overline{D} \equiv \neg D | D \in sig(Sat(\Phi_B, \Phi_S, \mathcal{S}_A))\}$.

Now let $\mathcal{K}_1$ be the result of performing all introductions of CIs in (1)–(6) above.

Lemma A.1.5. *Every model of $\mathcal{K}_1$ can be transformed into a model of $Sat(\Phi_B, \Phi_S, \mathcal{S}_A)$ by changing only the interpretation of definers. Every model of $Sat(\Phi_B, \Phi_S, \mathcal{S}_A)$ can be extended to a model of $\mathcal{K}_0$ by setting $\overline{D}^{\mathcal{I}} = \neg D^{\mathcal{I}}$ for all definers D in $sig(Sat(\Phi_B, \Phi_S, \mathcal{S}_A))$.*

Proof: *By examining the axioms introduced in (3) – (6) above, it follows that $\mathcal{K}_1 \models \mathcal{K}_0$ and $\mathcal{K}_0 \cup \{\overline{D} \equiv \neg D | D \in sig(Sat(\Phi_B, \Phi_S, \mathcal{S}_A))\} \models \mathcal{K}_1$. From Lemma A.1.4, it is possible to transform any model $\mathcal{I}$ of $\mathcal{K}_1$ into a model $\mathcal{I}'$ of $\mathcal{K}_0$ by changing the interpretation of the definer symbols in $\mathcal{I}$. As such, $\mathcal{I}'$ is a model of $\mathcal{K}_1$ also. Every clause $\phi \in Sat(\Phi_B, \Phi_S, \mathcal{S}_A)$ has a corresponding axiom $\beta \in \mathcal{K}_0$ such that $\mathcal{I}' \models \phi$ if and only if $\mathcal{I}' \models \beta$. As such, $\mathcal{I}'$ is a model of $Sat(\Phi_B, \Phi_S, \mathcal{S}_A)$.*

Now let $\mathcal{I}$ be a model of $Sat(\Phi_B, \Phi_S, \mathcal{S}_A)$ and let $\mathcal{I}'$ coincide with $\mathcal{I}$ except for the interpretation of definer symbols $(\overline{D})^{\mathcal{I}'} = \neg D^{\mathcal{I}}$ for every definer symbol D in $sig(Sat(\Phi_B, \Phi_S, \mathcal{S}_A)$.

For every axiom $\beta \in \mathcal{K}_0$, there is a clause $\phi \in Sat(\Phi_B, \Phi_S, \mathcal{S}_A)$ such that $\mathcal{I}' \models \phi$ if and only if $\mathcal{I}' \models \beta$. From this, $\mathcal{I}'$ is a model of $\mathcal{K}_1$.

From Lemma A.1.5, all consequences of $\mathrm{Sat}(\Phi_B, \Phi_S, \mathcal{S}_A)$ that do not use definers are retained and every negative occurrence of a definer symbol has a corresponding CI. As such, it is possible to use the same definer elimination technique as in Figure 3.3 [KS15b].

For the resulting definer-free KBs, the following theorem follows directly from Lemma A.1.1, Lemma A.1.2, Ackermann's Lemma [Ack35] and the Generalised Ackermann's Lemma [NS95].

Theorem A.1.2. *Let $\mathcal{K}$ be a definer-free KB. Then, the following holds: $\mathcal{K}_2 \models \mathcal{K}$ if and only if $Sat(\Phi_B, \Phi_S, \mathcal{S}_A) \models \mathcal{K}$.*

As a consequence of Theorems A.1.1 and A.1.2, Theorem A.1.3 holds for the negation of the result of denormalising $\mathrm{Sat}(\Phi_B, \Phi_S, \mathcal{S}_A)$, i.e., the approximately reduced forgetting solution $\mathcal{V}^S_{app}$ described in Chapter 7.

Theorem A.1.3. *For a given abduction problem $\langle \mathcal{K}, \Psi, \mathcal{S}_A \rangle$, applying the set-of-support inspired forgetting approach (including denormalisation) results in a disjunction of $\mathcal{ALCOI}\mu$ KBs that satisfies conditions (i), (ii) and (iv) of Definition 7.3.1 but does not satisfy the inter-disjunct redundancy requirement in condition (iii).*

Performing the following check in Step (3) of Figure 7.2 over the above result:

$$\mathcal{K}, \mathcal{K}_i \models \mathcal{K}_1 \vee ... \vee \mathcal{K}_{i-1} \vee \mathcal{K}_{i+1} \vee ... \vee \mathcal{K}_n$$

as described in Chapter 7 then yields a hypothesis fully satisfying Definition 7.3.1, up to potential inter-disjunct redundancy in explanations (disjuncts) containing fixpoint operators if they are present.