Tableau-Based ABox Abduction for the $\mathcal{ALCHO}$ Description Logic

Júlia Pukancová and Martin Homola

Comenius University in Bratislava
Mlynská dolina, 84248 Bratislava
{pukancova,homola}@fmph.uniba.sk

Abstract. Abduction is a useful decision problem that is related to diagnostics. Given some observation in form of a set of axioms, that is not entailed by a knowledge base, we are looking for explanations, sets of axioms, that can be added to the knowledge base in order to entail the observation. ABox abduction limits both observations and explanations to ABox assertions. In this work we focus on direct tableau-based approach to answer ABox abduction. We develop an ABox abduction algorithm for the $\mathcal{ALCHO}$ DL, that is based on Reiter's minimal hitting set algorithm. We focus on the class of explanations allowing atomic and negated atomic concept assertions, role assertions, and negated role assertions. The algorithm is sound and complete for this class. The algorithm was also implemented, on top of the Pellet reasoner.

Keywords: Description logics, ABox abduction, implementation.

1 Introduction

Abductive reasoning [11] focuses on deriving explanations. Given a knowledge base $\mathcal{K}$ and an observation O that is not entailed (i.e., $\mathcal{K} \not\models O$) we are looking for an explanation $\mathcal{E}$ that, when added to $\mathcal{K}$, would allow to entail O (i.e., $\mathcal{K} \cup \mathcal{E} \models O$). From the DL perspective, we distinguish between TBox abduction, where both $\mathcal{E}$ and O are limited to TBox axioms, and ABox abduction, where they are limited to ABox assertions [3]. While TBox abduction may be used, e.g., in ontology engineering, ABox abduction found uses in diagnostic reasoning [8, 13, 3], or tasks such as multimedia interpretation [12]. In our research we focus on the latter problem.

Compared to the approaches based on translations to different formalisms [10, 2], Halland and Britz [5, 4] propose a direct tableau-based approach built on top of Reiter's minimal hitting set algorithm [15]. This method avoids the translation overhead, and may also build on the existing tableau optimization techniques for DLs which have been intensively studied [17].

In our previous paper [14], we have extended the approach of Halland and Britz, and we have provided an implementation on top of the Pellet reasoner [16]. In this paper, we have further extended this work: we lifted the algorithm to the $\mathcal{ALCHO}$ DL; we have enabled support for multiple observations in form of any ABox assertions; we have also extended the explanations to include role assertions and negated role assertions; and we have proven soundness and completeness w.r.t. this class of observations and explanations.

2 ABox Abduction in DL

We build on top of the $\mathcal{ALCHO}$ DL [1]. A DL vocabulary consists of countably infinite mutually disjoint sets of individuals N_{I}, roles N_{R}, and atomic concepts N_{C}. Concepts are recursively built using constructors $\neg$, $\sqcap$, $\exists$, $\{a\}$, as shown in Table 1. Additional concepts union ($C \sqcup D := \neg(\neg C \sqcap \neg D)$) and value restriction ($\forall R.C := \neg\exists R.\neg C$) are defined as syntactic sugar; and also $\neg\neg C := C$ by definition.

A knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ consists of a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$. A TBox is a finite set of GCI and RIA axioms of the form $C \sqsubseteq D$ and $R \sqsubseteq S$, where C, D are concepts and $R, S \in N_{\mathrm{R}}$. An ABox is a finite set of concept assertions of the form $C(a)$, and role assertions of the form $R(a, b)$, where $a, b \in N_{\mathrm{I}}$, C is a concept, and $R \in N_{\mathrm{R}}$.

Table 1. $\mathcal{ALCHO}$ syntax and semantics

Concept	**Constraint**
$\neg C$ (complement)	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
$C \sqcap D$ (intersection)	$C^{\mathcal{I}} \sqcap D^{\mathcal{I}}$
$\exists R.C$ (existential restriction)	$\{x \in \Delta^{\mathcal{I}} \mid (x, y) \in R^{\mathcal{I}} \land y \in C^{\mathcal{I}}\}$
$\{a\}$ (nominal)	$\{a^{\mathcal{I}}\}$

Axiom	**Constraint**
$R \sqsubseteq S$	$R^{\mathcal{I}} \subseteq S^{\mathcal{I}}$
$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
$C(a)$	$a^{\mathcal{I}} \in C^{\mathcal{I}}$
$R(a, b)$	$(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$

An interpretation is a pair $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}} \neq \emptyset$ is a domain, and the interpretation function $\cdot^{\mathcal{I}}$ maps each individual $a \in N_{\mathrm{I}}$ to $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, each atomic concept $A \in N_{\mathrm{C}}$ to $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, each role $R \in N_{\mathrm{R}}$ to $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ in such a way that the constraints on the left-hand side of Table 1 are satisfied.

An interpretation $\mathcal{I}$ satisfies an axiom φ (denoted $\mathcal{I} \models \varphi$) if the respective constraint in Table 1 is satisfied. It is a model of a knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ (denoted $\mathcal{I} \models \mathcal{K}$) if $\mathcal{I} \models \varphi$ for all $\varphi \in \mathcal{T} \cup \mathcal{A}$. A knowledge base is consistent, if there is at least one interpretation $\mathcal{I}$ such that $\mathcal{I} \models \mathcal{K}$. A knowledge base entails an axiom φ (denoted $\mathcal{K} \models \varphi$) if $\mathcal{I} \models \varphi$ for each $\mathcal{I} \models \mathcal{K}$.

We define $\neg\varphi = \neg C(a)$ for a concept assertion $\varphi = C(a)$. Thanks to presence of nominals in $\mathcal{ALCHO}$ [7, 5] we are also able to define $\neg\varphi = \neg R(a, b) := \forall R \neg\{b\}(a)$ for a role assertion $\varphi = R(a, b)$, and $\neg\varphi := R(a, b)$ for $\varphi = \neg R(a, b)$. In addition, $\neg\mathcal{A} = \{\neg\varphi \mid \varphi \in \mathcal{A}\}$ for any set of ABox assertions $\mathcal{A}$.

In *ABox abduction*, we are given a knowledge base $\mathcal{K}$ and an observation O consisting of ABox assertions, that is, some evidence we have observed. The task is to find an explanation $\mathcal{E}$, again, consisting of ABox assertions, such that $\mathcal{K} \cup \mathcal{E} \models O$.

Definition 1 (ABox Abduction Problem [3]). *An ABox abduction problem is a pair $\mathcal{P} = (\mathcal{K}, O)$ such that $\mathcal{K}$ is a knowledge base in DL and O is a set of ABox assertions. A solution of $\mathcal{P}$ (also called explanation) is any finite set $\mathcal{E}$ of ABox assertions such that $\mathcal{K} \cup \mathcal{E} \models O$.*

Example 1. Consider the ABox abduction problem $\mathcal{P} = (\mathcal{K}, O)$, where knowledge base $\mathcal{K}$ has two axioms:

$$\mathsf{Professor} \sqcup \mathsf{Scientist} \sqsubseteq \mathsf{Academician} \tag{1}$$

$$\mathsf{AssocProfessor} \sqsubseteq \mathsf{Professor} \tag{2}$$

and we observe $O = \{\mathsf{Academician(jack)}\}$, while $\mathcal{K} \not\models O$. There is a number of explanations $\mathcal{E}_i$ s.t. $\mathcal{K} \cup \mathcal{E}_i \models O$, e.g. $\mathcal{E}_1 = \{\mathsf{Professor(jack)}\}$, $\mathcal{E}_2 = \{\mathsf{Scientist(jack)}\}$, $\mathcal{E}_3 = \{\mathsf{Professor(jack)}, \mathsf{Scientist(jack)}\}$, $\mathcal{E}_4 = \{\mathsf{AssocProfessor(jack)}\}$, and even $\mathcal{E}_5 = \{\mathsf{Academician(jack)}\}$.

While Definition 1 establishes the basic reasoning mechanism of abduction, some of the explanations it permits are clearly undesired. The explanations should, at minimum, fulfil some basic sanity requirements.

Definition 2 ([3]). *Given an ABox abduction problem $\mathcal{P} = (\mathcal{K}, O)$ and its solution $\mathcal{E}$ we say that:*

1. *$\mathcal{E}$ is* consistent *if $\mathcal{E} \cup \mathcal{K} \not\models \bot$, i.e. $\mathcal{E}$ is consistent w.r.t. $\mathcal{K}$;*
2. *$\mathcal{E}$ is* relevant *if $\mathcal{E} \not\models O$, i.e. $\mathcal{E}$ does not entail O;*
3. *$\mathcal{E}$ is* explanatory *if $\mathcal{K} \not\models O$, i.e. $\mathcal{K}$ does not entail O.*

An explanation should be consistent, as anything follows from inconsistency; and so, an explanation that makes $\mathcal{K}$ inconsistent does not really explain the observation. It should be relevant – it should not imply the observation directly without requiring the knowledge base $\mathcal{K}$ at all. And it should be explanatory, that is, we should not be able to explain the observation without it.

Example 2. Consider the ABox abduction problem $\mathcal{P} = (\mathcal{K}, O)$ and its solutions $\mathcal{E}_1, \ldots, \mathcal{E}_5$ from Example 1. The explanations $\mathcal{E}_1, \ldots, \mathcal{E}_4$ are consistent, relevant, and explanatory. However, $\mathcal{E}_5 = \{\mathsf{Academician(jack)}\}$ is not relevant, since $\mathcal{E}_5 \models O$.

Hereafter, when we say explanation we always mean a consistent, relevant, and explanatory explanation, unless indicated otherwise. Subsequently we can think about further requirements to eliminate undesired explanations. Usually it is clear that we want to explain observations only with sufficient assumptions and not to hypothesize too much. Therefore syntactic minimality is defined.

Definition 3 (Syntactic Minimality). *Assume an ABox abduction problem $\mathcal{P} = (\mathcal{K}, O)$. Given two solutions $\mathcal{E}$ and $\mathcal{E}'$ of $\mathcal{P}$, we say that $\mathcal{E}$ is (syntactically)* smaller *than* $\mathcal{E}'$ *if* $\mathcal{E} \subseteq \mathcal{E}'$.[1] *We further say that a solution* $\mathcal{E}$ *of* $\mathcal{P}$ *is* syntactically minimal *if there is no other solution* $\mathcal{E}'$ *of* $\mathcal{P}$ *that is smaller than* $\mathcal{E}$.

Example 3. Consider the ABox abduction problem $\mathcal{P} = (\mathcal{K}, O)$ from Example 2. Four explanatory, consistent, and relevant explanations were found. We may observe that $\mathcal{E}_1 = \{\mathsf{Professor(jack)}\}$ and $\mathcal{E}_2 = \{\mathsf{Scientist(jack)}\}$ are both smaller than $\mathcal{E}_3 = \{\mathsf{Professor(jack)}, \mathsf{Scientist(jack)}\}$, i.e. $\mathcal{E}_1 \subseteq \mathcal{E}_3$ and $\mathcal{E}_2 \subseteq \mathcal{E}_3$. Therefore $\mathcal{E}_3$ is not a syntactically minimal explanation, while the other three are.

3 Our Approach

Based on Reiter's work [15], on the proposal of Halland and Britz [5, 4], and on our previous works [13, 14] we define an ABox abduction algorithm for $\mathcal{ALCHO}$.

For a single observation O, a solution of an abduction problem $\mathcal{P} = (\mathcal{K}, O)$ according to Definition 1 can be obtained as any $\mathcal{E}$ s.t. $\mathcal{K} \cup \mathcal{E} \cup \{\neg O\}$ is inconsistent (due to reducibility of entailment into consistency checking [1]). As showed by Reiter [15], we can compute the minimal explanations of $\mathcal{P}$ by finding all minimal hitting sets for all models of $\mathcal{K} \cup \{\neg O\}$.

Definition 4 (Minimal Hitting Set [15, 9]). *A hitting set for a collection of sets* F *is a set* H *s.t.* $H \cap S \neq \{\}$ *for every* $S \in F$. *A hitting set* H *for* F *is minimal if there is no other hitting set for* F *s.t.* $H' \subsetneq H$.

Definition 5 (HS-Tree [15]). *A HS-tree for* F *is* $T = (V, E, L, H)$, *where* (V, E) *is a minimal tree in which the labelling function* L *labels the nodes of* V *by elements of* F, *the edges of* E *by elements of sets in* F, *and* $H(n)$ *is the set of edge-labels from the root node to* $n \in V$, *s.t.: (a) for the root* $r \in V$: $L(r) = S$ *for some* $S \in F$, *if* $F \neq \{\}$, *otherwise* $L(r) = \{\}$; *(b) for each* $n \in V$: $L(n) = S$ *for some* $S \in F$ *s.t.* $S \cap H(n) = \{\}$, *if such* $S \in F$ *exists, otherwise* $L(n) = \{\}$; *(c) each* $n \in V$ *has a successor* n_σ *for each* $\sigma \in L(n)$ *with* $L(n, n_\sigma) = \sigma$.

A HS-tree T for F contains all minimal hitting sets but it may contain some other hitting sets. For sake of optimization Reiter [15] proposed to construct HS-tree by breadth-first search and to prune it as follows: a node $n \in V$ is *pruned* if there is $n' \in V$ s.t.: (a) either $H(n') \subseteq H(n)$ and $L(n') = \{\}$ (i.e., $H(n)$ is not minimal because its subset $H(n')$ is also a hitting set); (b) or $H(n') = H(n)$ and $L(n') = S \in F$ (i.e., both paths are equivalent therefore we can prune one of them). A pruned HS-tree is obtained from a HS-tree by removing all pruned nodes including their descendants.

Theorem 1 (Reiter [15]). *Let* $T = (V, E, L, H)$ *be a pruned HS-tree for a collection of sets* F. *Then* $\{H(n) \mid n \in V,\ L(n) = \{\}$, *and* n *is not pruned*$\}$ *is the collection of all minimal hitting sets for* F.

[1] Note that before we compare two solutions $\mathcal{E}$ and $\mathcal{E}'$ of $\mathcal{P}$ syntactically, we typically normalize the assertions w.r.t. (outermost) concept conjunction: as $C \sqcap D(a)$ is equivalent to the pair of assertions $C(a)$ and $D(a)$, we replace the former form by the latter while possible.

3.1 Single Concept Observation

The algorithm is given in Algorithm 1. It starts by calling the tableau algorithm (TA) on $\mathcal{K}' = \mathcal{K} \cup \{\neg O\}$. If there is no model, the observation already follows from $\mathcal{K}$ – there is nothing to explain. If there is a model $\mathcal{I}$, we extract all atomic and negated atomic concept assertions from it into $M = \{C(a) \mid \mathcal{I} \models C(a),\, C \in \{A, \neg A\},\, A \in N_{\mathrm{C}},\, a \in N_{\mathrm{I}}\}$, and store this representation of M for later reuse (lines 1–6). Note, hereafter whenever we refer to a model M we mean this representation of it.

A new HS-tree $T = (V, E, L, H)$ is then initialized with root $r \in V$, labelled by $\neg M$. And, a successor node is added to V for each $\sigma \in \neg M$, together with a respective edge labelled by σ (lines 7–9).

The root node r is now fully processed. We initialize the output set of explanations as $\mathcal{S}_{\mathcal{E}} = \{\}$ and traverse the remaining nodes in V by breadth-first search (lines 10–30).

For each such node n, we first evaluate $H(n)$ and check if pruning can be applied: if there is a clash within $H(n)$, or if some $S \subseteq H(n)$ is already in $\mathcal{S}_{\mathcal{E}}$, or there is some $n' \in V$ such that $H(n') = H(n)$, we are not interested in $H(n)$ and so we label n by $\{\}$ (lines 13–14).

If none of this is the case, we try to find a model of $\mathcal{K}' \cup H(n)$. We first try to reuse a suitable model M which was previously computed (line 16). If there is none, we call TA on $\mathcal{K}' \cup H(n)$. If we obtain a model $\mathcal{I}$ we compute its representation M and store it for later reuse (lines 18–20). We then label n by $\neg M$ and initialize its successors and respective edge-labels similarly as for the root node (lines 27–29).

If no model was returned by TA, then $H(n)$ is a candidate explanation: we add it to $\mathcal{S}_{\mathcal{E}}$, if it is consistent and relevant. In this case n is also labelled by $\{\}$ (lines 21–24).

Once we traversed all nodes in T a minimal HS-tree is constructed and $\mathcal{S}_{\mathcal{E}}$ contains all minimal explanations (line 30).

3.2 Role Assertions

We will now describe how Algorithm 1 is extended to allow also role assertions, including negated, as the observation and also in the explanations. The case of the observation is trivial thanks to the choice of DL with nominals: we can simply permit the observation O on the input to be also in the form $R(a, b)$ or $\neg R(a, b)$ for $R \in N_{\mathrm{R}}$, $a, b \in N_{\mathrm{I}}$. This only affects line 1 where $\neg O$ is computed as given in Section 2.

In order to include role assertions also in the explanations we need to modify the construction of the model M which was previously described in Section 3.1. Given the model $\mathcal{I}$ returned by the TA, we redefine its representation M as:

$$\begin{aligned} M = {} & \{C(a) \mid \mathcal{I} \models C(a),\, C \in \{A, \neg A\},\, A \in N_{\mathrm{C}},\, a \in N_{\mathrm{I}}\} \\ & \cup \{R(a, b) \mid \mathcal{I} \models R(a, b),\, R \in N_{\mathrm{R}},\, a, b \in N_{\mathrm{I}}\} \\ & \cup \{\neg R(a, b) \mid \mathcal{I} \models \neg R(a, b),\, R \in N_{\mathrm{R}},\, a, b \in N_{\mathrm{I}}\}\,. \end{aligned} \tag{3}$$

Algorithm 1 SOA($\mathcal{K}$,O): Single Observation Abduction

Input: knowledge base $\mathcal{K}$, observation O
Output: set of all explanations $\mathcal{S}_\mathcal{E}$

$\mathcal{K}' \leftarrow \mathcal{K} \cup \{\neg O\}$
$M \leftarrow$ call TA with input $\mathcal{K}'$ ▷ TA returns a model of $\mathcal{K}'$
if $M = \{\}$ **then**
  return `"nothing to explain"`
end if
$MS \leftarrow \{M\}$
create new HS-tree $T = (V, E, L, H)$ with root r
$L(r) \leftarrow \neg M$
for each $\sigma \in \neg M$ create a successor n_σ of r and label the resp. edge by σ
$\mathcal{S}_\mathcal{E} \leftarrow \{\}$
$n \leftarrow$ next node w.r.t. r in T by breadth-first search
while $n \neq$ null **do**
  if (clash in $H(n)$) ▷ $H(n)$ – set of edge-labels on path r–n
    or ($S \in \mathcal{S}_\mathcal{E}$ **and** $S \subseteq H(n)$)
    or ($n' \in T$ **and** $H(n) = H(n')$ **and** $L(n') \neq$ null) **then**
    $M \leftarrow \{\}$ ▷ prune the path
  else if $N \in MS$ **and** $H(n) \subseteq N$ **then**
    $M \leftarrow N$ ▷ reuse model
  else
    $M \leftarrow$ call TA for $\mathcal{K}' \cup H(n)$
    if $M \neq \{\}$ **then**
      $MS \leftarrow MS \cup \{M\}$ ▷ store the model for later reuse
    else ▷ the case $H(n)$ explains O
      if $H(n)$ is relevant and consistent explanation **then**
        $\mathcal{S}_\mathcal{E} \leftarrow \mathcal{S}_\mathcal{E} \cup \{H(n)\}$
      end if
    end if
  end if
  $L(n) \leftarrow \neg M$
  for each $\sigma \in \neg M$ create a successor n_σ of n and label the resp. edge by σ
  $n \leftarrow$ next node in T w.r.t. n by breadth-first search
end while
return $\mathcal{S}_\mathcal{E}$

The first part involving concept assertions is unchanged, plus we also add all role assertions and negated role assertions that hold in $\mathcal{I}$. Note, from now on, whenever we talk about model M we mean the representation in this form. The actual extraction of the models from TA is described below in Section 5.

Example 4. Consider the knowledge base $\mathcal{K}$:

$$\mathsf{SlovakScientist} \sqsubseteq \exists \mathsf{livesIn}.\{\mathsf{slovakia}\} \tag{4}$$

$$\mathsf{coauthors} \sqsubseteq \mathsf{workWith} \tag{5}$$

Given the observation $O_1 = \{\mathsf{livesIn}(\mathsf{jack}, \mathsf{slovakia})\}$ and the ABox abduction problem $\mathcal{P}_1 = (\mathcal{K}, O_1)$ we are able to find one consistent, relevant, explanatory, and subset minimal explanation $\mathcal{E}_1 = \{\mathsf{SlovakScientist(jack)}\}$ of $\mathcal{P}_1$, i.e., $\mathcal{K} \cup \mathcal{E}_1 \models O_1$. Similarly, given the observation $O_2 = \{\mathsf{workWith(jack, mary)}\}$, $\mathcal{E}_2 = \{\mathsf{coauthors(jack, mary)}\}$ is an explanation of $\mathcal{P}_2 = (\mathcal{K}, O_2)$, i.e., $\mathcal{K} \cup \mathcal{E}_2 \models O_2$.

3.3 Multiple Observations

Consider the abduction problem $\mathcal{P} = (\mathcal{K}, O)$ with a set of observations $O = \{O_1, \ldots, O_n\}$. The problem $\mathcal{P}$ can be simply split into n subproblems $\mathcal{P}_1 = (\mathcal{K}, O_1)$, $\ldots$, $\mathcal{P}_n = (\mathcal{K}, O_n)$. Observe that if $\mathcal{K} \cup \mathcal{E}_i \models O_i$ for $1 \leq i \leq n$, then $\mathcal{E} = \mathcal{E}_1 \cup \ldots \cup \mathcal{E}_n$ is an explanation of $\mathcal{P}$, i.e., $\mathcal{K} \cup \mathcal{E} \models O$.

Hence, in order to compute all explanations for $\mathcal{P} = (\mathcal{K}, O)$ we need to compute all combinations from explanations of every observation O_i. The algorithm is described more precisely in Algorithm 2.

Example 5. Consider the knowledge base $\mathcal{K}$ from the Example 4 and the observation $O = \{\mathsf{livesIn(jack, slovakia)}, \mathsf{workWith(jack, mary)}\}$. We are looking for $\mathcal{E}$ s.t. $\mathcal{K} \cup \mathcal{E} \models O$. We split $\mathcal{P} = (\mathcal{K}, O)$ into two subproblems, where $\mathcal{P}_1 = (\mathcal{K}, O_1)$, $O_1 = \{\mathsf{livesIn(jack, slovakia)}\}$ and $\mathcal{P}_2 = (\mathcal{K}, O_2)$, $O_2 = \{\mathsf{workWith(jack, mary)}\}$. Actually, $\mathcal{P}_1$ and $\mathcal{P}_2$ are already solved in Example 4; the set of all explanations for O_1 is $\mathcal{S}_{\mathcal{E}_1} = \{\mathcal{E}_1\}$ and for O_2 is $\mathcal{S}_{\mathcal{E}_2} = \{\mathcal{E}_2\}$. In this very simple example, $\mathcal{S}_{\mathcal{E}}^{\min} = \{\mathcal{E}_1 \cup \mathcal{E}_2\}$ and so the only solution of $\mathcal{P}$ is $\mathcal{E} = \{\mathsf{SlovakScientist(jack)}, \mathsf{coauthors(jack, mary)}\}$.

Note that there are some special cases the algorithm needs to observe. Firstly, if there is no explanation $\mathcal{E}_i$ that explains $\mathcal{P}_i$ for at least one i, then there is also no explanation that explains $\mathcal{P}$ (lines 4–5).

Secondly, if $\mathcal{K} \models O_i$ then $\mathcal{P}_i$ does not contribute to $\mathcal{E}$, as given more precisely in Observation 1.

Observation 1. *Given a set of observations $O = \{O_1, \ldots, O_n\}$ and given $O' = O \setminus \{O_i \in O \mid \mathcal{K} \models O_i\}$, we have $\mathcal{K} \cup \mathcal{E} \models O$ if and only if $\mathcal{K} \cup \mathcal{E} \models O'$. Hence $\mathcal{P} = (\mathcal{K}, O)$ has the same set of explanations as $\mathcal{P} = (\mathcal{K}, O')$.*

The algorithm uses the set Σ to collect explanations only for $O_i \in O'$ (lines 6–7).

Thirdly, even if the explanation $\mathcal{E}_i$ of $\mathcal{P}_i$ is minimal (consistent and relevant) for all $1 \leq i \leq n$, it is not guaranteed that $\mathcal{E} = \mathcal{E}_1 \cup \ldots \cup \mathcal{E}_n$ is minimal, consistent and relevant. Therefore we need to filter out the undesired explanations additionally (line 14).

4 Soundness and Completeness

Lemma 1. *Let $\mathcal{K}$ be an $\mathcal{ALCHO}$ knowledge base and let O be an observation in form of an $\mathcal{ALCHO}$ ABox assertion. Let $\mathcal{S}_{\mathcal{E}}$ be the output of the SOA algorithm initialized with $\mathcal{K}$ and O on the input. Then each $\mathcal{E} \in \mathcal{S}_{\mathcal{E}}$ is a consistent, relevant, explanatory, and subset minimal explanation of the abduction problem $\mathcal{P} = (\mathcal{K}, O)$.*

Algorithm 2 AAA($\mathcal{K}$,O): ABox Abductive Algorithm

Input: knowledge base $\mathcal{K}$, set of observations O
Output: set of all minimal explanations $\mathcal{S}_{\mathcal{E}}^{\min}$

$\Sigma \leftarrow \{\}$ ▷ collection of the sets of explanations for all observations
for all $O_i \in O$ **do**
  $\mathcal{S}_{\mathcal{E}_i} \leftarrow \mathrm{SOA}(\mathcal{K}, O_i)$ ▷ set of explanations for the subproblem O_i
  if $\mathcal{S}_{\mathcal{E}_i} = \{\}$ **then**
    return $\{\}$ ▷ if O_i has no explanation then neither O has
  else if $\mathcal{S}_{\mathcal{E}_i} \neq$ `"nothing to explain"` **then** ▷ $\mathcal{K} \models O_i$ – exclude O_i
    $\Sigma \leftarrow \Sigma \cup \{\mathcal{S}_{\mathcal{E}_i}\}$
  end if
end for
if $\Sigma = \{\}$ **then**
  return `"nothing to explain"`
else
  $\mathcal{S}_{\mathcal{E}} \leftarrow \{\mathcal{E}_1 \cup \ldots \cup \mathcal{E}_m \mid \mathcal{E}_i \in \mathcal{S}_{\mathcal{E}_i}, \mathcal{S}_{\mathcal{E}_i} \in \Sigma, m = |\Sigma|\}$ ▷ all combinations of the expl.
  $\mathcal{S}_{\mathcal{E}}^{\min} \leftarrow \{\mathcal{E} \mid \mathcal{E} \in \mathcal{S}_{\mathcal{E}}$ and $\forall \mathcal{E}' \in \mathcal{S}_{\mathcal{E}} : \mathcal{E}' \not\subseteq \mathcal{E}$ and $\mathcal{E}$ is consistent and relevant$\}$
end if
return $\mathcal{S}_{\mathcal{E}}^{\min}$

Proof. If SOA returned `"nothing to explain"`, it terminated in line 4, and this was because $\mathcal{K} \cup \{\neg O\}$ was inconsistent, which is the same as $\mathcal{K} \models O$, and in such a case there are no explanations.

In the other case SOA returned a set $\mathcal{S}_{\mathcal{E}}$. Let $\mathcal{E} \in \mathcal{S}_{\mathcal{E}}$. In such a case $\mathcal{E} = H(n)$ for some node n and it was added to $\mathcal{S}_{\mathcal{E}}$ on line 23. However in this case we have also called TA on $\mathcal{K} \cup \{\neg O\} \cup \mathcal{E}$ on line 18 and it returned no model (as we tested on line 19). Hence $\mathcal{K} \cup \mathcal{E} \models O$, i.e., $\mathcal{E}$ is an explanation of $\mathcal{P}$.

In addition, $\mathcal{E}$ is consistent and relevant, because we have tested this (on line 22) immediately before adding it to $\mathcal{S}_{\mathcal{E}}$. It is also explanatory because in the other case the algorithm returned `"nothing to explain"` and terminated already on line 4 as described above.

The minimality of $\mathcal{E}$ follows from the fact that we only add such $\mathcal{E} = H(n)$ into $\mathcal{S}_{\mathcal{E}}$ on line 23 which correspond to paths from root to a leaf which are not pruned in the HS-tree, and as showed by Reiter [15], in a pruned HS-tree all such paths correspond to minimal hitting sets. This can be verified by observing the HS-tree is constructed breadth-first, that is, when $\mathcal{E} = H(n)$ is considered as an explanation, all smaller explanations are already stored in $\mathcal{S}_{\mathcal{E}}$. Consequently if some $S \subseteq H(n)$ was previously found, the if-condition on line 13 is evaluated as true and hence the assignment of $\mathcal{E}$ into $\mathcal{S}_{\mathcal{E}}$ on line 23 is not executed.

Theorem 2 (Soundness). *Let $\mathcal{K}$ be an $\mathcal{ALCHO}$ knowledge base and let O be a set of observations in form of $\mathcal{ALCHO}$ ABox assertions. Let $\mathcal{S}_{\mathcal{E}}$ be the output of the AAA algorithm initialized with $\mathcal{K}$ and O on the input. Then each $\mathcal{E} \in \mathcal{S}_{\mathcal{E}}$ is a consistent, relevant, explanatory, and subset minimal explanation of the abduction problem $\mathcal{P} = (\mathcal{K}, O)$.*

Proof. If AAA returned `"nothing to explain"`, it was because on line 10 collection of the sets of all explanations Σ was empty. This can only be the case when SOA returned `"nothing to explain"` for each O_i (line 6), that is, according to Observation 1, $O' = \{\}$. This means that $\mathcal{K} \models O_i$ for each O_i, and so $\mathcal{K} \models O$.

In the other case AAA returned a set $\mathcal{S}_\mathcal{E}$. Let $\mathcal{E} \in \mathcal{S}_\mathcal{E}$. From lines 13–14 it is apparent that $\mathcal{E} = \mathcal{E}_1 \cup \ldots \cup \mathcal{E}_m$ where $\mathcal{E}_i \in \mathcal{S}_{\mathcal{E}_i}$ and each $\mathcal{S}_{\mathcal{E}_i} \in \Sigma$ is the set of minimal explanations for O_i returned by SOA on line 3.

From Lemma 1 we have $\mathcal{K} \cup \mathcal{E}_i \models O_i$ for all $\mathcal{E}_i \in \mathcal{S}_{\mathcal{E}_i}$. Observe, that Σ collects $\mathcal{S}_{\mathcal{E}_i}$ for all those O_i, for which $\mathcal{K} \not\models O_i$ (lines 6–7), hence from Observation 1 we have $\mathcal{K} \cup \mathcal{E} \models O$, that is $\mathcal{E}$ is an explanation of $\mathcal{P} = (\mathcal{K}, O)$. Moreover, subset minimality, consistency, and relevancy of each $\mathcal{E} \in \mathcal{S}_\mathcal{E}$ is consecutively verified on line 14. $\mathcal{E}$ is also explanatory, as otherwise $\mathcal{K} \models O$, i.e., $\mathcal{K} \models O_i$ for all i, and thus $\Sigma = \{\}$. In such a case the algorithm already terminates on line 11.

Lemma 2. *Let $\mathcal{K}$ be an $\mathcal{ALCHO}$ knowledge base and let O be an observation in form of an $\mathcal{ALCHO}$ ABox assertion. Let $\mathcal{E} \subseteq \{A(a), \neg A(a), R(a,b), \neg R(a,b) \mid A \in N_\mathrm{C}, R \in N_\mathrm{R}, a, b \in N_\mathrm{I}\}$ be a consistent, relevant, explanatory, and subset minimal explanation of the abduction problem $\mathcal{P} = (\mathcal{K}, O)$. Then the SOA algorithm, initialized with $\mathcal{K}$ and O on the input, produces $\mathcal{E}$ as one of its outputs.*

Proof. Given an abduction problem $\mathcal{P} = (\mathcal{K}, O)$, let $\mathcal{S}_\mathcal{E}$ be an output of SOA for $\mathcal{P}$. Let $\mathcal{E}$ be a consistent, relevant, explanatory, and subset minimal explanation of $\mathcal{P}$.

As $\mathcal{E}$ is explanatory, $\mathcal{K} \cup \{\neg O\}$ has at least one model. Hence the root r of HS-tree T constructed by SOA is labelled by $\neg M$, where M is a model of $\mathcal{K} \cup \{\neg O\}$ (line 8). Note that, from the construction of M (3) it follows that $\varphi \in \neg M$ or $\neg\varphi \in \neg M$ for every atomic ABox assertion φ.

It is clear that $\mathcal{K} \cup M \cup \{\neg O\}$ is consistent and so $\mathcal{E} \not\subseteq M$, i.e. there is an ABox assertion $\sigma_1 \in \mathcal{E}$ s.t. $\sigma_1 \notin M$. Hence $\neg\sigma_1 \in M$, and so $\sigma_1 \in \neg M$, and also $L(r, n_{\sigma_1}) = \sigma_1$ for some successor n_{σ_1} of r, from line 9.

The rest of the proof is by induction. Let us assume that SOA extended T until there is a node n_{σ_k} s.t. $H(n_{\sigma_k}) \subseteq \mathcal{E}$ and $|H(n_{\sigma_k})| = k$. We will show that $(*)$ either $H(n_{\sigma_k}) = \mathcal{E}$ or there is some $\sigma_{k+1} \in \mathcal{E} \setminus H(n_{\sigma_k})$ which will become the label of some new edge leading from n_{σ_k}. Observe that none of the pruning conditions applies on n_{σ_k}: $\mathcal{E}$ is consistent, and hence $H(n_{\sigma_k})$ does not contain a clash; no $S \subsetneq H(n_{\sigma_k})$ was previously added into $\mathcal{S}_\mathcal{E}$ because $\mathcal{E}$ is minimal; and if there is some other node n in T such that $H(n) = H(n_{\sigma_k})$ we can assume w.l.o.g. that n_{σ_k} is the one which is visited first and hence it is not pruned. Next we distinguish between two cases. In the first case $\mathcal{K} \cup H(n_{\sigma_k}) \cup \{\neg O\}$ is inconsistent, and so $H(n_{\sigma_k}) = \mathcal{E}$, $L(n_{\sigma_k}) = \{\}$ and therefore SOA adds $H(n_{\sigma_k}) = \mathcal{E}$ into $\mathcal{S}_\mathcal{E}$. In the second case $\mathcal{K} \cup H(n_{\sigma_k}) \cup \{\neg O\}$ is consistent, i.e. it has a model M_k and $L(n_{\sigma_k}) = \neg M_k$. It is clear that $H(n_{\sigma_k}) \subseteq M_k$ and that $\mathcal{K} \cup M_k \cup \{\neg O\}$ is consistent and so $\mathcal{E} \not\subseteq M_k$, i.e. there is $\sigma_{k+1} \in \mathcal{E} \setminus H(n_{\sigma_k})$ s.t. $\sigma_{k+1} \notin M_k$, and so $\sigma_{k+1} \in \neg M_k$. Therefore SOA consequently creates a node $n_{\sigma_{k+1}}$ with $L(n_{\sigma_k}, n_{\sigma_{k+1}}) = \sigma_{k+1}$.

Since we have proved (∗) for any k, by induction SOA will eventually create a node n_{σ_m} s.t. $H(n_{\sigma_m}) = \mathcal{E}$ and $L(n_{\sigma_m}) = \{\}$. That means that SOA has added $H(n_{\sigma_m}) = \mathcal{E}$ into $\mathcal{S}_\mathcal{E}$.

Theorem 3 (Completeness). *Let $\mathcal{K}$ be an $\mathcal{ALCHO}$ knowledge base and let O be an observation set of $\mathcal{ALCHO}$ ABox assertions. Let $\mathcal{E} \subseteq \{A(a), \neg A(a), R(a,b), \neg R(a,b) \mid A \in N_\mathrm{C}, R \in N_\mathrm{R}, a, b \in N_\mathrm{I}\}$ be a consistent, relevant, explanatory, and subset minimal explanation of the abduction problem $\mathcal{P} = (\mathcal{K}, O)$. Then the AAA algorithm, initialized with $\mathcal{K}$ and O on the input, produces $\mathcal{E}$ as one of its outputs.*

Proof. Given an abduction problem $\mathcal{P} = (\mathcal{K}, O)$, let $\mathcal{S}_\mathcal{E}^{\min}$ be the output of AAA for $\mathcal{P}$. Let $\mathcal{E}$ be a consistent, relevant, explanatory, and subset minimal explanation of $\mathcal{P}$.

As $\mathcal{K} \cup \mathcal{E} \models O = \{O_1, \dots, O_n\}$, then also $\mathcal{K} \cup \mathcal{E} \models O_i$ and hence $\mathcal{E}$ explains each $\mathcal{P}_i = (\mathcal{K}, O_i)$.

Let $\mathcal{E}_i$ be a smallest subset of $\mathcal{E}$ that explains $\mathcal{P}_i$. For some i's $\mathcal{E}_i$ may equal to $\{\}$ but not for all, as then $\mathcal{K} \models O$ which is not the case. If $\mathcal{E}_i \neq \{\}$ then $\mathcal{E}_i$ is a minimal explanation of $\mathcal{P}_i$, otherwise it would not be a smallest subset of $\mathcal{E}$ that explains $\mathcal{P}_i$. It is trivially explanatory, and it is also consistent and relevant w.r.t. $\mathcal{P}_i$ (because whole $\mathcal{E}$ is). In addition $\mathcal{E} = \mathcal{E}_1 \cup \cdots \cup \mathcal{E}_n$ because $\mathcal{E}_1 \cup \cdots \cup \mathcal{E}_n$ explains all $O_1, \dots, O_n$ hence otherwise $\mathcal{E}$ would not be minimal.

Now, AAA called SOA and obtained the set of explanations $\mathcal{S}_{\mathcal{E}_i}$ for each $\mathcal{P}_i$. From Lemma 2 we have that $\mathcal{S}_{\mathcal{E}_i}$ contains all minimal, consistent, relevant, and explanatory explanations of $\mathcal{P}_i$. Hence on line 13 $\mathcal{E}$ was surely added into $\mathcal{S}_{\mathcal{E}_i}$. But since $\mathcal{E}$ is minimal, consistent, relevant, and explanatory, it was also added to $\mathcal{S}_\mathcal{E}^{\min}$ on line 14.

5 Implementation

Our algorithm is implemented in Java. It is based on our previous implementation that is extended with more forms of observations and explanations. As described in our previous work [14], knowledge base consistency is verified by the Pellet reasoner [16] (version 2.3.1). Pellet is an optimized tableau-based reasoner, that will enable to involve optimizations such as incremental reasoning in our implementation in the future. The run of our algorithm corresponds to Algorithm 2.

OWL ontology and the set of ABox assertions representing the observations are loaded as input and consequently processed into Pellet knowledge base $\mathcal{K}$. Firstly, Pellet decides a consistency of $\mathcal{K}$. After a successful consistency check, we obtain the ABox $\mathcal{A}$ corresponding to the model of $\mathcal{K}$ using the `getABox()` method and construct M as described in Section 3.2. For each node a we extract from $\mathcal{A}$ all atomic concepts in its label and all outgoing edges (using the `getTypes()` and `getOutEdges()` methods), and we add the corresponding ABox assertions to M. Consequently, we compute the completion of M by adding the negation of all atomic assertions which are not already in M.

Our implementation uses Pellet for consistency checking and for model construction. All other features, from initializing AAA algorithm through constructing the HS-tree to answering the set of all minimal hitting sets for the input set of observations, are executed by our own implementation. All optimizations presented in this paper, such as HS-tree pruning or model reusing, are implemented.

Our implementation is available for download at: `http://dai.fmph.uniba.sk/~pukancova/aaa/`.

6 Conclusions

We have described an ABox abduction algorithm for the $\mathcal{ALCHO}$ DL, which is based on Reiter's work on minimal hitting sets [15]. The algorithm calls a DL reasoner as a black box; the current approach is tableau-based as we rely on Pellet reasoner.

Our algorithm permits a set of any ABox assertions (including negated role assertions) as the observation, and computes explanations constrained to minimal sets of atomic and negated atomic ABox assertions. Our work extends the works of Halland and Britz [5, 4] in the following respects: (a) permitting $\mathcal{ALCHO}$ instead of $\mathcal{ALC}$; (b) computing models on the fly, during the search for explanations, instead of during pre-processing; (c) formally establishing both soundness and completeness for the given class of observations and explanations; and (d) developing also an implementation built on top of the Pellet [16] reasoner. Compared to our previous work [14], we have extended the algorithm as well as the implementation to handle multiple observations and to permit role assertions in both observations and explanations, and we have established soundness and completeness.

Regarding the complexity, $\mathcal{ALCHO}$ is ExpTime-complete [6], and Reiter's minimal hitting set algorithm is NP-complete [15, 9], so the combined overall complexity of our algorithm is still ExpTime. In the future, we also plan to further optimize the implementation of our algorithm, especially by exploiting incremental reasoning that is partly available in Pellet 2.3.1. Pellet enables to reuse the previously built tableau structures in cases when assertions are added to the ABox, however it is not able to handle deletions. We plan to exploit techniques similar to tableau caching to deal with the latter problem.

In the future, we would like to consider also semantically minimal explanations [14]. In our opinion, this notion of minimality is highly relevant for practical problems. We also want to focus on explanations involving complex concept assertions, and anonymous individuals, as there are some specific abduction problems where such explanations are interesting. This could be difficult because of the possibility of infinitely many solutions. It is necessary to specify the concrete target forms of these explanations and to propose new form of model representations including also complex concept assertion. As the complexity of the algorithm will increase, we need to analyse also other possible optimizations.

Acknoledgements. We would like to thank to the anonymous reviewers for a number of fruitful comments.

References

1. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F. (eds.): The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press (2003)
2. Du, J., Qi, G., Shen, Y., Pan, J.Z.: Towards practical ABox abduction in large description logic ontologies. Int. J. Semantic Web Inf. Syst. 8(2), 1–33 (2012)
3. Elsenbroich, C., Kutz, O., Sattler, U.: A case for abductive reasoning over ontologies. In: Proceedings of the OWLED*06 Workshop on OWL: Experiences and Directions, Athens, Georgia, USA, November 10-11, 2006 (2006)
4. Halland, K., Britz, K.: Abox abduction in $\mathcal{ALC}$ using a DL tableau. In: 2012 South African Institute of Computer Scientists and Information Technologists Conference, SAICSIT '12, Pretoria, South Africa, October 1-3, 2012. pp. 51–58 (2012)
5. Halland, K., Britz, K.: Naïve ABox abduction in $\mathcal{ALC}$ using a DL tableau. In: Proceedings of the 2012 International Workshop on Description Logics, DL-2012, Rome, Italy, June 7-10, 2012. Sun SITE Central Europe (CEUR) (2012)
6. Hladik, J., Model, J.: Tableau systems for $\mathcal{SHIO}$ and $\mathcal{SHIQ}$. In: Haarslev, V., Möller, R. (eds.) Proc. of the 17th Int. Workshop on Description Logics (DL 2004). CEUR Workshop Proceedings, vol. 104, pp. 168–177 (2004)
7. Horrocks, I., Kutz, O., Sattler, U.: The even more irresistible SROIQ. In: Proceedings, Tenth International Conference on Principles of Knowledge Representation and Reasoning, Lake District of the United Kingdom, June 2-5, 2006. pp. 57–67 (2006)
8. Hubauer, T., Legat, C., Seitz, C.: Empowering adaptive manufacturing with interactive diagnostics: A multi-agent approach. In: Advances on Practical Applications of Agents and Multiagent Systems - 9th International Conference on Practical Applications of Agents and Multiagent Systems, PAAMS 2011, Salamanca, Spain, 6-8 April 2011. pp. 47–56 (2011)
9. Karp, R.M.: Reducibility among combinatorial problems. In: Proceedings of a symposium on the Complexity of Computer Computations, held March 20-22, 1972, at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York. pp. 85–103 (1972)
10. Klarman, S., Endriss, U., Schlobach, S.: ABox abduction in the description logic $\mathcal{ALC}$. Journal of Automated Reasoning 46(1), 43–80 (2011)
11. Peirce, C.S.: Deduction, induction, and hypothesis. Popular science monthly 13, 470–482 (1878)
12. Petasis, G., Möller, R., Karkaletsis, V.: BOEMIE: Reasoning-based information extraction. In: Proceedings of the 1st Workshop on Natural Language Processing and Automated Reasoning co-located with 12th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR 2013), A Corunna, Spain, September 15th, 2013. pp. 60–75 (2013)
13. Pukancová, J., Homola, M.: Abductive reasoning with description logics: Use case in medical diagnosis. In: Proceedings of the 28th International Workshop on Description Logics (DL 2015), Athens,Greece, June 7-10, 2015. (2015)
14. Pukancová, J., Homola, M.: Tableau-based abox abduction for description logics: Preliminary report. In: Proceedings of the 29th International Workshop on Description Logics, Cape Town, South Africa, April 22-25, 2016. (2016)
15. Reiter, R.: A theory of diagnosis from first principles. Artif. Intell. 32(1), 57–95 (1987)

16. Sirin, E., Parsia, B., Grau, B.C., Kalyanpur, A., Katz, Y.: Pellet: A practical OWL-DL reasoner. Web Semantics: science, services and agents on the World Wide Web 5(2), 51–53 (2007)
17. Tsarkov, D., Horrocks, I., Patel-Schneider, P.F.: Optimizing terminological reasoning for expressive description logics. J. Autom. Reasoning 39(3), 277–316 (2007)