# ABox Abduction via Forgetting in $\mathcal{ALC}$

**Warren Del-Pinto, Renate A. Schmidt**
School of Computer Science, The University of Manchester
Oxford Road, Manchester M13 9PL, United Kingdom

### Abstract

Abductive reasoning generates explanatory hypotheses for new observations using prior knowledge. This paper investigates the use of forgetting, also known as uniform interpolation, to perform ABox abduction in description logic ($\mathcal{ALC}$) ontologies. Non-abducibles are specified by a forgetting signature which can contain concept, but not role, symbols. The resulting hypotheses are semantically minimal and consist of a disjunction of ABox axioms. These disjuncts are each independent explanations, and are not redundant with respect to the background ontology or the other disjuncts, representing a form of hypothesis space. The observations and hypotheses handled by the method can contain both atomic or complex $\mathcal{ALC}$ concepts, excluding role assertions, and are not restricted to Horn clauses. Two approaches to redundancy elimination are explored in practice: full and approximate. Using a prototype implementation, experiments were performed over a corpus of real world ontologies to investigate the practicality of both approaches across several settings.

## Introduction

The aim of abductive reasoning is to generate explanatory hypotheses for new observations, enabling the discovery of new knowledge. Abduction was identified as a form of reasoning by (Peirce 1878). It has also become a recurring topic of interest in the field of AI, leading to work such as abductive extensions of Prolog for natural language interpretation (Stickel 1991; Hobbs et al. 1993), the integration of abduction with induction in machine learning (Mooney 2000) including work in the fields of inductive (Muggleton and Bryant 2000) and abductive logic programming (Kakas, Kowalski, and Toni 1992; Ray 2009) and statistical relational AI (Raghavan and Mooney 2010).

This paper focuses on abduction in description logic (DL) ontologies. These consist of a TBox of information about general entities known as *concepts* and *roles* and an ABox of assertions over instances of these concepts known as *individuals*. DL ontologies are widely used to express background knowledge and as alternative data models for knowledge management. They are commonly used in fields such as AI, computational linguistics, bio-informatics and robotics. The need for abductive reasoning in ontologies was highlighted by (Elsenbroich, Kutz, and Sattler 2006). Use cases include hypothesis generation, diagnostics and belief expansion for which most current reasoning methods for ontologies are not suitable. This has led to work on abduction in DLs, including studies in $\mathcal{EL}$ (Bienvenu 2008) and applications such as ontology repair (Lambrix, Dragisic, and Ivanova 2012; Wei-Kleiner, Dragisic, and Lambrix 2014) and query explanation (Calvanese et al. 2013). For ABox abduction, methods in more expressive logics such as $\mathcal{ALC}$ and its extensions have been proposed (Klarman, Endriss, and Schlobach 2011; Halland and Britz 2012; Pukancová and Homola 2017). Similarly, work exists on TBox abduction (Wei-Kleiner, Dragisic, and Lambrix 2014; Halland, Britz, and Klarman 2014). However, few implementations and evaluations are available for abductive reasoning. Exceptions include the ABox abduction method of (Du, Wang, and Shen 2014) in Datalog rewritable ontologies and a TBox abduction method using justification patterns (Du, Wan, and Ma 2017).

The aim of this paper is to investigate the use of *forgetting* for ABox abduction in DL ontologies. Forgetting is a non-standard reasoning method that restricts ontologies to a specified set of symbols, retaining all entailments preservable in the restricted signature. It is also referred to as *uniform interpolation* or *second-order quantifier elimination*, and has been proposed as a method for abduction in different contexts (Doherty, Łukaszewicz, and Szałas 2001; Gabbay, Schmidt, and Szałas 2008; Wernhard 2013; Koopmann and Schmidt 2015b). However, so far the forgetting-based approach has been insufficiently studied or applied, particularly in terms of preferred characteristics of abductive hypotheses and in the setting of large DL ontologies.

This work investigates hypotheses obtained using forgetting-based abductive reasoning. These hypotheses are weakest sufficient conditions (Lin 2001), related to the DL literature notion of semantic minimality (Halland, Britz, and Klarman 2014), meaning that they make the fewest assumptions necessary to explain an observation given the background knowledge. However, without additional steps, these hypotheses are not guaranteed to be consistent and are likely to be mostly redundant when the forgetting based approach is applied to large ontologies. In this work, additional constraints are investigated to capture these redundancies and practical methods for their removal are presented.

The main contributions of this paper are: (1) Forgetting-based ABox abduction in DL ontologies is explored and formalised. The aim is to compute hypotheses that do not contain unnecessary assumptions nor misleading, i.e. redundant, explanations. The need to eliminate redundancies from uniform interpolants is motivated and solved. (2) A practical method for this task is presented for $\mathcal{ALC}$. It computes hypotheses containing only abducible symbols. Non-abducibles are specified by a forgetting signature consisting of any set of concept, but not role, symbols. Both the observations and hypotheses may contain any atomic or complex $\mathcal{ALC}$ (or $\mathcal{ALC}\mu$) concepts, but cannot contain role assertions. An efficient annotation-based filtering method is proposed to eliminate redundancies from uniform interpolants. The method uses the forgetting tool LETHE which is shown to be applicable to ABox abduction, thereby answering an open question in (Koopmann and Schmidt 2015b). However, the general framework could use any forgetting method designed for $\mathcal{ALC}$. (3) The method is evaluated empirically over a corpus of real-world ontologies. An approximate and a full approach to redundancy elimination are compared.

Proofs and additional examples are in the long version of this paper https://arxiv.org/abs/1811.05420.

## Problem Definition

Concepts in the description logic $\mathcal{ALC}$ have the following forms: $\top \mid A \mid \neg C \mid C \sqcup D \mid C \sqcap D \mid \forall r.C \mid \exists r.D$, where $A$ denotes a concept name, $C$ and $D$ are arbitrary $\mathcal{ALC}$ concepts and $r$ is a role name. *Atomic* concepts are concept names, while concepts such as $\forall r.(A \sqcap B)$ are said to be *complex*. A knowledge base or ontology $\mathcal{O}$ in $\mathcal{ALC}$ consists of a TBox and an ABox. The TBox consists of a set of general concept inclusions of the form $C \sqsubseteq D$, where $C$ and $D$ are any $\mathcal{ALC}$ concept. The ABox contains axioms $C(a)$ and role assertions of the form $r(a,b)$, where $C$ is any $\mathcal{ALC}$ concept and $a$ and $b$ are individuals. The signature of $X$, denoted as $sig(X)$, is the set of all concept and role names occurring in $X$, where $X$ can be any ontology or axiom.

The aim of abduction is to compute a hypothesis to explain a new observation. Our focus is on this problem:

**Definition 1. Abduction in Ontologies.** *Let $\mathcal{O}$ be an ontology and $\psi$ a set of ABox axioms, where $\psi$ does not contain role assertions, 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 *which contains all role symbols in $(\mathcal{O}, \psi)$. The* abduction problem *is to find a hypothesis $\mathcal{H}$ as a disjunction of ABox axioms, without role assertions, that contains only those symbols specified in $\mathcal{S}_A$ such that:* (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 $\alpha_i$ in $\mathcal{H}$ such that $\mathcal{O}, \alpha_i \models \alpha_1 \sqcup ... \sqcup \alpha_{i-1} \sqcup \alpha_{i+1} \sqcup ... \sqcup \alpha_n$ and* (iv) *for any $\mathcal{H}'$ satisfying conditions (i)–(iii) where $sig(\mathcal{H}') \subseteq \mathcal{S}_A$, if $\mathcal{O}, \mathcal{H} \models \mathcal{O}, \mathcal{H}'$ then $\mathcal{O}, \mathcal{H}' \models \mathcal{O}, \mathcal{H}$.*

The set of abducibles $\mathcal{S}_A$ defines the subset of symbols in the ontology that may appear in the hypothesis $\mathcal{H}$. Here, $\mathcal{S}_A$ must contain all role symbols in $(\mathcal{O}, \psi)$ and both the observation $\psi$ and $\mathcal{H}$ may not contain role assertions. For our approach, the language of $\mathcal{ALC}$ must be extended to include disjunctive ABox assertions over multiple individuals, and in some specific cases fixpoints (Calvanese, De Giacomo, and Lenzerini 1999) to represent cyclic results. These will be discussed alongside the proposed method.

The rationale for the problem conditions is to focus efforts on computing informative hypotheses. Otherwise, the search space for hypotheses would be too large. Defining the set of abducibles $\mathcal{S}_A$ allows a user to focus on explanations containing specific information represented as symbols, utilising their own knowledge of the problem domain. Conditions (i) and (ii) 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 $\bot$ would be entailed from which everything follows. Condition (ii) ensures that $\mathcal{H}$ explains the observation $\psi$ when added to the background knowledge in $\mathcal{O}$. Conditions (iii) and (iv) capture two distinct notions. Condition (iii) ensures that each of the disjuncts in the hypothesis $\mathcal{H}$ are independent explanations (Konolige 1992) for the observation $\psi$. That is, there are no disjuncts in $\mathcal{H}$ that express information that is the same or more specific than that which is already expressed by the other disjuncts in $\mathcal{H}$. This also excludes disjuncts that are simply inconsistent with the background knowledge as a special case, since if for a disjunct $\alpha$ in $\mathcal{H}$ we have $\mathcal{O}, \alpha \models \bot$ then everything follows. Condition (iii) is referred to as *inter-disjunct redundancy*. The following example illustrates this, where Brain Drain (BD) is a disease, "BDV1" and "BDV2" are viruses, p1 is a patient and the acronyms *hD, hS* and *cO* stand for "hasDisease", "hasSymptom" and "carrierOf" respectively:

**Example 1.** *Let $\mathcal{O} = \{\exists hD.BD \sqsubseteq \exists hS.Headache$, $TiredScientist \sqsubseteq \exists hS.Headache$, $\exists cO.(BDV1 \sqcup BDV2) \sqsubseteq \exists hD.BD$, $TiredAccountant \sqsubseteq \exists hS.Headache$, $\neg TiredAccountant(p1)\}$, $\psi = \{\exists hS.Headache(p1)\}$ and let $\mathcal{S}_A$ include all symbols in $\mathcal{O}$ except Headache. Consider the hypothesis $\mathcal{H}' = (\exists hD.BD \sqcup TiredScientist \sqcup \exists cO.(BDV1 \sqcup BDV2) \sqcup TiredAccountant)(p1)$. This satisfies conditions (i), (ii) and (iv). However, there are two redundant disjuncts: $TiredAccountant(p1)$ and $\exists cO.(BDV1 \sqcup BDV2)(p1)$. 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 $\mathcal{H}'$. A user may mistakenly believe that these two redundancies are valid, independent explanations for the symptom. Thus, condition (iii) excludes these redundancies, resulting in the preferred hypothesis $\mathcal{H} = (\exists hD.BD \sqcup TiredScientist)(p1)$.*

As condition (iii) requires that each disjunct be consistent with the ontology $\mathcal{O}$, condition (i) is not strictly needed. However, as consistency is a key condition in most abduction contexts it is useful to emphasise it separately.

Condition (iv) captures the notion of *semantic minimality* (Halland, Britz, and Klarman 2014) under the background knowledge $\mathcal{O}$. It restricts hypotheses to those that make the fewest assumptions necessary to explain the observation $\psi$ given $\mathcal{O}$. This is shown in the example below.

**Example 2.** *Let $\mathcal{O} = \{A \sqsubseteq B, B \sqsubseteq C\}$, $\psi = \{C(a)\}$ and $\mathcal{S}_A = \{A, B\}$. Consider the hypotheses $\mathcal{H}_1 = B(a)$ and $\mathcal{H}_2 = A(a)$. Both satisfy the conditions in Definition 1(i) and (ii). However, hypothesis $\mathcal{H}_2$ does not satisfy (iv), since $\mathcal{O}, \mathcal{H}_2 \models \mathcal{O}, \mathcal{H}_1$, but the reverse does not hold. Thus, $\mathcal{H}_2$ is* *a stronger or "less minimal" hypothesis than $\mathcal{H}_1$.*

From this, it can be seen that condition (iv) rejects semantically stronger hypotheses. It should be noted that, unlike some other settings such as (Halland, Britz, and Klarman 2014), here $\mathcal{H}$ can contain disjunction. Thus, redundant disjuncts must be considered separately, as in condition (iii), since condition (iv) does not account for these.

With these conditions, the aim of this work is to compute a semantically minimal hypothesis consisting of all disjuncts that each represent an independent explanation of the observation $\psi$, none of which overlaps with either the background knowledge or the other disjuncts.

Definition 1 does not remove all choices between or redundancies in the forms taken by each disjunct in $\mathcal{H}$ if they are equivalent under $\mathcal{O}$. For example, condition (iv) does not account for conjunctively joined redundancies that follow directly from $\mathcal{O}$. If Example 2 is extended so that the axiom $C \sqsubseteq D$ is in $\mathcal{O}$ and the signature of abducibles $\mathcal{S}_A$ also contains $D$, then $\mathcal{H}_3 = (B \sqcap D)(a)$ is also a valid hypothesis under conditions (i), (ii) and (iv). While $\mathcal{H}_3$ is not stronger than $\mathcal{H}_1$, it contains a form of redundancy: $D(a)$.

To eliminate these redundancies and simplify the disjuncts themselves may require the use of preference criteria over the disjuncts in $\mathcal{H}$. As there are a variety of methods for defining and realising preference handling (Cialdea Mayer and Pirri 1996; Pino-Peréz and Uzcátegui 2003; Delgrande et al. 2004) we do not discuss this aspect. Here, the focus is on computing the space of independent explanations, rather than ensuring each takes the simplest form.

## Forgetting and Uniform Interpolation

Forgetting is a 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 $\mathcal{O}$. 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, which is a uniform interpolant:

**Definition 2.** ***Uniform Interpolation in $\mathcal{ALC}$ (Lutz and Wolter 2011).*** *Let $\mathcal{O}$ be an $\mathcal{ALC}$ ontology and $\mathcal{F}$ a set of symbols to be forgotten from $\mathcal{O}$. Let $\mathcal{S}_A = sig(\mathcal{O}) \setminus \mathcal{F}$ be the complement of $\mathcal{F}$. The* uniform interpolation problem *is the task of finding an ontology $\mathcal{V}$ such that the following conditions hold:* (i) $sig(\mathcal{V}) \subseteq \mathcal{S}_A$, (ii) *for any axiom $\beta$: $\mathcal{O} \models \beta$ iff $\mathcal{V} \models \beta$ provided that $sig(\beta) \subseteq \mathcal{S}_A$. The ontology $\mathcal{V}$ is a* uniform interpolant *of $\mathcal{O}$ for the signature $\mathcal{S}_A$. We also say that $\mathcal{V}$ is the* result of forgetting *$\mathcal{F}$ from $\mathcal{O}$.*

Uniform interpolants are strongest necessary entailments, in general, it holds that:

**Theorem 1.** *$\mathcal{V}$ is a uniform interpolant of ontology $\mathcal{O}$ for $\mathcal{S}_A$ iff $\mathcal{V}$ is a strongest necessary entailment of $\mathcal{O}$ in $\mathcal{S}_A$.*

This means that for any ontology $\mathcal{V}'$, if $sig(\mathcal{V}') \subseteq \mathcal{S}_A$ and $\mathcal{V}' \models \mathcal{V}$, then $\mathcal{V} \models \mathcal{V}'$. Of the methods for uniform interpolation in $\mathcal{ALC}$, e.g., (Ludwig and Konev 2014; Koopmann and Schmidt 2015a), our abduction method uses the resolution-based method developed by Koopmann and Schmidt [2013; 2015a; 2015b].

Here, this method is referred to as $Int_{\mathcal{ALC}}$. Motivations for utilising $Int_{\mathcal{ALC}}$ include the fact that it can perform forgetting for $\mathcal{ALC}$ with ABoxes (Koopmann and Schmidt 2015a), making it suitable for the setting in this paper. Furthermore, in theory the result of forgetting (and abduction) can involve an infinite chain of axioms. Using $Int_{\mathcal{ALC}}$, such chains can be finitely represented using fixpoint operators. In practice, these are rarely required: in previous work only 7.2% of uniform interpolants contained cycles (Koopmann and Schmidt 2013). $Int_{\mathcal{ALC}}$ can also handle disjunctive ABox assertions which are not representable in pure $\mathcal{ALC}$. These will be needed for some abduction cases involving multiple individuals. In terms of efficiency, the size of the forgetting result is constrained to at most a double exponential bound with respect to the input ontology and $Int_{\mathcal{ALC}}$ is guaranteed to terminate (Koopmann and Schmidt 2015a).

The method $Int_{\mathcal{ALC}}$ has two properties that are also essential to the proposed abduction method. (i) *Soundness*: any ontology $\mathcal{O}'$ returned by applying $Int_{\mathcal{ALC}}$ to an ontology $\mathcal{O}$ is a uniform interpolant. (ii) *Interpolation Completeness*: if there exists a uniform interpolant $\mathcal{O}'$ 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 uniform interpolant.

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

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

**∃-Role Restriction Elimination:**
$$\frac{C \vee (\exists r.D)(t) \qquad \neg D(x)}{C}$$

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

$D_1$ and $D_2$ are definer symbols, Q $\in \{\forall, \exists\}$, $\sigma$ 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 1: $Int_{\mathcal{ALC}}$ rules utilised in our abduction method.

The method $Int_{\mathcal{ALC}}$ relies on the transformation of the ontology to a normal form given by a set of clauses of concept literals. The inference rules of the forgetting calculus utilised in $Int_{\mathcal{ALC}}$ are shown in Figure 1. Definer symbols are introduced to represent concepts that fall under the scope of a quantifier. Resolution inferences are restricted to concepts in $\mathcal{F}$ or definer symbols. Once all possible inferences have been made, any clauses containing symbols in $\mathcal{F}$ are removed and the definer symbols are eliminated resulting in an ontology $\mathcal{O}'$ that is free of all symbols in $\mathcal{F}$. A discussion of this calculus and the associated method, including proofs, can be found in (Koopmann and Schmidt 2015a).

We will also 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. For example, the premises $\{A \sqsubseteq B, C \sqsubseteq A, \neg B(a)\}$ are expressed as the clauses: $\{\neg A(x) \vee B(x), \neg C(x) \vee A(x), \neg B(a)\}$. For a forgetting signature $\mathcal{F} = \{B, A\}$, resolution between $\neg A(x) \vee B(x)$ and $\neg B(a)$ gives $\neg A(a)$. Resolution between $\neg A(a)$ and $\neg C(x) \vee A(x)$ gives $\neg C(a)$. The axioms $A \sqsubseteq B$ and $\neg B(a)$ are the *parents* of the axiom $\neg A(a)$ and the *ancestors* of $\neg C(a)$.

In this paper, we focus on ABox abduction where the set of abducibles includes all role symbols. Non-abducibles are specified by the *forgetting signature* $\mathcal{F}$ which contains only concept symbols occurring in the ontology $\mathcal{O}$ or observation $\psi$. The proposed method utilises $Int_{\mathcal{ALC}}$ to compute semantically minimal hypotheses via forgetting and contrapositive reasoning, exploiting: $\mathcal{O}, \mathcal{H} \models \psi$ iff $\mathcal{O}, \neg\psi \models \neg\mathcal{H}$ where $\mathcal{O}$ is an ontology and $\psi$, $\mathcal{H}$ are (ABox) axioms.

## A Forgetting-Based Abduction Method

The abduction algorithm takes as **input** an $\mathcal{ALC}$ ontology $\mathcal{O}$, an observation $\psi$ as a set of ABox axioms and a forgetting signature $\mathcal{F}$. Several assumptions are made regarding this input. The method $Int_{\mathcal{ALC}}$ does not cater for negated role assertions as can be seen in Figure 1, and the form of role forgetting in $Int_{\mathcal{ALC}}$ is not complete for abduction. As a result, $\psi$ cannot contain role assertions and $\mathcal{F}$ is restricted to concept symbols in $sig(\mathcal{O} \cup \psi)$. Correspondingly, the signature of abducibles $\mathcal{S}_A$ must contain all role symbols occurring in $sig(\mathcal{O} \cup \psi)$. Also, if $\mathcal{F}$ does not contain at least one symbol in the observation $\psi$, the semantically minimal hypothesis will simply be $\psi$ 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 concept symbol in the signature of $\psi$. 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 (Koopmann and Schmidt 2013) to finitely represent infinite forgetting solutions. For our method, this means that the abduction result may contain least fixpoints due to the negation of greatest fixpoints under contraposition. In these cases, the output language would be $\mathcal{ALC}\mu$.

The **output** is a hypothesis $\mathcal{H} = \alpha_1(a_1) \sqcup ... \sqcup \alpha_n(a_n)$ containing only the abducible symbols $\mathcal{S}_A = sig(\mathcal{O} \cup \psi) \setminus \mathcal{F}$, that satisfies the conditions (i)–(iv) in Definition 1. Note that $\mathcal{H}$ may be a disjunctive assertion over several individuals, motivating the need to extend $\mathcal{ALC}$ with these.

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

**(1)** Compute the uniform interpolant $\mathcal{V} = \{\beta_1, ..., \beta_n\}$ of $(\mathcal{O}, \neg\psi)$ with respect to the forgetting signature $\mathcal{F}$.

**(2)** Extract the set $\mathcal{V}^* \subseteq \mathcal{V}$ by omitting axioms $\beta_i \in \mathcal{V}$ such that $\mathcal{O}, \beta_1, ..., \beta_{i-1}, \beta_{i+1}, ..., \beta_n \models \beta_i$.

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

In more detail, the observation $\psi$ 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 $\neg\psi = \neg C_1(a_1) \sqcup ... \sqcup \neg C_k(a_k)$. The forgetting method $Int_{\mathcal{ALC}}$ is used to compute the uniform interpolant $\mathcal{V}$ of $(\mathcal{O}, \neg\psi)$ by forgetting the concept names in $\mathcal{F}$, i.e., $\mathcal{V} = (\mathcal{O}, \neg\psi)^{-\mathcal{F}}$.

If forgetting was used in isolation, the negation of $\mathcal{V}$ would be the hypothesis for $\psi$ under contraposition. However, this is only guaranteed to satisfy conditions (ii) and (iv) of Definition 1: since $\mathcal{V}$ is the strongest necessary entailment of $(\mathcal{O}, \neg\psi)$ in $\mathcal{S}_A$ as in Theorem 1, its negation would be the weakest sufficient condition (Lin 2001; Doherty, Łukaszewicz, and Szałas 2001). Thus the hypothesis would be semantically minimal in $\mathcal{S}_A$, but would not necessarily satisfy condition (i), consistency, nor condition (iii), absence of inter-disjunct redundancy. In practice most of the disjuncts will be redundant, as the experimental results show (Table 2). In the case that there is no suitable hypothesis, an inconsistent or "false" hypothesis will be returned since all of the axioms in $\mathcal{V}$ would follow directly from $\mathcal{O}$.

Step (2) therefore omits information in $\mathcal{V}$ that follows from the background knowledge $\mathcal{O}$ together with other axioms in $\mathcal{V}$ itself. This check is the dual of Definition 1(iii), and therefore eliminates inter-disjunct redundancies such as those in Example 1. The result is a *reduced uniform interpolant* $\mathcal{V}^*$ which takes the form $\mathcal{V}^* = \{\beta_1(a_1), ..., \beta_k(a_k)\}$ where each $\beta_i$ is an $\mathcal{ALC}(\mu)$ concept.

If an axiom $\beta_i$ 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 $\mathcal{H}$. The order in which the axioms are checked can be random, or can be based on some preference relation (Cialdea Mayer and Pirri 1996).

In Step (3) the reduced uniform interpolant $\mathcal{V}^*$ is negated, resulting in the hypothesis $\mathcal{H}$. Thus, each disjunct $\alpha_i$ in $\mathcal{H}$ is the negation of an axiom $\beta_i$ in $\mathcal{V}^*$, i.e., $\alpha_i \equiv \neg\beta_i$.

The described method is sound and complete.

**Theorem 2.** *Let $\mathcal{O}$ be an $\mathcal{ALC}$ ontology, $\psi$ an observation as a set of ABox axioms, excluding role assertions, and $\mathcal{S}_A$ a set of abducible symbols containing all role symbols in $\mathcal{O}, \psi$ and $\mathcal{S}_A \subseteq sig(O, \psi)$.* (i) **Soundness***: The hypothesis $\mathcal{H}$ returned by the method is a disjunction of ABox axioms such that sig($\mathcal{H}$) $\subseteq \mathcal{S}_A$ and $\mathcal{H}$ satisfies Definition 1(i)-(iv).* (ii) **Completeness***: If there exists a hypothesis $\mathcal{H}'$ such that $sig(\mathcal{H}') \subseteq \mathcal{S}_A$ and $\mathcal{H}'$ satisfies Definition 1(i)–(iv), then the method returns a hypothesis $\mathcal{H}$ such that $\mathcal{O}, \mathcal{H} \equiv \mathcal{O}, \mathcal{H}'$.*

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

## Practical Realisation

For redundancy elimination, Step (2) requires checking the relation $\mathcal{O}, \mathcal{V} \setminus \beta_i \not\models \beta_i$ for every axiom $\beta_i$ in $\mathcal{V}$. Since entailment checking in $\mathcal{ALC}$ has exponential complexity and $\mathcal{V}$ is in the worst case double exponential in the size of $(\mathcal{O}, \neg\psi)$, this step has a 3EXPTIME upper bound which is very expensive particularly for large ontologies. Regardless, Step (2) is essential; without it there will be a large number of inter-disjunct redundancies (Definition 1(iii)) in the hypotheses obtained. This is reflected in the experiments (Table 2).

To obtain a computationally feasible implementation of Step (2), 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 $\neg\psi$. An axiom $\beta$ 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 $\mathcal{V}^*$.

In this paper, dependency tracing is achieved by using *annotations*, similar to (Kazakov and Skočovský 2017; Koopmann and Chen 2017; Penaloza et al. 2017). 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 4.** *Let $\mathcal{O}$ be an ontology, $\psi$ an observation as a set of ABox axioms, $\mathcal{F}$ a forgetting signature and $\ell$ 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 $\beta$ in the uniform interpolant $\mathcal{V} = (\mathcal{O}, \neg\psi)^{-\mathcal{F}}$, $\beta$ 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 $\ell$. 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 $\ell$. Then, $\ell$ can be replaced with $\bot$ to obtain the annotation-free set $\mathcal{V}^*_{app}$.

Since this annotation based filtering is sound, i.e., it only removes axioms that are not dependent on $\psi$, as these are directly derivable from $\mathcal{O}$ and are thus guaranteed to be redundant, it can be used at the start of Step (2) to compute $\mathcal{V}^*_{app}$. To guarantee the computation of the reduced uniform interpolant $\mathcal{V}^*$, the entailment check in Step (2) must then be performed for each axiom $\beta \in \mathcal{V}^*_{app}$ to eliminate any redundancies not captured by the annotation-based filtering. Since some axioms may have multiple derivations, they can contain the annotation concept but still be redundant with respect to Definition 1. For example:

**Example 3.** *Let $\mathcal{O} = \{A \sqsubseteq C, B \sqsubseteq C, A \sqcap D \sqsubseteq \bot, D(a)\}$ and $\psi = C(a)$. The annotated form of $\neg\psi$ is $\neg\psi = \ell \sqcup \neg C(a)$. Using $\mathcal{F} = \{C\}$, the result of Step (1) 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 $\mathcal{D} \notin \mathcal{F}$. In Step (2) extracting all axioms with annotations and setting $\ell = \bot$ gives the set $\{\neg A(a), \neg B(a)\}$. 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 1(iii). This can now be removed via the entailment check in Step (2).*

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 logics. Second, by removing axioms that are not dependent on $\psi$, the method reduces the cost of Step (2) as checking the signature of each axiom for the presence of $\ell$ is linear in the size of $\mathcal{V}$. In the worst case $\mathcal{V}^*_{app}$ could be equal to $\mathcal{V}$ and a double exponential number of entailment checks would still be required. In practice, this is unlikely as $\mathcal{V}^*_{app}$ is usually a small fraction of $\mathcal{V}$ as shown by the experiments (Table 2). In these cases, each redundancy eliminated from $\mathcal{V}$ to $\mathcal{V}^*_{app}$ replaces an exponential check with a linear one.

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 $\psi$ 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 approximate approach results in a hypothesis $\mathcal{H}_{app}$ which satisfies conditions (i), (ii) and (iv) in Definition 1, but not condition (iii). The results in Table 2 illustrate the effect in practice.

To summarise, we suggest two realisations of Step (2) of the proposed abduction method: (a) *approximate* filtering, which computes an approximation of the hypothesis $\mathcal{H}_{app}$ by negating $\mathcal{V}^*_{app}$, (b) *full* filtering, which performs the entailment check in Step (2) for each axiom in $\mathcal{V}^*_{app}$ to obtain $\mathcal{V}^*$ and thus $\mathcal{H}$ which is guaranteed to fully satisfy Definition 1. Note that for setting (b), the approximation step is still used to reduce the overall cost of Step (2).

## Experimental Evaluation

| 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 1: Characteristics of the experimental corpus.

A Java prototype was implemented using the OWL-API[1] and the forgetting tool LETHE which implements the $Int_{\mathcal{ALC}}$ method.[2] 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

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

[2]http://www.cs.man.ac.uk/ koopmanp/lethe/index.html

| Ont. Name | Mean Time Taken /s $\mathcal{V}^*_{app}$ | Mean Time Taken /s $\mathcal{V}^*$ | Mean Time Taken /s $\mathcal{V}^*$ no app. | Max Time Taken /s $\mathcal{V}^*_{app}$ | Max Time Taken /s $\mathcal{V}^*$ | Max Time Taken /s $\mathcal{V}^*$ no app. | Mean Redund. Removed $\mathcal{V} \to \mathcal{V}^*_{app}$ | Mean Redund. Removed $\mathcal{V}^*_{app} \to \mathcal{V}^*$ | Size $\mathcal{H}$ /disjuncts Mean | Size $\mathcal{H}$ /disjuncts Max | Mean % of $\mathcal{H}_{app}$ Redund. |
|---|---|---|---|---|---|---|---|---|---|---|---|
| BFO | 0.01 | 0.01 | 0.09 | 0.01 | 0.07 | 0.14 | 52 | 0 | 1.97 | 4 | 0 |
| LUBM | 0.02 | 0.03 | 0.30 | 0.11 | 0.16 | 1.21 | 90 | 0.80 | 2.73 | 11 | 29.30 |
| HOM | 0.03 | 0.05 | 0.18 | 0.40 | 0.54 | 0.86 | 82 | 0.03 | 2.07 | 13 | 1.45 |
| DOID | 0.44 | 1.09 | 1071.35 | 1.11 | 6.98 | 1095.07 | 7891 | 0 | 7.23 | 104 | 0 |
| SYN | 0.95 | 3.92 | 2421.96 | 2.33 | 61.52 | 2593.13 | 15351 | 0.03 | 20.63 | 457 | 0.15 |
| ICF | 0.30 | 0.56 | t.o. | 0.52 | 1.58 | t.o. | 8505 | 0 | 2.30 | 7 | 0 |
| Semin. | 3.13 | 5.12 | t.o. | 9.29 | 15.36 | t.o. | 72827 | 0.03 | 3.60 | 10 | 0.83 |
| OBI* | 3.82 | 32.17 | t.o. | 25.18 | 95.37 | t.o. | 29191 | 6.48 | 52.48 | 161 | 12.35 |
| NATP. | 26.54 | 179.70 | t.o. | 39.51 | 544.50 | t.o. | 111318 | 0.03 | 48.70 | 204 | 0.06 |

Table 2: Results 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. The size of $\mathcal{H}$ reported is that obtained via full computation of $\mathcal{V}^*$. Times shown are the total times taken to return $\mathcal{H}$ (or $\mathcal{H}_{app}$).

$\leq nr.C$ 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 $\exists r^-.\top \sqsubseteq C$ is converted to $\top \sqsubseteq \forall r.C$, where $r^-$ is the inverse role of $r$. 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,[3][4] and the LUBM (Guo, Pan, and Heflin 2005) and Semintec ontologies.[5] The corpus is summarised in Table 1. The experiments were performed on a machine using a 4.00GHz Intel Core i7-6700K CPU and 16GB RAM.

For each ontology, 30 consistent, non-entailed observations were randomly generated using any $\mathcal{ALC}$ concepts from the associated ontology, some of which were combined using $\mathcal{ALC}$ operators to encourage variety. The aim was to emulate the information that may be observed in practice for each ontology, while adhering to the requirements for $\psi$ expressed in Definition 1. As the current prototype uses the OWL-API, which does not allow disjunctive assertions over multiple individuals, the experiments here are limited to observations involving one individual. For the filtering in Step (2), the preference relation used in these experiments was simply based on order of appearance of each disjunct.

For the first experiment, $\mathcal{F}$ was set to one random concept symbol from $sig(\psi)$. The assumption was that users may first seek the most general hypothesis, i.e., the semantically minimal hypothesis for the largest set of abducibles. This allows the user to pursue stronger hypotheses subsequently by forgetting further symbols from the initial hypothesis. This experiment is therefore also representative of incremental abduction steps using a small $\mathcal{F}$. The second experiment was performed over the DOID, ICF and SYN ontologies to evaluate the performance as the size of $\mathcal{F}$ increases. These ontologies were used as they have a sufficiently large signature of concepts and LETHE did not time out when forgetting in any case. In all cases, at least one symbol from $\psi$ was present in $\mathcal{F}$ to avoid trivial hypotheses.

In both experiments, the approaches based on (a) *approximate* and (b) *full* filtering were compared for the same observations and same random selection of $\mathcal{F}$. Thus, the tradeoff between the additional time for entailment checking and redundancy in the final hypothesis is evaluated. In all cases, LETHE was subject to a 300 second time limit.

Table 2 shows the results for the first experiment. For the smaller 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. In all cases, it can be seen that the annotation-based filtering eliminated the majority of redundancies. In three cases (BFO, DOID, ICF), for all 30 observations the result of the approximation, $\mathcal{V}^*_{app}$, contained no redundancies and thus $\mathcal{H}_{app} = \mathcal{H}$. For the other ontologies, in most cases $\mathcal{V}^*_{app}$ contained few redundancies in both absolute terms and relative to the size of the final hypothesis. For the LUBM and OBI ontologies, however, these redundancies made up a more significant portion of $\mathcal{V}^*_{app}$.

The full filtering setting still uses the annotation-based method as a preprocessing step. To assess the benefit of this preprocessing, results for applying the entailment check in Step (2) directly to $\mathcal{V}$ instead of $\mathcal{V}^*_{app}$ were collected and are shown in the "$\mathcal{V}^*$ no app." columns. For the largest $\mathcal{EL}$ ontologies, the time taken increased significantly, e.g., taking almost 1000 times longer for the DOID ontology. For all of the $\mathcal{ALC}$ ontologies the experiments were terminated after several days runtime, i.e., it took at least several hours to compute a single hypothesis on average. This indicates that the annotation-based filtering significantly reduces the time taken, particularly over large or more expressive ontologies.

Figure 2 shows the results of the second experiment. The time taken for the forgetting step, Step (1), 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 (2), did not increase with the size of $\mathcal{F}$. However, for each ontology, maxima were observed for different sizes of $\mathcal{F}$. This implies that including

[3] https://bioportal.bioontology.org/

[4] http://www.obofoundry.org/

[5] http://www.cs.put.poznan.pl/alawrynowicz/semintec.htm

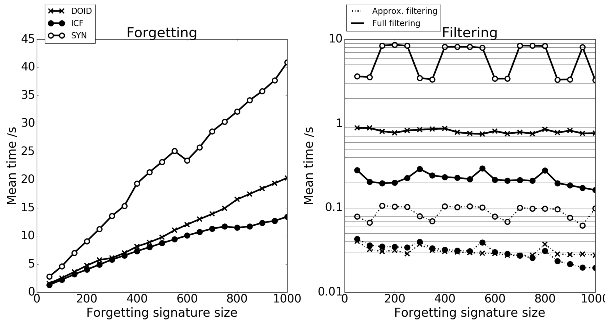


Figure 2: Mean forgetting and filtering times with varying $\mathcal{F}$ signature sizes for the ICF, DOID and SYN ontologies.

certain symbols in $\mathcal{F}$ increases the filtering time. Forgetting commonly used concepts results in more inferences and a larger $\mathcal{V}$, which may explain the maxima as the annotation-based filtering depends solely on the number of axioms in $\mathcal{V}$. The size of $\mathcal{V}^*_{app}$ will also increase in these cases, leading to more exponential entailment checks for full filtering. The full filtering took an average of 27, 11 and 70 times longer than the approximate case 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.

## Discussion

The use of forgetting for abduction has been suggested in classical logics (Doherty, Łukaszewicz, and Szałas 2001; Gabbay, Schmidt, and Szałas 2008; Wernhard 2013), and a form of TBox abduction (Koopmann and Schmidt 2015b). Our work extends on these suggestions in several ways. As the focus is on large DL ontologies, and not small theories in classical logics, an interpretable hypothesis cannot be obtained by negating the forgetting result $\mathcal{V}$ as most of it will be redundant (Table 2). Thus, we gain insight into the redundancies in $\mathcal{V}$ in terms of abductive notions, such as (Konolige 1992; Halland, Britz, and Klarman 2014), resulting in Definition 1(iii) and (iv). Efficient redundancy removal is achieved via the annotation-based filtering. The overall approach, including two options emphasising (a) practicality and (b) full redundancy removal, is then evaluated over a corpus of real-world ontologies. This gives us the first realisation and evaluation of a practical forgetting-based approach to ABox abduction in DL ontologies.

Restricting inferences in $Int_{\mathcal{ALC}}$ to axioms dependent on $\neg\psi$, rather than filtering the output, was considered. However, this would not circumvent the need to perform entailment checking, as illustrated in Example 3. Second, computing the full uniform interpolant $\mathcal{V}$ has an interesting use case: iterative abduction. For example:

**Example 4.** *Let $\mathcal{O} = \{A \sqsubseteq C, B \sqsubseteq C, C \sqsubseteq D\}$ and $\psi = D(a^*)$. In Step (1), using $\mathcal{F}$=$\{D\}$ results in $\mathcal{V}$=$\{A \sqsubseteq C, B \sqsubseteq C, \neg C(a^*)\}$. Steps (2)–(3) result in $\mathcal{H} = C(a^*)$. Now, forgetting $\mathcal{F}_2 = \{C\}$ from $\mathcal{V}$ of the previous iteration results in $\mathcal{V}_2 = \{\neg A(a^*), \neg B(a^*)\}$. Repeating Steps (2) and (3) gives $\mathcal{H}_2 = (A \sqcup B)(a^*)$, which is stronger than $\mathcal{H}$ and is the same as the result of computing the uniform interpolant of $(\mathcal{O}, \neg\psi)$ using $\mathcal{F} = \{D, C\}$, but will be more efficient.*

This iterative process enables hypothesis refinement, and has potential synergy with induction. Data could inform the selection of new forgetting signatures to find stronger hypotheses following from prior likely hypotheses: a cycle of abduction, deduction and induction.

Limitations include the lack of role assertions in the observations and hypotheses, due to the inability of $Int_{\mathcal{ALC}}$ to handle negated role assertions, and the incompleteness of role forgetting for abduction, as illustrated by the following:

**Example 5.** *Let $\mathcal{O} = \{C \sqsubseteq \exists r.D\}$ and $\psi = \exists r.D(a)$. Using $\mathcal{F} = \{r\}$ the result of Step (1) is $\mathcal{V} = \emptyset$. This is due to the fact that no inferences are possible on the symbol $D$, since resolution is restricted to $\mathcal{F}$. Thus, the hypothesis obtained is $\mathcal{H} = \emptyset$, while the expected result is $\mathcal{H} = C(a)$.*

With the use of nominals, this limitation can be overcome. Options include the use of other forgetting approaches (Zhao and Schmidt 2015; 2016) or the extension of $Int_{\mathcal{ALC}}$.

It should be noted that methods such as (Klarman, Endriss, and Schlobach 2011; Pukancová and Homola 2017) can already handle role assertions. The former is a purely theoretical work, which restricts the abductive observations and solutions to $\mathcal{ALE}$: the fragment of $\mathcal{ALC}$ without disjunctions of concepts and allowing only atomic negation. The method of (Pukancová and Homola 2017) performs abductive reasoning up to $\mathcal{ALCHO}$, restricting observations and hypotheses to atomic and negated atomic concept and role assertions. This method considers syntactic, but not semantic, minimality, though the authors note the importance of semantic minimality in practical applications.

## Conclusion and Future Work

In this paper, a practical method for ABox abduction in $\mathcal{ALC}$ ontologies was presented. The method computes semantically minimal hypotheses with independent disjuncts to explain observations, where both may contain complex $\mathcal{ALC}$ concepts but not role assertions, and the set of abducibles must contain all role symbols. The practicality of the method, including the proposed annotation-based filtering, was evaluated over a corpus of real-world ontologies. To the best of our knowledge, this is the first method that computes such hypotheses efficiently in large ontologies. The ability to produce a semantically minimal space of independent explanations will likely be beneficial in real-world applications as it can, e.g., provide engineers with multiple, non-redundant suggestions for fixing errors in an ontology or explaining negative query results, even over large knowledge bases. For scientific investigation using ontologies, the ability to produce independent avenues of explanation starting with the least assumptions necessary captures the essence of scientific hypothesis formation. The ability to refine these hypotheses via repeated forgetting also provides a goal-oriented, potentially data driven, way to derive stronger insights from the hypotheses produced.

Future work will include removing the restriction on role assertions. Also, though forgetting in DLs can be applied to a form of TBox abduction (Koopmann and Schmidt 2015b), the hypotheses take the form $\top \sqsubseteq \alpha_1 \sqcup ... \sqcup \alpha_n$ where each $\alpha$ is an $\mathcal{ALC}$ concept. Thus, the problem of determining inter-disjunct redundancy and the proposed approach differ in several aspects. This will be investigated, as will the iterative abduction use case.

## References


Bienvenu, M. 2008. Complexity of abduction in the $\mathcal{EL}$ family of lightweight description logics. In *Proc. KR'08*, 220–230. AAAI Press.

Calvanese, D.; Ortiz, M.; Simkus, M.; and Stefanoni, G. 2013. Reasoning about explanations for negative query answers in DL-Lite. *J. Artificial Intelligence Research* 48:635–669.

Calvanese, D.; De Giacomo, G.; and Lenzerini, M. 1999. Reasoning in expressive description logics with fixpoints based on automata on finite trees. In *Proc. IJCAI'99*, 84–89. AAAI Press.

Cialdea Mayer, M., and Pirri, F. 1996. Abduction is not deduction in reverse. *Journal of the IGPL* 4:95–108.

Delgrande, J.; Tompits, H.; Schaub, T.; and Kewen, W. 2004. A classification and survey of preference handling approaches in non-monotonic reasoning. *Computational Intelligence* 20:308–334.

Doherty, P.; Łukaszewicz, W.; and Szałas, A. 2001. Computing strongest necessary and weakest sufficient conditions of first-order formulas. In *Proc. IJCAI'01*, 145–151. AAAI Press.

Du, J.; Wan, H.; and Ma, H. 2017. Practical TBox abduction based on justification patterns. In *Proc. AAAI'17*, 1100–1106. AAAI Press.

Du, J.; Wang, K.; and Shen, Y. 2014. A tractable approach to ABox abduction over description logic ontologies. In *Proc. AAAI'14*, 1034–1040. AAAI Press.

Elsenbroich, C.; Kutz, O.; and Sattler, U. 2006. A case for abductive reasoning over ontologies. In *Proc. OWL: Experiences and Directions*, volume 216. CEUR Workshop Proceedings.

Gabbay, D. M.; Schmidt, R. A.; and Szałas, A. 2008. Second-order quantifier elimination: Foundations, computational aspects and applications. *College Publications* 12.

Guo, Y.; Pan, Z.; and Heflin, J. 2005. LUBM: A benchmark for OWL knowledge base systems. *J. Web Semantics* 3:158–182.

Halland, K., and Britz, K. 2012. ABox abduction in $\mathcal{ALC}$ using a DL tableau. In *Proc. SAICSIT'12*, 51–58. ACM.

Halland, K.; Britz, K.; and Klarman, S. 2014. TBox abduction in $\mathcal{ALC}$ using a DL tableau. In *Proc. DL'14*, volume 1193, 556–566. CEUR Workshop Proceedings.

Hobbs, J. R.; Stickel, M.; Martin, P.; and Edwards, D. 1993. Interpretation as abduction. *Artificial Intelligence* 63:69–142.

Kakas, A.; Kowalski, R.; and Toni, F. 1992. Abductive logic programming. *J. Logic and Computation* 2 (6):719–770.

Kazakov, Y., and Skočovský, P. 2017. Enumerating justifications using resolution. In *Proc. DL'17*, volume 1879. CEUR Workshop Proceedings.

Klarman, S.; Endriss, U.; and Schlobach, S. 2011. ABox abduction in the description logic $\mathcal{ALC}$. *J. Automated Reasoning* 46:43–80.

Konolige, K. 1992. Abduction versus closure in causal theories. *Artificial Intelligence* 53:255–272.

Koopmann, P., and Chen, J. 2017. Computing $\mathcal{ALCH}$-Subsumption modules using uniform interpolation. In *Proc. SOQE'17*, volume 2013, 51–66. CEUR Workshop Proceedings.

Koopmann, P., and Schmidt, R. A. 2013. Uniform interpolation of $\mathcal{ALC}$ ontologies using fixpoints. In *Proc. FroCoS'13*, volume 8152 of *LNCS*, 87–102. Springer.

Koopmann, P., and Schmidt, R. A. 2015a. Uniform interpolation and forgetting for $\mathcal{ALC}$ ontologies with ABoxes. In *Proc. AAAI'15*, 175–181. AAAI Press.

Koopmann, P., and Schmidt, R. A. 2015b. LETHE: Saturation based reasoning for non-standard reasoning tasks. In *Proc. ORE'15*, volume 1387, 23–30. CEUR Workshop Proceedings.

Lambrix, P.; Dragisic, Z.; and Ivanova, V. 2012. Get my pizza right: Repairing missing is-a relations in $\mathcal{ALC}$ ontologies. In *Proc. JIST'12*, volume 7774 of *LNCS*, 17–32. Springer.

Lin, F. 2001. On strongest necessary and weakest sufficient conditions. *Artificial Intelligence* 128:143–159.

Ludwig, M., and Konev, B. 2014. Practical uniform interpolation and forgetting for ALC TBoxes with applications to logical difference. In *Proc. KR'14*, 318–327. AAAI Press.

Lutz, C., and Wolter, F. 2011. Foundations for uniform interpolation and forgetting in expressive description logics. In *Proc. IJCAI'11*, 989–995. AAAI Press.

Mooney, R. 2000. Integrating abduction and induction in machine learning. In *Abduction and Induction*, 181–191. P. A. Flach and A. C. Kakas, Eds. Kluwer.

Muggleton, S., and Bryant, C. 2000. Theory completion using inverse entailment. In *Proc. ILP'00*, volume 1866 of *LNCS*, 130–146. Springer.

Peirce, C. S. 1878. Deduction, induction and hypothesis. *Popular Science Monthly* 13:470–482.

Penaloza, R.; Mencía, C.; Ignatiev, A.; and Marques-Silva, J. 2017. Lean kernels in description logics. In *Proc. ESWC'17*, volume 10249 of *LNCS*, 518–533. Springer.

Pino-Peréz, R., and Uzcátegui, C. 2003. Preferences and explanations. *Artificial Intelligence* 149:1–30.

Pukancová, J., and Homola, M. 2017. Tableau-based ABox abduction for the $\mathcal{ALCHO}$ description logic. In *Proc. DL'17*, volume 1879. CEUR Workshop Proceedings.

Raghavan, S., and Mooney, R. 2010. Bayesian abductive logic programs. In *AAAI'10 Workshop on Statistical Relational AI*, 82–87. AAAI Press.

Ray, O. 2009. Nonmonotonic abductive inductive learning. *J. Applied Logic* 7:329–340.

Stickel, M. 1991. A Prolog-like inference system for computing minimum-cost abductive explanations in natural-language interpretation. *Ann. Math. and Artificial Intelligence* 4:89–106.

Wei-Kleiner, F.; Dragisic, Z.; and Lambrix, P. 2014. Abduction framework for repairing incomplete $\mathcal{EL}$ ontologies: Complexity results and algorithms. In *Proc. AAAI'14*, 1120–1127. AAAI Press.

Wernhard, C. 2013. Abduction in logic programming as second-order quantifier elimination. In *Proc. FroCoS'13*, volume 8152 of *LNCS*, 103–119. Springer.

Zhao, Y., and Schmidt, R. A. 2015. Concept forgetting in $\mathcal{ALCOI}$-ontologies using an Ackermann approach. In *Proc. ISWC'15*, volume 9366 of *LNCS*, 587–602. Springer.

Zhao, Y., and Schmidt, R. A. 2016. Forgetting concept and role symbols in $\mathcal{ALCOIH}\mu^+(\nabla, \sqcap)$-ontologies. In *Proc. IJCAI'16*, 1345–1352. AAAI Press.