

# First-Order Expressibility Results for Queries over Inconsistent DL-Lite Knowledge Bases

Meghyn Bienvenu

CNRS & Université Paris Sud, France

meghyn@lri.fr

## 1 Introduction

In recent years, there has been growing interest in ontology-based data access, in which information in the ontology is used to derive additional answers to queries posed over instance data. The *DL-Lite* family of description logics [3, 2]) is considered especially well-suited for such applications due to the fact that query answering can be performed by first incorporating the relevant information from the ontology into the query, and then posing the modified query to the bare data. This property, known as first-order rewritability, means that query answering over *DL-Lite* ontologies has very low data complexity, which is considered key to scalability.

An important problem which arises in ontology-based data access is how to handle inconsistencies. This problem is especially relevant in an information integration setting where the data comes from multiple sources and one generally lacks the ability to modify the data so as to remove the inconsistency. In the database community, the related problem of querying databases which violate integrity constraints has been extensively studied (cf. [4] for a survey) under the name of consistent query answering. The standard approach is based on the notion of a repair, which is a database which satisfies the integrity constraints and is as similar as possible to the original database. Consistent answers are defined as those answers which hold in all repairs. A similar strategy can be used for description logics by replacing the integrity constraints with the ontology.

Consistent query answering for the *DL-Lite* family of description logics was recently studied in [8, 7]. Unfortunately, the obtained complexity results are rather negative: consistent query answering is co-NP-hard in data complexity, even for instance queries and the simplest dialect *DL-Lite<sub>core</sub>*. In the database community, negative results were also encountered, but this spurred a line of research [5, 6, 9] aimed at identifying cases where consistent query answering is feasible, and in particular, can be done using query rewriting. We propose to carry out a similar investigation for *DL-Lite* ontologies, with the aim of better understanding the cases in which query rewriting can be profitably used. In this paper, we make some first steps towards this goal. Specifically, we formulate general conditions which can be used to prove that a consistent rewriting does or does not exist for a given *DL-Lite<sub>core</sub>* TBox and instance query.

The paper is organized as follows. After some preliminaries, we introduce in Sections 3 and 4 the problem of consistent query answering and some useful notions and terminology. Our main results are presented in Sections 4, 5, and 6, where we present general conditions which yield co-NP-hardness, first-order inexpressibility, or first-order expressibility of consistent instance checking in *DL-Lite<sub>core</sub>*. Finally, in Section 7, we

show that query rewriting is always possible if we adopt a previously studied alternative semantics. Note that proofs have been omitted for lack of space but can be found in [1].

## 2 Preliminaries

**Syntax.** *DL-Lite<sub>core</sub>* knowledge bases (KBs) are built up from a set  $N_I$  of constants, called *individuals*, a set  $N_C$  of unary predicates, called *atomic concepts*, and a set  $N_R$  binary predicates, called *atomic roles*. Complex concept and role expressions are constructed using the following syntax:

$$B \rightarrow A \mid \exists R \quad C \rightarrow B \mid \neg B \quad R \rightarrow P \mid P^-$$

where  $A \in N_C$  and  $P \in N_R$ . Here  $B$  (resp.  $R$ ) denotes a *basic concept* (resp. *basic role*), and  $C$  denotes a *general concept*. A *TBox* is a finite set of *inclusions* of the form  $B \sqsubseteq C$  (with  $B, C$  as above). An *ABox* is a finite set of assertions of the form  $B(a)$  ( $B \in N_C$ ) or  $R(a, b)$  ( $R \in N_R$ ) where  $a, b \in N_I$ . A KB consists of a TBox and an ABox.

**Notational conventions** We use  $\text{lhs}(I)$  (resp.  $\text{rhs}(I)$ ) to refer to the basic concept appearing on the left (resp. right) side of an inclusion  $I$ , e.g.  $\text{lhs}(\exists P \sqsubseteq \neg D) = \exists P$  and  $\text{rhs}(\exists P \sqsubseteq \neg D) = D$ . We sometimes use  $R^-$  to mean  $P^-$  if  $R = P \in N_R$  and  $P$  if  $R = P^-$ , and write  $R(a, b)$  to mean  $P(a, b)$  if  $R = P$  and  $R(b, a)$  if  $R = P^-$ .

**Semantics** An *interpretation* is  $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ , where  $\Delta^{\mathcal{I}}$  is a non-empty set and  $\cdot^{\mathcal{I}}$  maps each  $a \in N_I$  to  $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$ , each  $A \in N_C$  to  $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ , and each  $P \in N_R$  to  $P^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ . The function  $\cdot^{\mathcal{I}}$  is straightforwardly extended to general concepts and roles, e.g.  $(\exists S)^{\mathcal{I}} = \{c \mid \exists d(c, d) \in S^{\mathcal{I}}\}$ .  $\mathcal{I}$  satisfies  $G \sqsubseteq H$  if  $G^{\mathcal{I}} \subseteq H^{\mathcal{I}}$ ; it satisfies  $A(a)$  (resp.  $P(a, b)$ ) if  $a^{\mathcal{I}} \in A^{\mathcal{I}}$  (resp.  $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in P^{\mathcal{I}}$ ). We write  $\mathcal{I} \models \alpha$  if  $\mathcal{I}$  satisfies inclusion/assertion  $\alpha$ .  $\mathcal{I}$  is a *model* of  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  if  $\mathcal{I}$  satisfies all inclusions in  $\mathcal{T}$  and assertions in  $\mathcal{A}$ . A KB  $\mathcal{K}$  is *satisfiable/consistent* if it has a model; otherwise it is *unsatisfiable/inconsistent* ( $\mathcal{K} \models \perp$ ). We say that  $\mathcal{K}$  *entails*  $\alpha$ , written  $\mathcal{K} \models \alpha$ , if every model of  $\mathcal{K}$  is a model of  $\alpha$ . The *closure* of  $\mathcal{T}$ , written  $cl(\mathcal{T})$ , consists of all inclusions which are entailed from  $\mathcal{T}$ . Given an ABox  $\mathcal{A}$ , we denote by  $\mathcal{I}_{\mathcal{A}}$  the interpretation which has as its domain the individuals in  $\mathcal{A}$  and which makes true precisely the assertions in  $\mathcal{A}$ .

**Queries** A *query* is a formula of first-order logic with equality (FOL), whose atoms are of the form  $A(t)$ ,  $P(t, t')$ , or  $t = t'$  with  $t, t'$  *terms*, i.e., variables or individuals. *Conjunctive queries* are queries which do not contain  $\forall$ ,  $\neg$ , or  $=$ . *Instance queries* (IQs) are queries consisting of a single atom with no variables (i.e. ABox assertions). A *Boolean query* is a query with no free variables. For a Boolean query  $q$ , we write  $\mathcal{I} \models q$  when  $q$  holds in the interpretation  $\mathcal{I}$ , and  $\mathcal{K} \models q$  when  $\mathcal{I} \models q$  for all models  $\mathcal{I}$  of  $\mathcal{K}$ .

## 3 Consistent query answering

The most commonly used approach to query answering over inconsistent KBs is known as *consistent query answering* and relies on the notion of a repair:

**Definition 1.** A *repair* of a knowledge base  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  is an inclusion-maximal subset  $B$  of  $\mathcal{A}$  consistent with  $\mathcal{T}$ . We use  $\text{Rep}(\mathcal{K})$  to denote the set of repairs of  $\mathcal{K}$ .

Consistent query answering consists in performing standard query answering on each of the repairs and intersecting the answers. For Boolean queries, we have:

**Definition 2.** A Boolean query  $q$  is said to be consistently entailed from  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ , written  $\mathcal{K} \models_{cons} q$ , if  $\langle \mathcal{T}, \mathcal{B} \rangle \models q$  for every repair  $\mathcal{B} \in Rep(\mathcal{K})$ .

Just as with standard query entailment, we can ask whether consistent query entailment can be tested by rewriting the query and evaluating it over the data.

**Definition 3.** A first-order query  $q'$  is a consistent rewriting of a Boolean query  $q$  w.r.t. a TBox  $\mathcal{T}$  if for every ABox  $\mathcal{A}$ , we have  $\mathcal{T}, \mathcal{A} \models_{cons} q$  if and only if  $\mathcal{I}_{\mathcal{A}} \models q'$ .

We illustrate the notion of consistent rewriting on an example.

*Example 1.* Consider the query  $q = R(a, b)$  and the TBox  $\mathcal{T} = \{ \exists R \sqsubseteq \neg D, \exists R \sqsubseteq \neg \exists S^-, \exists R^- \sqsubseteq \neg B, B \sqsubseteq \neg D \}$ . We claim  $q' = R(a, b) \wedge \neg D(a) \wedge \neg \exists x S(x, a) \wedge \neg B(b)$  is a consistent rewriting of  $q$  w.r.t.  $\mathcal{T}$ . To see why, note that if a repair implies  $q$ , then it must contain  $R(a, b)$ . Moreover, if the ABox  $\mathcal{A}$  contains any assertion that contradicts  $R(a, b)$  then we can build a repair which does not contain  $R(a, b)$ . Thus,  $R(a, b)$  is consistently entailed just in the case that  $R(a, b) \in \mathcal{A}$  and there are no assertions in  $\mathcal{A}$  which conflict with  $R(a, b)$ , which is precisely what  $q'$  states.

The method used in Example 1 can be generalized to show that a consistent rewriting exists for all role instance queries<sup>1</sup>. Unfortunately, the same is not true for concept IQs. Indeed, in [7], it was shown that consistent instance checking in  $DL\text{-}Lite_{core}$  is co-NP-hard in data complexity. We present the reduction in the following example.

*Example 2.* Consider an instance  $\varphi = c_1 \wedge \dots \wedge c_m$  of UNSAT, where each  $c_i$  is a propositional clause. Let  $v_1, \dots, v_k$  be the propositional variables appearing in  $\varphi$ . We define the  $DL\text{-}Lite_{core}$  knowledge base  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  as follows:

$$\begin{aligned} \mathcal{T} &= \{ \exists P^- \sqsubseteq \neg \exists N^-, \exists P \sqsubseteq \neg \exists U^-, \exists N \sqsubseteq \neg \exists U^-, \exists U \sqsubseteq A \} \\ \mathcal{A} &= \{ U(a, c_i) \mid 1 \leq i \leq m \} \cup \{ P(c_i, v_j) \mid v_j \in c_i \} \cup \{ N(c_i, v_j) \mid \neg v_j \in c_i \} \end{aligned}$$

It is not hard to verify that  $\varphi$  is unsatisfiable if and only if  $\mathcal{K} \models_{cons} A(a)$ . The basic idea is that because of the inclusion  $\exists P^- \sqsubseteq \neg \exists N^-$ , each repair corresponds to a valuation of the variables, with  $v_j$  assigned true if it has an incoming  $P$ -edge in the repair.

The focus in this paper will be on distinguishing between hard and easy instances of the consistent query answering problem. More specifically, we will be interested in the problem of on deciding for a given TBox and IQ whether a consistent rewriting exists.

## 4 Causes and conflicts

In formulating our results, it will be convenient to introduce some terminology for referring to assertions which participate in the entailment of another assertion or its negation.

<sup>1</sup> Obviously this is no longer the case if we consider a logic with role inclusions like  $DL\text{-}Lite_{\mathcal{R}}$ .

**Definition 4.** Let  $\alpha, \beta$  be assertions and  $\mathcal{T}$  an inclusion. We say  $\beta$  causes (or is a cause of)  $\alpha$  given  $\mathcal{T}$ , written  $\beta \xrightarrow{\mathcal{T}} \alpha$ , if  $\{\mathcal{T}\}, \{\beta\} \models \alpha$ . We say  $\beta$  conflicts with (or is a conflict for)  $\alpha$  given  $\mathcal{T}$ , written  $\beta \bullet \xrightarrow{\mathcal{T}} \alpha$ , if  $\mathcal{T} = B_1 \sqsubseteq \neg B_2$  and  $\beta \models B_1(a)$  and  $\alpha \models B_2(a)$  for some  $a$ . Sometimes we omit  $\mathcal{T}$  if its identity is not relevant.

**Definition 5.** Concepts in the set  $\text{CauseT}(B) = \{D \mid \mathcal{T} \models D \sqsubseteq B\}$  (resp.  $\text{ConflT}(B) = \{D \mid \mathcal{T} \models D \sqsubseteq \neg B\}$ ) are called the cause-types (resp. conflict-types) of  $B$ .

The following straightforward proposition characterizes consistent instance checking in terms of causes and conflicts.

**Proposition 1.** Let  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  be a DL-Lite<sub>core</sub> KB and  $\alpha$  an instance query. Then  $\mathcal{K} \not\models_{\text{cons}} \alpha$  if and only if there exists a subset  $\mathcal{A}' \subseteq \mathcal{A}$  which is consistent with  $\mathcal{T}$  and such that for every  $\beta \in \mathcal{A}$  which causes  $\alpha$ , there is  $\gamma \in \mathcal{A}'$  which conflicts with  $\beta$ .

In other words, consistent instance checking comes down to deciding existence of a consistent subset of the ABox which contradicts all causes of the instance query.

We now introduce the notion of a cause-conflict chain. The intuition is as follows. Suppose that we have an assertion  $\mu_0$  in the ABox which causes the IQ  $\alpha$ . Then to show  $\mathcal{K} \not\models_{\text{cons}} \alpha$ , Proposition 1 says we must select some assertion  $\rho_0$  which conflicts with  $\mu_0$ . But if  $\rho_0$  conflicts with an assertion  $\lambda_1$  which is a conflict of another cause  $\mu_1$ , then this forces us to choose a different conflict  $\rho_1$  for  $\mu_1$  which is consistent with  $\rho_0$ . The presence of  $\rho_1$  may in turn attack a conflict of a third cause  $\mu_3$ , leading us to select a conflict  $\rho_3$  for  $\mu_3$ , and so on.

**Definition 6.** A cause-conflict chain (for TBox  $\mathcal{T}$  and IQ  $\alpha$ ) is a sequence  $\mu_0 \rho_0 \lambda_1 \mu_1 \rho_1 \lambda_2 \mu_2 \rho_2 \dots \lambda_n \mu_n \rho_n \lambda_{n+1} \mu_{n+1}$  of distinct assertions, together with a sequence  $\mathcal{Y}_0 \Gamma_0 \Sigma_1 \Omega_1 \mathcal{Y}_1 \Gamma_1 \Sigma_2 \dots \Omega_n \mathcal{Y}_n \Gamma_n \Sigma_{n+1} \Omega_{n+1} \mathcal{Y}_{n+1}$  of inclusions from  $cl(\mathcal{T})$ , which satisfy:

- for every  $i$ :  $\mu_i \xrightarrow{\mathcal{Y}_i} \alpha$ ,  $\mu_i \bullet \xrightarrow{\Gamma_i} \rho_i$ ,  $\rho_i \bullet \xrightarrow{\Sigma_{i+1}} \lambda_{i+1}$ , and  $\mu_i \bullet \xrightarrow{\Omega_i} \lambda_i$
- if  $j < i$ , then we do not have  $\mu_i \bullet \rightarrow \rho_j$
- $\{\rho_0, \rho_1, \dots, \rho_n\}$  is consistent with  $\mathcal{T}$

Examples of cause-conflict chains can be found in Figure 1a(b) and 2(b). In the following sections, we will consider particular types of cause-conflict chains and see how they are related to the presence of a consistent rewriting.

## 5 General co-NP-hardness result

In this section, we formulate a general condition which can be used to show co-NP-hardness of consistent instance checking. We begin by giving a more elaborate reduction from UNSAT, and then we analyze what is needed to make the proof go through.

*Example 3.* Consider the following TBox  $\mathcal{T}$ :

$$\{ \exists R_0 \sqsubseteq A, \exists R_1 \sqsubseteq A, \exists R_2 \sqsubseteq A, \exists R_3 \sqsubseteq A, \exists R_0 \sqsubseteq \neg \exists S, \exists S^- \sqsubseteq \neg B_1, B_1 \sqsubseteq \neg \exists R_1^-, \exists R_1^- \sqsubseteq \neg D_1, D_1 \sqsubseteq \neg B_2, B_2 \sqsubseteq \neg \exists R_2^-, \exists R_2^- \sqsubseteq \neg D_2, D_2 \sqsubseteq \neg \exists T^-, \exists T^- \sqsubseteq \neg \exists R_3^- \}$$

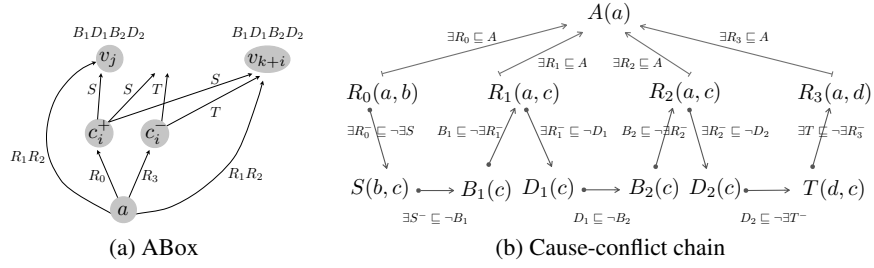


Fig. 1: ABox and type-1 cause-conflict chain for Example 3.

We show via a reduction from UNSAT that deciding whether  $\mathcal{T}, \mathcal{A} \models A(a)$  is co-NP-hard in data complexity. Given a propositional CNF  $\varphi = c_1 \wedge \dots \wedge c_m$  over  $v_1, \dots, v_k$ , we define the ABox  $\mathcal{A}$  as follows (see Figure 1(a) for a pictorial representation):

$$\begin{aligned} & \{ R_0(a, c_i^+), R_3(a, c_i^-) \mid 1 \leq i \leq m \} \cup \{ R_1(a, v_j), R_2(a, v_j) \mid 1 \leq j \leq k+m \} \cup \\ & \{ S(c_i^+, v_j) \mid v_j \in c_i \} \cup \{ T(c_i^-, v_j) \mid \neg v_j \in c_i \} \cup \{ S(c_i^+, v_{k+i}) \mid 1 \leq i \leq m \} \cup \\ & \{ T(c_i^-, v_{k+i}) \mid 1 \leq i \leq m \} \cup \{ B_1(v_j), B_2(v_j), D_1(v_j), D_2(v_j) \mid 1 \leq j \leq k+m \} \end{aligned}$$

We show that  $\varphi \models \perp$  if and only if  $\langle \mathcal{T}, \mathcal{A} \rangle \models_{cons} A(a)$ . For the first direction, suppose we have a satisfying valuation for  $\varphi$ , and let  $V$  be the set of variables which are affected to true. We assume without loss of generality that if a variable  $v_j$  appears only positively (resp. negatively) in  $\varphi$  then  $v_j \in V$  (resp.  $v_j \notin V$ ). Define the subset  $\mathcal{B}$  of  $\mathcal{A}$  as follows:

$$\begin{aligned} & \{ S(c_i^+, v_j) \in \mathcal{A} \mid v_j \in V \} \cup \{ D_1(v_j), D_2(v_j) \mid v_j \in V \} \cup \\ & \{ T(c_i^-, v_j) \in \mathcal{A} \mid v_j \notin V, 1 \leq j \leq k \} \cup \{ B_1(v_j), B_2(v_j) \mid v_j \notin V, 1 \leq j \leq k \} \cup \\ & \{ T(c_i^-, v_{k+i}), B_1(v_{k+i}), B_2(v_{k+i}) \in \mathcal{A} \mid \exists v_j \in V \text{ with } v_j \in c_i \} \cup \\ & \{ S(c_i^+, v_{k+i}), D_1(v_{k+i}), D_2(v_{k+i}) \in \mathcal{A} \mid \forall v_j \in V : v_j \notin c_i \} \end{aligned}$$

It is easy to check that  $\mathcal{B}$  is consistent with  $\mathcal{T}$  and that  $\langle \mathcal{T}, \mathcal{B} \rangle \not\models A(a)$ . It can also be verified that adding any additional assertions from  $\mathcal{A}$  to  $\mathcal{B}$  provokes a contradiction. To understand why, note that either a clause  $c_i$  has some positive variable  $v_j \in V$ , in which case  $S(c_i^+, v_j), T(c_i^-, v_{k+i}) \in \mathcal{B}$ , or it contains no such  $v_j$ , in which case  $S(c_i^+, v_{k+i}), T(c_i^-, v_j) \in \mathcal{B}$ . In either case, both  $R_0(a, c_i^+)$  and  $R_3(a, c_i^-)$  conflict with an assertion in  $\mathcal{B}$ . Thus,  $\mathcal{B}$  is a repair of  $\mathcal{A}$  w.r.t.  $\mathcal{T}$  which does not entail  $A(a)$ .

For the other direction, let  $\mathcal{B}$  be a repair with  $\langle \mathcal{T}, \mathcal{B} \rangle \not\models A(a)$ . It follows that none of the role assertions in  $\mathcal{A}$  involving  $R_0, R_1, R_2, R_3$  appear in  $\mathcal{B}$ . The absence of  $R_1$ - and  $R_2$ -assertions and the consistency of  $\mathcal{B}$  with  $\mathcal{T}$  together imply that for each  $v_j$ , we have either  $B_1$  and  $B_2$  or both  $D_1$  and  $D_2$ . This means each  $v_j$  has either incoming  $S$ -edges or incoming  $T$ -edges, but not both. We create a valuation in which  $v_j$  is affected to true if and only if  $v_j$  has an incoming  $S$ -edge. Clearly if  $c_i$  has a positive literal  $v_j$  which is affected to true, then it will be satisfied by this valuation. If instead all of the positive literals in  $c_i$  are affected to false, then the absence of  $R_0(a, c_i^+)$  can only be explained by the presence in  $\mathcal{B}$  of the assertion  $S(c_i^+, v_{k+i})$ . But this implies in turn the absence

of  $T(c_i^-, v_{k+i})$  in  $\mathcal{B}$ . As  $R_3(a, c_i^-) \notin \mathcal{B}$ , there must be some assertion in  $\mathcal{B}$  of the form  $T(c_i^-, v_\ell)$  ( $1 \leq \ell \leq k$ ). This means  $v_\ell$  will be affected to false our valuation, and hence the clause will be satisfied. Thus, the formula  $\varphi$  is satisfiable.

To understand how the preceding reduction can be generalized, it is helpful to consider the cause-conflict chain pictured in Figure 1(b). This chain contains the essential structure used in the reduction, with individuals  $b$ ,  $c$ , and  $d$  playing the roles of  $c_i^+$ ,  $v_j$ , and  $c_\ell^-$ . We first notice that at the start and end of the chain, there is a switch of individuals, which corresponds to moving from  $c_i^+$  to  $v_j$  and then back to  $c_\ell^-$ . Next remark that in order to show consistency of the constructed  $\mathcal{B}$ , we needed consistency of the sets of “forward” assertions  $\{S(b, c), D_1(c), D_2(c)\}$  and “backward” assertions  $\{B_1(c), B_2(c), T(d, c)\}$ . Also note that in order to use a repair to construct a satisfying valuation, we had to prove that no  $v_j$  had both incoming  $S$ - and  $T$ -edges. This involved showing that the only way to simultaneously contradict all  $R_i$  assertions while retaining consistency was to choose all of the forward ( $D_i$ ) or all of the backward ( $B_i$ ) assertions. Key to this reasoning was the fact that for each  $R_i(a, v_j)$  assertion, we were forced to choose either  $B_i(v_j)$  or  $D_i(v_j)$ . If we could use some  $B_\ell(v_j)$  or  $D_\ell(v_j)$  with  $\ell \neq j$ , the line of reasoning fails. Finally we note that none of the conflicts in the chain involves the query individual  $a$ . This is important because if we used some assertion  $C(a)$  to contradict  $R_i(a, v_j)$ , then we would also contradict  $R_i(a, v_\ell)$  when  $\ell \neq j$ , making it impossible to independently choose truth values for each variable.

The preceding analysis leads us to define the notion of a position (to be able to talk about switching to a new individual) and the notion of type-1 cause-conflict chains.

**Definition 7.** Concepts of the forms  $A$  or  $\exists P$  (resp.  $\exists P^-$ ) are said to have position 1 (resp. 2). An inclusion  $\mathcal{Y}$  begins (resp. concludes) on position  $p$ , written  $\text{bpos}(\mathcal{Y}) = p$  (resp.  $\text{cpos}(\mathcal{Y}) = p$ ), if  $p$  is the position associated with  $\text{lhs}(\mathcal{Y})$  (resp.  $\text{rhs}(\mathcal{Y})$ ).

**Definition 8.** A cause-conflict chain for  $\mathcal{T}$  and  $\alpha$  defined by the sequence of assertions  $\mu_0 \rho_0 \lambda_1 \mu_1 \dots \rho_n \lambda_{n+1} \mu_{n+1}$  and sequence of inclusions  $\mathcal{Y}_0 \Gamma_0 \Sigma_1 \Omega_1 \Upsilon_1 \dots \Sigma_{n+1} \Omega_{n+1} \Upsilon_{n+1}$  is said to be of type-1 if it satisfies the following conditions:

- (C1)  $\text{bpos}(\Upsilon_i) \neq \text{bpos}(\Gamma_i)$  and  $\text{bpos}(\Upsilon_i) \neq \text{cpos}(\Omega_i)$  for all  $i$
- (C2)  $\text{cpos}(\Gamma_0) \neq \text{bpos}(\Sigma_1)$
- (C3)  $\text{cpos}(\Sigma_{n+1}) \neq \text{bpos}(\Omega_{n+1})$
- (C4)  $\{\lambda_1, \dots, \lambda_{n+1}\}$  is consistent with  $\mathcal{T}$
- (C5) if  $j > i$ , then we do not have  $\mu_i \bullet \longrightarrow \lambda_j$

Condition C1 of the definition states that the query individual is not used in the conflicts, whereas C2 and C3 make sure there is a switch to a new individual at the start and end of the chain. Condition C4 guarantees consistency of the “backward” conflict assertions, and C5 ensures that when reading the chain from right to left all causes are relevant (i.e. not already contradicted by one of the previous choices).

*Example 4.* If  $B_1 \sqsubseteq \neg B_2$  were added to the TBox from Example 3, then the chain from Figure 1(b) would not be type-1, since  $B_1(c)$  and  $B_2(c)$  would conflict (violating C4).

The next result shows that the presence of a type-1 cause-conflict chain is sufficient to show co-NP-hardness (and *a fortiori*, the inexistence of a consistent rewriting). The proof generalizes the reduction from Example 3.

**Theorem 1.** *If a type-1 cause-conflict chain for  $\mathcal{T}$  and  $\alpha$  exists, then the problem of deciding whether  $\langle \mathcal{T}, \mathcal{A} \rangle \models_{\text{cons}} \alpha$  is co-NP-hard in data complexity.*

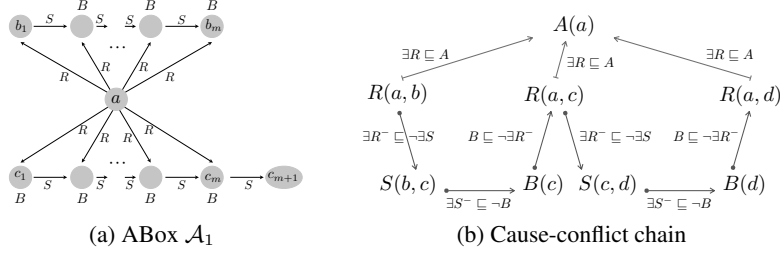


Fig. 2: ABox and Type-2 cause-conflict chain used in Example 5.

## 6 General first-order inexpressibility result

In this section, we show how to use Ehrenfeucht-Fraïssé games rather than complexity arguments, to prove inexistence of a consistent rewriting. As in the previous section, we start with an illustrative example, before formulating the general condition.

*Example 5.* Consider the following  $DL\text{-}Lite_{core}$  TBox  $\mathcal{T}$ :

$$\mathcal{T} = \{ \exists R \sqsubseteq A, \exists R^- \sqsubseteq \neg \exists S, \exists R^- \sqsubseteq \neg B, \exists S^- \sqsubseteq \neg B, \}$$

We show using Ehrenfeucht-Fraïssé games that there is no consistent first-order rewriting of the query  $A(a)$  w.r.t.  $\mathcal{T}$ . Consider some  $k \in \mathbb{N}$ , and let  $m = 2^k + 1$ . We construct two ABoxes  $\mathcal{A}_1$  and  $\mathcal{A}_2$  as follows ( $\mathcal{A}_1$  is pictured in Figure 2(a)):

$$\begin{aligned} \mathcal{A}_1 &= \{ R(a, b_i), R(a, c_i), B(c_i), S(c_i, c_{i+1}) \mid 1 \leq i \leq m \} \cup \\ &\quad \{ B(b_i) \mid 2 \leq i \leq m \} \cup \{ S(b_i, b_{i+1}), \mid 1 \leq i \leq m-1 \} \\ \mathcal{A}_2 &= \mathcal{A}_1 \setminus \{ B(c_1) \} \cup \{ B(b_1) \} \end{aligned}$$

We show that  $\langle \mathcal{T}, \mathcal{A}_1 \rangle \models_{cons} A(a)$  and  $\langle \mathcal{T}, \mathcal{A}_2 \rangle \not\models_{cons} A(a)$ . For the first point, suppose for a contradiction that there is a repair  $\mathcal{B}$  of  $\mathcal{A}_1$  w.r.t.  $\mathcal{T}$  such that  $\langle \mathcal{T}, \mathcal{B} \rangle \not\models A(a)$ . Then there can be no assertions in  $\mathcal{B}$  of the form  $R(a, b_i)$ , and hence each such assertion must provoke a contradiction when added to  $\mathcal{B}$ . In order for  $\mathcal{B} \cup \{ R(a, b_1) \}$  to be inconsistent with  $\mathcal{T}$ , we must have  $S(b_1, b_2) \in \mathcal{B}$ , as  $S(b_1, b_2)$  is the only assertion in  $\mathcal{A}$  which conflicts with  $R(a, b_1)$ . But this means that  $B(b_2) \notin \mathcal{B}$ , and hence that  $S(b_2, b_3) \in \mathcal{B}$ , or else we could add  $R(a, b_2)$  to  $\mathcal{B}$  without provoking a contradiction. Continuing in this manner, we find that  $S(b_{m-1}, b_m) \in \mathcal{B}$ , and so  $B(b_m) \notin \mathcal{B}$ . But in this case,  $\mathcal{B} \cup \{ R(a, b_m) \}$  is consistent with  $\mathcal{T}$ , which contradicts the maximality of  $\mathcal{B}$ . For the second point, we remark that the set  $\mathcal{B} = \{ B(b_i), S(c_i, c_{i+1}) \mid 1 \leq i \leq m \}$  is a repair of  $\mathcal{A}_2$  w.r.t.  $\mathcal{T}$  such that  $\langle \mathcal{T}, \mathcal{B} \rangle \not\models A(a)$ .

We now must show that duplicator has a  $k$ -round winning strategy in the Ehrenfeucht-Fraïssé game based on interpretations  $\mathcal{I}_{\mathcal{A}_1}$  and  $\mathcal{I}_{\mathcal{A}_2}$ . The basic idea is as follows (we defer the full argument to [1]). Whenever spoiler selects a point which is “closer” to the side of  $b_m/c_{m+1}$  in  $\mathcal{I}_{\mathcal{A}_1}$ , duplicator responds with the identical point in  $\mathcal{I}_{\mathcal{A}_2}$ . When spoiler plays “closer” to the  $b_1/c_1$  side, then duplicator plays the  $c_i$  if  $b_i$  was played,

and  $b_i$  if  $c_i$  was played. The important thing is to make sure there is sufficient distance between the indices  $j$  where duplicator copies spoiler and those where he chooses differently. This can be done by keeping track of the leftmost point where the choices differ and the rightmost point where they coincide and ensuring that the distance between these points is always at least  $2^{k-i}$ , where  $i$  is the the current round of play.

Figure 2(b) presents a cause-conflict chain for the preceding example. Most of the conditions we identified in the previous section continue to hold for this chain. The only exception is that we do not have a switch of individuals at the end of the chain. Instead, we can remark that the initial cause-type is repeated further down the chain and can be contradicted in the same way, and this is what we use to create the long chain structure required in the proof. This leads us to define a second class of cause-conflict chains, in which we replace C3 with a new condition which captures this repetition.

**Definition 9.** A cause-conflict chain for  $\mathcal{T}$  and  $\alpha$  whose sequence of inclusions is  $\Upsilon_0 \Gamma_0 \Sigma_1 \Omega_1 \Upsilon_1 \dots \Sigma_{n+1} \Omega_{n+1} \Upsilon_{n+1}$  called type-2 if it satisfies C1, C2, C4, C5, and C6:

$$(C6) \quad \Upsilon_0 = \Upsilon_n \text{ and } \Gamma_0 = \Gamma_n$$

The following theorem states that type-2 cause-conflict chains witness non-existence of a consistent rewriting. The proof generalizes the argument outlined in Example 5.

**Theorem 2.** *If there exists a type-2 cause-conflict chain for  $\mathcal{T}$  and  $\alpha$ , then there is no consistent first-order rewriting for  $\alpha$  w.r.t.  $\mathcal{T}$ .*

We next establish the relationship between type-1 and type-2 chains.

**Theorem 3.** *If there exists a type-1 cause-conflict chain for  $\mathcal{T}$  and  $\alpha$ , then there also exists a type-2 cause-conflict chain. The converse does not hold (assuming  $P \neq NP$ ).*

*Proof (Sketch).* For the first point, the idea to take a second copy of the type-1 chain, reverse it, and append it to the original. For the second point, we show that consistent instance checking for the TBox and IQ from Example 5 can be done in polynomial time by iteratively applying the following rule: if  $R(a, c) \in \mathcal{A}$  and there is no  $S(c, d) \in \mathcal{A}$ , then delete all incoming  $S$ -edges to  $c$ . We continue until either we find  $R(a, c) \in \mathcal{A}$  such that neither  $B(c)$  nor any  $S(c, d)$  belongs to  $\mathcal{A}$  (in which case  $A(a)$  is consistently entailed), or the rule is no longer applicable (and  $A(a)$  is not consistently entailed).

## 7 Rewriting Procedure

In this section, we develop a procedure which is guaranteed to produce a consistent rewriting whenever the TBox  $\mathcal{T}$  and query  $\alpha = A(a)$  satisfy the following two criteria:

**Ordering** There exists a total order  $<$  on  $\text{CauseT}(A)$  such that whenever a cause-conflict chain begins with inclusion  $B_1 \sqsubseteq A$ , ends with inclusion  $B_2 \sqsubseteq A$ , and satisfies conditions C1 and C3, we have  $B_2 < B_1$ .

**No loops** Every cause-conflict chain for  $\mathcal{T}$ ,  $\alpha$  of length  $n+1$  which satisfies  $\text{cpos}(\Sigma_i) = \text{bpos}(\Omega_i)$  for every  $1 \leq i \leq n+1$  is such that  $\Upsilon_i \neq \Upsilon_j$  for all  $i \neq j < n+1$ .



---

**Algorithm 1** Rewrite

---

**Input:** TBox  $\mathcal{T}$ , IQ  $A(a)$     **Output:** a first-order query  $\varphi$   
Initialize  $\varphi$  to  $\perp$  and initialize  $\mathcal{G}$  to the set of all tuples  $(\mathcal{C}, \mathcal{D})$  which satisfy:  
(a)  $\mathcal{C} = \{C \in \text{CauseT}(A) \mid \exists D \in \mathcal{D} \text{ with } D \in \text{ConflT}(C)\}$   
(b) for all  $D \in \mathcal{D}$ , there exists  $C \in \mathcal{C}$  such that  $D \in \text{ConflT}(C)$   
(c) there do not exist  $D_1, D_2 \in \mathcal{D}$  with  $D_2 \in \text{ConflT}(D_1)$   
For every  $(\mathcal{C}, \mathcal{D}) \in \mathcal{G}$  // choose which cause-types to treat globally  
Let  $\mathcal{D} = \{B_1, \dots, B_k, \exists P_1, \dots, \exists P_\ell, \exists P_{\ell+1}^-, \dots, \exists P_m^-\}$  ( $B_i \in \mathbb{N}_C, P_i \in \mathbb{N}_R$ )  
 $S = \{B_i(a)\}_{i=1}^k \cup \{P_i(a, w_i)\}_{i=1}^\ell \cup \{P_i(w_i, a)\}_{i=\ell+1}^m$  // realize concepts in  $\mathcal{D}$  at  $a$   
// compute inequalities needed to ensure consistency (treating variables as individuals)  
 $I = \{v_i \neq v_j \mid v_i, v_j \in \{a, w_1, \dots, w_m\} \text{ and } \mathcal{T}, S \cup \{v_i = v_j\} \models \perp\}$   
 $U = \text{CauseT}(A) \setminus \mathcal{C}$  // cause-types not yet treated  
 $\varphi = \varphi \vee \exists w_1 \dots w_m \bigwedge_{\beta \in S} \beta \wedge \bigwedge_{\gamma \in I} \gamma \wedge \bigwedge_{C \in U} (\forall x \text{auxRewrite}(\mathcal{T}, A(a), C, x, S))$   
Output  $\neg\varphi$ 

---

Our algorithm `Rewrite` creates a big disjunction, where each disjunct corresponds to a choice of a set of cause-types to be conflicted *globally*, i.e. one single assertion involving the query individual is used to conflict all causes of that type. For each disjunct, we first fix the assertions which realize these global conflicts, and then invoke subroutine `auxRewrite` to build one conjunct per untreated cause-type whose purpose is to see whether for each cause of that type there is an assertion which conflicts with it and can safely be added to the repair under construction. These conjuncts have a tree-like structure whose “paths” are cause-conflict chains which satisfy  $\text{cpos}(\Sigma_i) = \text{bpos}(\Omega_i)$  for all  $i$ . Property **No Loops** can thus be applied to show that the recursion depth of `auxRewrite` is no more than  $|\text{CauseT}(A)| + 1$ , ensuring termination. The difficult step in the correctness proof is to show  $\mathcal{I}_{\mathcal{A}} \not\models \text{Rewrite}(\mathcal{T}, q)$  implies  $\langle \mathcal{T}, \mathcal{A} \rangle \not\models_{\text{cons}} q$ . The basic idea is to use the way the negation of the formula is satisfied to direct our construction of a repair which conflicts with every cause of  $q$ . **Ordering** is used to decide in which order we should treat the causes. We illustrate this idea on a concrete example:

*Example 6.* Let  $q = A(a)$  and  $\mathcal{T}$  be the following TBox:

$$\{\exists R_0 \sqsubseteq A, \exists R_1 \sqsubseteq A, \exists R_2 \sqsubseteq A, \exists R_0^- \sqsubseteq \neg \exists S, \exists S^- \sqsubseteq \neg B_1, B_1 \sqsubseteq \neg \exists R_1^-, \\ \exists R_1^- \sqsubseteq \neg D_1, D_1 \sqsubseteq \neg \exists T^-, B_1 \sqsubseteq \neg \exists T^-, \exists T \sqsubseteq \neg \exists R_2^-\}$$

It can be verified that the negation of `Rewrite`( $\mathcal{T}, q$ ) consists of a single disjunct:

$$\forall x R_0(a, x) \rightarrow \exists y (S(x, y) \wedge (R_1(a, y) \rightarrow D_1(y))) \\ \wedge \forall x R_1(a, x) \rightarrow (B_1(x) \vee D_1(x)) \\ \wedge \forall x R_2(a, x) \rightarrow \exists y (T(x, y) \wedge \neg R_1(a, y))$$

We show that if this formula is satisfied in  $\mathcal{I}_{\mathcal{A}}$ , then we can construct a repair  $\mathcal{B}$  of  $\mathcal{A}$  w.r.t.  $\mathcal{T}$  which does not entail  $A(a)$ . First we fix an order on  $\text{CauseT}(A)$  satisfying the conditions in **Ordering**:  $\exists R_0 < \exists R_2 < \exists R_1$ . This means we start by considering causes via  $\exists R_0$ . If  $R_0(a, b) \in \mathcal{A}$ , then the first conjunct allows us to find  $c$  such that  $S(b, c) \in \mathcal{A}$  and  $R_1(a, c) \in \mathcal{A}$  implies  $D_1(c) \in \mathcal{A}$ . We add  $S(b, c)$  to  $\mathcal{B}$ , and also add  $D_1(c)$  if  $R_1(a, c) \in \mathcal{A}$ . Next we move on to the next cause-type in the order,  $\exists R_2$ . If

---

**Algorithm 2** auxRewrite

---

**Input:** TBox  $\mathcal{T}$ , IQ  $A(a)$ ,  $C \in \text{CauseT}(A)$ , variable  $x$ ,  $S$  set of atoms,  $\Theta \subseteq \text{CauseT}(A)$

**Output:** a first-order query  $\chi$

If  $C \in \text{Nc}$ , output  $\neg C(a)$

Set  $\alpha = R(a, x)$ ,  $\chi = \neg\alpha$ , and  $B = \exists R^-$  where  $C = \exists R$  // *R basic role*

For each  $D \in \text{ConflT}(B)$  // *Consider different ways to contradict  $\alpha$  on  $x$*

Set  $\beta = D(x)$  if  $D \in \text{Nc}$  and  $\beta = T(x, y)$  [ $y$  fresh variable] if  $D = \exists T$

If  $\beta$  is necessarily inconsistent with  $S$  given  $\mathcal{T}$ , exit the for-loop

Else, let  $\epsilon$  be the inequalities needed to ensure  $\{\beta\} \cup S$  is consistent with  $\mathcal{T}$

// *Compute untreated causes which are affected by choice of  $\beta$*

Initialize  $\Delta$  to  $\emptyset$

For all  $\exists V \in \text{CauseT}(A)$  such that  $\langle \mathcal{T}, S \cup \{\beta\} \cup \{V(a, x)\} \rangle \not\models \perp$  and

$\text{ConflT}(\exists V^-) \cap \text{ConflT}(D) \neq \emptyset$

Add  $(\exists V, x)$  to  $\Delta$  // *need to find conflict for cause  $V(a, x)$*

If  $D = \exists T$ , then for all  $\exists V \in \text{CauseT}(A)$  with  $\langle \mathcal{T}, S \cup \{\beta\} \cup \{V(a, y)\} \rangle \not\models \perp$

and  $\text{ConflT}(\exists V^-) \cap \text{ConflT}(\exists T^-) \neq \emptyset$

Add  $(\exists V, y)$  to  $\Delta$  // *need to find conflict for cause  $V(a, y)$*

$\chi = \chi \vee (\exists y)(\beta \wedge \epsilon \wedge \bigwedge_{(H,v) \in \Delta} \text{auxRewrite}(\mathcal{T}, A(a), H, v, S \cup \{\beta\}))$

Output  $\chi$

---

we have  $R_2(a, b) \in \mathcal{A}$ , then we use the third conjunct to find  $c$  such that  $T(b, c) \in \mathcal{A}$  and  $R_1(a, c) \notin \mathcal{A}$ , and we add  $T(b, c)$  to  $\mathcal{B}$ . Finally we turn to the final cause-type  $\exists R_1$ , and let  $R_1(a, b) \in \mathcal{A}$ . Possibly we have already added  $D_1(b)$  when dealing with the first conjunct, in which case we do nothing. Otherwise, because of the second conjunct, we have either  $B_1(b) \in \mathcal{A}$  or  $D_1(b) \in \mathcal{A}$ , which we can add to  $\mathcal{B}$ . The set  $\mathcal{B}$  is still consistent with  $\mathcal{T}$  after this step, since if  $T(e, b) \in \mathcal{B}$  then we would have  $R_1(a, b) \notin \mathcal{A}$ , and if  $S(e, b) \in \mathcal{B}$ , then we would have already added a conflict for  $R_1(a, b)$ . We have thus found a set  $\mathcal{B}$  which is consistent with  $\mathcal{T}$  and contradicts every assertion which could cause entailment of  $A(a)$ . By Proposition 1, we have  $\langle \mathcal{T}, \mathcal{A} \rangle \not\models_{\text{cons}} A(a)$ .

**Theorem 4.** *If a TBox  $\mathcal{T}$  and IQ  $q$  satisfy conditions **Ordering** and **No Loops**, then  $\text{Rewrite}(\mathcal{T}, q)$  terminates and outputs a consistent rewriting of  $q$  w.r.t.  $\mathcal{T}$ .*

## 8 Approximating Consistent Query Answering

In order to obtain a more generally applicable positive result, we consider a sound approximation of consistent query answering, which we term *cautious query answering*.

**Definition 10.** *A query  $q$  is cautiously entailed by a knowledge base  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ , written  $\mathcal{K} \models_{\text{caut}} q$ , if  $\langle \mathcal{T}, \bigcap_{\mathcal{B} \in \text{Rep}(\mathcal{K})} \mathcal{B} \rangle \models q$ .*

Note that assertions which belong to all repairs do not participate in any conflict, so we can be quite confident in the answers we obtain from them. For this reason, cautious query answering remains of interest even when consistent query answering is feasible.

In [7], cautious conjunctive query answering (there called Intersection ABox Repair semantics) was shown to be tractable for *DL-Lite<sub>R</sub>* KBs. The proposed algorithm first deletes all assertions involved in some conflict, and then queries the resulting ABox. It was left open whether query rewriting techniques could be used instead. We answer this question in the affirmative and thus obtain an improved upper bound of  $AC_0$ .

**Theorem 5.** *Cautious conjunctive query answering is in  $AC_0$  for  $DL-Lite_{core}$ .*

*Proof (Sketch).* Given a  $DL-Lite_{core}$  TBox  $\mathcal{T}$  a CQ  $q$ , we first compute (in the standard manner) a UCQ  $q' = q_1 \vee \dots \vee q_n$  such that for all ABoxes  $\mathcal{A}$ , we have  $\langle \mathcal{T}, \mathcal{A} \rangle \models q$  if and only if  $\mathcal{I}_{\mathcal{A}} \models q'$ . Then to each disjunct we add the negation of each atomic query which could contradict one of the atoms in the disjunct.

*Example 7.* If  $q = \exists y B(x) \wedge R(x, y)$  and  $\mathcal{T} = \{A \sqsubseteq B, A \sqsubseteq \exists R, B \sqsubseteq \neg D, \exists R^- \sqsubseteq \neg \exists S^-\}$ , standard rewriting yields  $A(x) \vee \exists y B(x) \wedge R(x, y)$ . We then add  $\neg \exists z S(z, y)$  to the second disjunct and  $\neg D(x)$  to both to obtain the cautious rewriting.

Theorem 5 is easily extended to other  $DL-Lite$  logics enjoying FO-rewritability.

## 9 Conclusion and Future Work

In this paper, we took a closer look at the problem of consistent instance checking in  $DL-Lite$  and identified some general conditions which can be used to prove the absence or existence of a consistent rewriting. While our results were formulated for  $DL-Lite_{core}$ , we expect they can be easily lifted to more expressive  $DL-Lite$  dialects like  $DL-Lite_{\mathcal{F}}$  or  $DL-Lite_{\mathcal{R}}$ .

The main objective for future work is to strengthen our results so as to be able to decide for every TBox and instance query whether a consistent rewriting exists. We conjecture that the absence of a type-2 cause-conflict chain is both a necessary and sufficient condition for existence of a consistent rewriting. Extending our investigation to conjunctive queries would be interesting but quite challenging, as it would likely involve confronting longstanding open problems from the database community, where a full characterization of rewritable cases remains elusive [9].

## References

1. <http://www.lri.fr/~meghyn/dl2011-long.pdf>.
2. Alessandro Artale, Diego Calvanese, Roman Kontchakov, and Michael Zakharyashev. The DL-Lite family and relations. *Journal of Artificial Intelligence Research*, 36:1–69, 2009.
3. Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Tractable reasoning and efficient query answering in description logics: The DL-Lite family. *Journal of Automated Reasoning*, 39(3):385–429, 2007.
4. Jan Chomicki. Consistent query answering: Five easy pieces. In *Proc. of ICDT*, pages 1–17, 2007.
5. Ariel Fuxman and Renée J. Miller. First-order query rewriting for inconsistent databases. In *Proc. of ICDT*, pages 337–351, 2005.
6. Luca Grieco, Domenico Lembo, Riccardo Rosati, and Marco Ruzzi. Consistent query answering under key and exclusion dependencies: algorithms and experiments. In *Proc. of CIKM*, pages 792–799, 2005.
7. Domenico Lembo, Maurizio Lenzerini, Riccardo Rosati, Marco Ruzzi, and Domenico Fabio Savo. Inconsistency-tolerant semantics for description logics. In *Proc. of RR*, pages 103–117, 2010.
8. Domenico Lembo and Marco Ruzzi. Consistent query answering over description logic ontologies. In *Proc. of RR*, pages 194–208, 2007.
9. Jef Wijsen. On the first-order expressibility of computing certain answers to conjunctive queries over uncertain databases. In *Proc. of PODS*, pages 179–190, 2010.

## A Proof of Theorem 1

We will use the notation  $\alpha[p]$  to denote the  $p$ -th argument of an assertion  $\alpha$ . The following straightforward lemma will be prove helpful.

**Lemma 1.** *Let  $\alpha, \beta$  be assertions,  $\Upsilon$  be a positive inclusion, and  $\text{bpos}(\Upsilon) = p_1$  and  $\text{cpos}(\Upsilon) = p_2$ . Then  $\alpha \xrightarrow{\Upsilon} \beta$  if and only if  $\alpha[p_1] = \beta[p_2]$ ,  $\alpha$  and  $\text{lhs}(\Upsilon)$  use the same atomic concept / role, and  $\beta$  and  $\text{rhs}(\Upsilon)$  use the same atomic concept / role. Same holds if  $\Upsilon$  is a negative inclusion and we replace  $\alpha \xrightarrow{\Upsilon} \beta$  by  $\alpha \bullet \xrightarrow{\Upsilon} \beta$ .*

**Proof of Theorem 1.** Assume a type-1 cause-conflict chain for  $\mathcal{T}$  and  $\alpha = A(a)$  exists. We first remark that we can assume without loss of generality that  $\text{cpos}(\Gamma_i) = \text{bpos}(\Sigma_{i+1})$  for all  $1 \leq i \leq n$ , since otherwise, we can simply start the chain at  $\mu_i$ , to obtain another (shorter) type-1 cause-conflict chain. Similar reasoning leads us to assume w.l.o.g. that  $\text{cpos}(\Sigma_i) = \text{bpos}(\Omega_i)$  for  $1 \leq i \leq n$ .

We can also assume without loss of generality that if  $j > i$ , then do not have  $\mu_i \bullet \rightarrow \rho_j$ , because otherwise we could connect  $\mu_i$  directly to  $\rho_j$ , and obtain a shorter type-1 chain. Likewise, we can assume that if  $j < i$ , we do not have  $\mu_i \bullet \rightarrow \lambda_j$ , or else we could connect  $\lambda_i$  to  $\mu_i$  to get a shorter type-1 chain.

We know from Lemma 1 that  $\rho_i[\text{bpos}(\Sigma_{i+1})] = \lambda_{i+1}[\text{cpos}(\Sigma_{i+1})]$ ,  $\lambda_i[\text{bpos}(\Omega_i)] = \mu_i[\text{cpos}(\Omega_i)]$ , and  $\mu_i[\text{bpos}(\Gamma_i)] = \rho_i[\text{cpos}(\Gamma_i)]$ . Moreover,  $\text{cpos}(\Sigma_i) = \text{bpos}(\Omega_i)$  from above implies  $\lambda_i[\text{cpos}(\Sigma_i)] = \lambda_i[\text{bpos}(\Sigma_i)]$ , while  $\text{cpos}(\Gamma_i) = \text{bpos}(\Sigma_{i+1})$  implies  $\rho_i[\text{cpos}(\Gamma_i)] = \rho_i[\text{bpos}(\Sigma_{i+1})]$   $i > 0$ . Additionally, because of condition C1, we know that  $\text{bpos}(\Upsilon_i) \neq \text{bpos}(\Gamma_i)$  and  $\text{bpos}(\Upsilon_i) \neq \text{cpos}(\Omega_i)$ , hence  $\mu_i[\text{cpos}(\Omega_i)] = \mu_i[\text{bpos}(\Gamma_i)]$ . Taken altogether, we find that:

$$\begin{aligned} \rho_0[\text{bpos}(\Sigma_1)] &= \lambda_1[\text{cpos}(\Sigma_1)] = \lambda_1[\text{bpos}(\Sigma_1)] = \mu_1[\text{cpos}(\Omega_1)] \\ &= \mu_1[\text{bpos}(\Gamma_1)] = \rho_1[\text{cpos}(\Gamma_1)] = \dots = \rho_n[\text{bpos}(\Sigma_{n+1})] = \lambda_{n+1}[\text{cpos}(\Sigma_{n+1})] \end{aligned} \quad (*)$$

We use this fact to show that (given our assumptions)  $\Upsilon_i \neq \Upsilon_j$  whenever  $1 \leq i, j \leq n$ . To see why, suppose  $\Upsilon_i = \Upsilon_j$  for  $1 \leq i < j \leq n$ , and let  $p = \text{bpos}(\Upsilon_i) = \text{bpos}(\Upsilon_j)$  and  $q = \text{cpos}(\Upsilon_i) = \text{cpos}(\Upsilon_j)$ . Then by Lemma 1,  $\mu_i[p] = \alpha[q] = \mu_j[p]$  and  $\mu_i$  and  $\mu_j$  involve the same role. Moreover, from above, we have  $\mu_i[\text{cpos}(\Omega_i)] = \mu_j[\text{cpos}(\Omega_j)]$ . As we know  $\text{bpos}(\mu_i) \neq \text{cpos}(\Omega_i)$  and  $\text{bpos}(\mu_i) \neq \text{cpos}(\Omega_i)$  (by C1), it follows that  $\mu_i = \mu_j$ , which contradicts the fact that cause-conflict chains must not have two occurrences of an assertion.

We next remark that condition C1 implies that every  $\mu_i$  is a role assertion, and thus for every  $i$  we can find a basic role  $R_i$  such that  $\Upsilon_i = \exists R_i \sqsubseteq A$ . Conditions C1 and C2 together imply that there is a basic role  $S$  such that  $\Gamma_0 = \exists R^- \sqsubseteq \neg \exists S$ . Then because of C2, we can find a basic concept  $B_1$  such that  $\Sigma_1 = \exists S^- \sqsubseteq \neg B_1$ . At the end of the chain, because of C1 and C3, there must be a basic role  $T$  and basic concept  $D_n$  such that  $\Omega_{n+1} = \exists T \sqsubseteq \neg \exists R_{n+1}^-$  and  $\Sigma_{n+1} = D_n \sqsubseteq \neg \exists T^-$ . We further note that because of Lemma 1,  $\text{cpos}(\Sigma_i) = \text{bpos}(\Omega_i)$ , and  $\text{cpos}(\Gamma_i) = \text{bpos}(\Sigma_{i+1})$  for all  $1 \leq i \leq n$ , we can find basic concepts  $B_i$  ( $2 \leq i \leq n$ ) and  $D_i$  ( $1 \leq i \leq n-1$ ) such that  $\Sigma_i = D_i \sqsubseteq \neg B_{i+1}$  ( $2 \leq i \leq n$ ),  $\Omega_i = B_i \sqsubseteq \neg \exists R_i^-$  ( $1 \leq i \leq n$ ), and  $\Gamma_i = \exists R_i^- \sqsubseteq \neg D_i$  ( $1 \leq i \leq n$ ). We also have  $\Omega_1 = B_1 \sqsubseteq \neg \exists R_1^-$  and

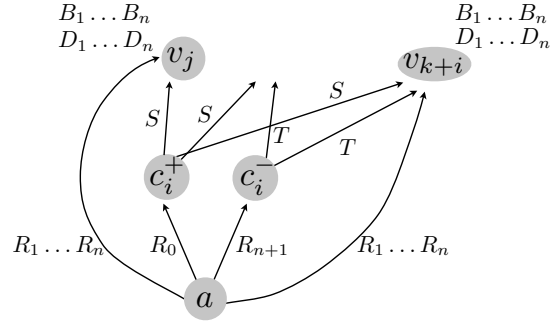


Fig. 3: ABox for SAT reduction in proof of Theorem 1.

$$\Gamma_n = \exists R_n^- \sqsubseteq \neg D_n.$$

Now that we have a better grasp on the types of inclusions in  $\mathcal{T}$  (or rather  $cl(\mathcal{T})$ ), we are ready to give our SAT reduction. Consider a propositional CNF  $\varphi = c_1 \wedge \dots \wedge c_m$  over variables  $v_1, \dots, v_k$ . Our ABox will use the following individuals:  $a, v_j$  ( $1 \leq j \leq k + m$ ),  $c_i^+, c_i^-$  ( $1 \leq i \leq m$ ),  $b_\ell$  ( $B_\ell \notin \mathbf{N}_C$ ), and  $d_\ell$  ( $D_\ell \notin \mathbf{N}_C$ ). Individuals  $v_1, \dots, v_k$  represent the variables of  $\varphi$ , while  $v_{k+1}, \dots, v_{k+m}$  are extra “dummy” variables, whose purpose will be made clear later. We use individuals  $c_i^+, c_i^-$  for the positive and negative parts of clause  $c_i$ . The variables  $b_\ell$  (resp.  $d_\ell$ ) will be used as witnesses for existential concepts  $B_\ell$  (resp.  $D_\ell$ ). The ABox  $\mathcal{A}$  is defined as follows (see Figure 3 for a simplified pictorial representation):

$$\begin{aligned} \mathcal{A} = & \{ R_0(a, c_i^+), R_{n+1}(a, c_i^-) \mid 1 \leq i \leq m \} \cup \\ & \{ R_i(a, v_j) \mid 1 \leq i \leq n, 1 \leq j \leq k + m \} \cup \\ & \{ S(c_i^+, v_j) \mid v_j \text{ appears positively in } c_i \} \cup \\ & \{ T(c_i^-, v_j) \mid v_j \text{ appears negatively in } c_i \} \cup \\ & \{ S(c_i^+, v_{k+i}), T(c_i^-, v_{k+i}) \mid 1 \leq i \leq m \} \cup \\ & \{ B_\ell(v_j) \mid B_\ell \in \mathbf{N}_C, 1 \leq j \leq k + m \} \cup \\ & \{ D_\ell(v_j) \mid D_\ell \in \mathbf{N}_C, 1 \leq j \leq k + m \} \cup \\ & \{ U_\ell(v_j, b_\ell) \mid B_\ell = \exists U_\ell, 1 \leq j \leq k + m \} \cup \\ & \{ U_\ell(v_j, d_\ell) \mid D_\ell = \exists U_\ell, 1 \leq j \leq k + m \} \end{aligned}$$

In what follows, we show that  $\varphi$  is satisfiable if and only if  $A(a)$  is not consistently entailed from  $\langle \mathcal{T}, \mathcal{A} \rangle$ . For the first direction, suppose that  $\varphi$  is satisfiable. Pick a satisfying valuation for  $\varphi$ , and let  $V$  be the set of variables affected to true by this valuation. We assume w.l.o.g. that if a variable appears only positively in  $\varphi$  then it is affected to true, and if it appears only negatively then it is assigned false. Consider the subset  $\mathcal{B}$  of  $\mathcal{A}$  defined as follows:

$$\mathcal{B} = \{ S(c_i^+, v_j) \in \mathcal{A} \mid v_j \in V \} \cup$$

$$\begin{aligned}
& \{D_\ell(v_j) \in \mathcal{A} \mid v_j \in V\} \cup \\
& \{U_\ell(v_j, d_\ell) \in \mathcal{A} \mid v_j \in V\} \cup \\
& \{T(c_i^-, v_j) \in \mathcal{A} \mid v_j \notin V, 1 \leq j \leq k\} \cup \\
& \{B_\ell(v_j) \in \mathcal{A} \mid v_j \notin V, 1 \leq j \leq k\} \cup \\
& \{U_\ell(v_j, b_\ell) \in \mathcal{A} \mid v_j \notin V, 1 \leq j \leq k\} \cup \\
& \{S(c_i^+, v_{k+i}) \in \mathcal{A} \mid \text{no } v_j \in V \text{ appear positively in } c_i\} \cup \\
& \{D_\ell(v_{k+i}) \in \mathcal{A} \mid \text{no } v_j \in V \text{ appear positively in } c_i\} \cup \\
& \{U_\ell(v_{k+i}, d_\ell) \in \mathcal{A} \mid \text{no } v_j \in V \text{ appear positively in } c_i\} \cup \\
& \{T(c_i^-, v_{k+i}) \in \mathcal{A} \mid \text{some } v_j \in V \text{ appear positively in } c_i\} \cup \\
& \{B_\ell(v_{k+i}) \in \mathcal{A} \mid \text{some } v_j \in V \text{ appear positively in } c_i\} \cup \\
& \{U_\ell(v_{k+i}, b_\ell) \in \mathcal{A} \mid \text{some } v_j \in V \text{ appear positively in } c_i\}
\end{aligned}$$

Clearly  $A(a)$  cannot be entailed by  $\langle \mathcal{T}, \mathcal{A} \rangle$ . We need to show that  $\mathcal{B}$  is a repair of  $\mathcal{A}$  w.r.t.  $\mathcal{T}$ . We begin by proving that  $\mathcal{B}$  is consistent with  $\mathcal{T}$ . By Proposition ??, if  $\mathcal{B}$  were inconsistent with  $\mathcal{T}$ , we could find an individual  $e$  and an inclusion  $G_1 \sqsubseteq \neg G_2 \in cl(\mathcal{T})$  such that  $\langle \mathcal{T}, \mathcal{B} \rangle \models \{G_1(e), G_2(e)\}$ . Thus, it is sufficient to show that there is no such individual in  $\mathcal{B}$ . For individuals of the form  $c_i^+$  (resp.  $c_i^-$ ), the only concept satisfied is  $\exists S$  (resp.  $\exists T$ ), which is satisfiable w.r.t.  $\mathcal{T}$  since  $\rho_0 \models \exists S(\rho_0[\text{cpos}(\Gamma_0)])$  (resp.  $\lambda_{n+1} \models \exists T(\lambda_{n+1}[\text{bpos}(\Omega_{n+1})])$ ). Similarly, individuals of the form  $b_\ell$  or  $d_\ell$  only satisfy a single concept whose consistency w.r.t.  $\mathcal{T}$  is witnessed by one of the assertions  $\rho_i$  or  $\lambda_i$ . The only interesting individuals are those of the form  $v_j$ , for which there are two cases. The first possibility is that  $v_j$  satisfies  $\exists S^-, D_1, \dots, D_n$ . In this case, we use (\*) which tells us

$$\rho_0[\text{bpos}(\Sigma_1)] = \rho_1[\text{bpos}(\Gamma_1)] = \dots = \rho_n[\text{bpos}(\Sigma_{n+1})]$$

and hence that  $\{\rho_0, \dots, \rho_n\} \models \{\exists S^-(e), D_1(e), \dots, D_n(e)\}$  for  $e = \rho_0[\text{bpos}(\Sigma_1)]$ . Since we know  $\{\rho_0, \dots, \rho_n\}$  is consistent with  $\mathcal{T}$ , it follows that the assertions concerning  $v_j$  do not provoke a contradiction. We can proceed in a similar manner for the second possibility, which is that  $v_j$  satisfies  $\exists T^-, B_1, \dots, B_n$ . The only difference is that instead of the definition of cause-conflict chains, we use condition C4, which guarantees satisfiability of  $\{\lambda_1, \dots, \lambda_{n+1}\}$  with  $\mathcal{T}$ .

It remains to be shown that  $\mathcal{B}$  is a maximal subset of  $\mathcal{A}$  consistent with  $\mathcal{T}$ . Actually, it is sufficient to show that none of the assertions concerning  $a$  can be added consistently to  $\mathcal{B}$ , since the remaining assertions cannot cause  $A(a)$ . We first note that by construction, for each  $v_j$  we have either  $\langle \mathcal{T}, \mathcal{B} \rangle \models \{B_1(v_j), \dots, B_n(v_j)\}$  or  $\langle \mathcal{T}, \mathcal{B} \rangle \models \{D_1(v_j), \dots, D_n(v_j)\}$ . It follows that we cannot add any assertion of the form  $R_i(a, v_j)$  without introducing a contradiction (via one of the inclusions  $\Gamma_i$  or  $\Omega_i$  in  $cl(\mathcal{T})$ ). Next consider some pair of assertions  $R_0(a, c_i^+)$  and  $R_{n+1}(a, c_i^-)$ . As  $V$  defines a satisfying valuation for  $\varphi$ , we know that either there is some  $v_j \in V$  which appears positively in  $c_i$  or some  $v_j \notin V$  which appears negatively. In the former case, the assertion  $S(c_i^+, v_j)$  will be present in  $\mathcal{B}$ , so adding  $R_0(a, c_i^+)$  will contradict  $\Gamma_0$ . Moreover, by definition,  $T(c_i^-, v_{k+i}) \in \mathcal{B}$ , so adding  $R_{n+1}(a, c_i^-)$  to  $\mathcal{B}$  would violate  $\Omega_{n+1}$ . Suppose then that there is no positive literal of  $c_i$  which is satisfied, and let  $v_j$

be a negative literal with  $v_j \notin V$ . Then we must have both  $T(c_i^-, v_j)$  and  $S(c_i^+, v_{k+i})$ , and so neither  $R_0(a, c_i^+)$  nor  $R_{n+1}(a, c_i^-)$  can be consistently added to  $\mathcal{B}$ .

For the second direction, suppose that  $A(a)$  is not consistently entailed by  $\langle \mathcal{T}, \mathcal{A} \rangle$ , and let  $\mathcal{B}$  be a repair of  $\mathcal{A}$  w.r.t.  $\mathcal{T}$  with  $\langle \mathcal{T}, \mathcal{B} \rangle \not\models A(a)$ . Then there must be no assertion of the form  $R_\ell(a, v_j)$ ,  $R_0(a, c_i^+)$ , or  $R_{n+1}(a, c_i^-)$  in  $\mathcal{B}$ . The absence of  $R_\ell(a, v_j)$  implies that there is some assertion in  $\mathcal{B}$  which conflicts with  $R_\ell(a, v_j)$  given  $\mathcal{T}$ . Moreover, we know that this assertion must involve  $v_j$  but not  $a$ . We note that  $R_\ell(a, v_j)$  cannot conflict with  $B_p(v_j)$  or  $U_p(v_j, b_p)$  for  $p \neq \ell$  since this would mean that  $B_p \sqsubseteq \neg \exists R_\ell^-$ , and hence  $\lambda_p \bullet \rightarrow \lambda_\ell$  for  $p \neq \ell$ . This cannot be since condition (C4) of type-1 chains tells us that  $\lambda_p \bullet \rightarrow \lambda_\ell$  does not hold for  $p > \ell$ , and the same was assumed above for  $p < \ell$ . Likewise,  $R_\ell(a, v_j)$  cannot conflict with  $D_p(v_j)$  or  $U_p(v_j, d_p)$  for  $p \neq \ell$ , since this would contradict either item (iii) of the definition of cause-conflict chains or our earlier assumption. It thus follows that the conflict must result from an assertion in  $\mathcal{B}$  having one of the following forms:  $B_\ell(v_j)$ ,  $U_\ell(v_j, b_\ell)$ ,  $D_\ell(v_j)$ , or  $U_\ell(v_j, d_\ell)$ . In fact, only one such assertion can belong to  $\mathcal{B}$ , since  $\mathcal{A}$  only contains two assertions from this list, and they conflict with each other. Since we must have one conflicting assertion for each  $R_\ell(a, v_j)$  with  $1 \leq \ell \leq n$ , and we know that  $\mathcal{B}$  is consistent with  $\mathcal{T}$ , and that  $\mathcal{T} \models D_p \sqsubseteq \neg B_{p+1}$  for all  $1 \leq p \leq n$ , it follows that one of the following holds for every  $1 \leq j \leq k + m$ :

- for  $1 \leq p \leq n$ :  $\mathcal{B}$  contains  $B_p(v_j)$ , if  $B_p \in \mathbf{N}_C$ , or  $U_p(v_j, b_j)$ , if  $B_p = \exists U_p$
- for  $1 \leq p \leq n$ :  $\mathcal{B}$  contains  $D_p(v_j)$ , if  $D_p \in \mathbf{N}_C$ , or  $U_p(v_j, b_p)$ , if  $D_p = \exists U_p$

In other words, for each  $v_j$ , the set  $\mathcal{B}$  must contain either all of the “ $B_p$  assertions” or all of the “ $D_p$  assertions”. Note that in the former case,  $\mathcal{B}$  will contain all assertions from  $\mathcal{A}$  of the form  $S(c_i^+, v_j)$  (consistency is guaranteed by item (ii) of cause-conflict chains), but it cannot contain any assertions of the form  $T(c_i^-, v_j)$  since they would cause inconsistency. In the latter case, just the opposite is true:  $\mathcal{B}$  will contain all assertions from  $\mathcal{A}$  of the form  $T(c_i^-, v_j)$  (here we use condition C4) and none of the form  $S(c_i^+, v_j)$ . We can thus define a valuation  $V \subseteq \{v_1, \dots, v_k\}$  as follows:  $v_j \in V$  if and only if  $\mathcal{B}$  contains an assertion of the form  $S(c_i^+, v_j)$ . We aim to show that this valuation satisfies  $\varphi$ . Consider some clause  $c_i$ . The absence of  $R_0(a, c_i^+)$  in  $\mathcal{B}$  tells us that there is some assertion in  $\mathcal{B}$  which conflicts with it. The only possibility is an assertion of the form  $S(c_i^+, v_j)$ . Likewise, the absence of  $R_{n+1}(a, c_i^-)$  implies the existence of some  $T(c_i^-, v_j)$  in  $\mathcal{B}$ . Note however that we know that  $v_{k+i}$  can have either incoming  $S$ -edges or incoming  $T$ -edges, but not both. Thus,  $\mathcal{B}$  must contain an assertion  $S(c_i^+, v_j)$  or  $T(c_i^-, v_j)$  with  $1 \leq j \leq k$ . In the former case,  $v_j$  has an incoming  $S$ -edge, and hence  $v_j \in V$ , so  $V$  satisfies  $c_i$ . In the latter case,  $v_j$  has an incoming  $T$ -edge, and hence no incoming  $S$ -edge, which means that  $v_j \notin V$ , and so  $V$  satisfies  $c_i$ . Thus, the valuation  $V$  satisfies all clauses in  $\varphi$ , i.e.  $\varphi$  is satisfiable.  $\square$

## B Proof of Theorem 2

Assume a type-2 cause-conflict chain for  $\mathcal{T}$  and  $\alpha = A(a)$  exists. We can assume without loss of generality that

$$\text{cpos}(\Sigma_i) = \text{bpos}(\Omega_i) \text{ for all } 1 \leq i \leq n + 1$$

since otherwise the sub-chain ending at  $\mu_i$  would be a type-1 cause-conflict chain, giving us co-NP-hardness (Theorem 1). Finally, for the same reasons as in the proof of Theorem 1, we can assume that our type-2 cause-conflict chain has the following properties:

- if  $j > i$ , then we do not have  $\mu_i \bullet \longrightarrow \rho_j$
- if  $j < i$ , then we do not have  $\mu_i \bullet \longrightarrow \lambda_j$

For type-1 chains, we could restrict our attention to chains satisfying  $\text{cpos}(\Gamma_i) = \text{bpos}(\Sigma_{i+1})$  for all  $1 \leq i \leq n$ , but we cannot make this assumption for type-2 chains. It will thus be necessary for us to distinguish the positions where this property fails. We will use  $h_0 < \dots < h_y$  to refer to the positions  $i \in \{0, 1, \dots, n\}$  such that  $\text{cpos}(\Gamma_i) \neq \text{bpos}(\Sigma_{i+1})$ . We denote by SP (for *switch points*) the set  $\{h_0, \dots, h_y\}$ . Note that because of earlier assumptions, we know that  $h_0 = 0$  and  $h_y = n$ . Also note that within each “segment”, all conflicts involve the same individual, and no inclusion  $\mathcal{Y}_i$  is repeated. To make this more formal, consider some  $0 \leq p < y$ . Using Lemma 1, our assumption that  $\text{cpos}(\Sigma_i) = \text{bpos}(\Omega_i)$  for all  $1 \leq i \leq n+1$ , condition C1 (which gives  $\mu_i[\text{cpos}(\Omega_i)] = \mu_i[\text{bpos}(\Gamma_i)]$  for all  $1 \leq i \leq n$ ) and the fact that  $\text{cpos}(\Gamma_i) = \text{bpos}(\Sigma_{i+1})$  for all  $h_p < i < h_{p+1}$ , we obtain:

$$\begin{aligned} \rho_{h_p}[\text{bpos}(\Sigma_{h_{p+1}})] &= \lambda_{h_{p+1}}[\text{cpos}(\Sigma_{h_{p+1}})] = \lambda_{h_{p+1}}[\text{bpos}(\Omega_{h_{p+1}})] \\ &= \mu_{h_{p+1}}[\text{cpos}(\Omega_{h_{p+1}})] = \mu_{h_{p+1}}[\text{bpos}(\Gamma_{h_{p+1}})] = \rho_{h_{p+1}}[\text{cpos}(\Gamma_{h_{p+1}})] \quad (**) \\ &= \dots = \mu_{h_{p+1}}[\text{cpos}(\Omega_{h_{p+1}})] = \mu_{h_{p+1}}[\text{bpos}(\Gamma_{h_{p+1}})] = \rho_{h_{p+1}}[\text{cpos}(\Gamma_{h_{p+1}})] \end{aligned}$$

Also note that for all  $h_p < i < j \leq h_{p+1}$ , we have  $\mathcal{Y}_i \neq \mathcal{Y}_j$ , since otherwise (\*\*) would imply that  $\mu_i = \mu_j$ , contradicting one of the conditions of cause-conflict chains.

As we noted in the proof of Theorem 1, condition C1 implies that every  $\mu_i$  is a role assertion, and thus for every  $i$  we can find a basic role  $R_i$  such that  $\mathcal{Y}_i = \exists R_i \sqsubseteq A$ . Because of condition C6, we know that  $R_0 = R_n$ . Conditions C1 and C2 together imply that there is a basic role  $T_0$  such that  $\Gamma_0 = \Gamma_n = \exists R_0^- \sqsubseteq \neg \exists T_0$ . C2 also tells us that we can find a basic concept  $B_1$  such that  $\Sigma_1 = \Sigma_{n+1} = \exists T_0^- \sqsubseteq \neg B_1$  and  $\Omega_1 = \Omega_{n+1} = B_1 \sqsubseteq \neg \exists R_1^-$ . For each  $i \in \text{SP} \setminus \{0, n\}$ , the fact that  $\text{cpos}(\Gamma_i) \neq \text{bpos}(\Sigma_i)$  means that we can find a basic role  $T_i$  and basic concept  $B_{i+1}$  such that  $\Gamma_i = \exists R_i^- \sqsubseteq \neg \exists T_i$ ,  $\Sigma_{i+1} = \exists T_i^- \sqsubseteq \neg B_{i+1}$ , and  $\Omega_{i+1} = B_{i+1} \sqsubseteq \neg \exists R_{i+1}^-$ . Finally, for every  $i \notin \text{SP}$  with  $1 \leq i \leq n-1$ , we can find basic concepts  $D_i$  and  $B_{i+1}$  such that:  $\Gamma_i = \exists R_i^- \sqsubseteq \neg D_i$ ,  $\Sigma_{i+1} = D_i \sqsubseteq \neg B_{i+1}$ ,  $\Omega_{i+1} = B_{i+1} \sqsubseteq \neg \exists R_{i+1}^-$ .

We show using Ehrenfeucht-Fraïssé games that there is no consistent FOL rewriting of the query  $A(a)$  w.r.t.  $\mathcal{T}$ . Consider some  $k \in \mathbb{N}$ , and let  $m = 2^k + 2$ . Our objective is to find ABoxes  $\mathcal{A}_1$  and  $\mathcal{A}_2$  which give different answers to  $q$  but cannot be distinguished by a  $k$ -round Ehrenfeucht-Fraïssé game. The ABoxes we construct will use the following individuals:  $a, v, b_i^p$  ( $1 \leq i \leq m, 0 \leq p < y$ ),  $c_i^p$  ( $1 \leq i \leq m, 0 \leq p < y$ ),  $w_\ell^B$  ( $B_\ell \notin \text{NC}$ ), and  $w_\ell^D$  ( $D_\ell \notin \text{NC}$ ). Individuals  $b_i^0, \dots, b_i^{y-1}$  (resp.  $c_i^0, \dots, c_i^{y-1}$ ) play the same role as  $b_i$  (resp.  $c_i$ ) in Example 5. The individuals  $w_\ell^B$  (resp.  $w_\ell^D$ ) are used as witnesses for existential concepts  $B_\ell$  (resp.  $D_\ell$ ), while the individual  $v$  is used as a witness for  $\exists T_0$ .



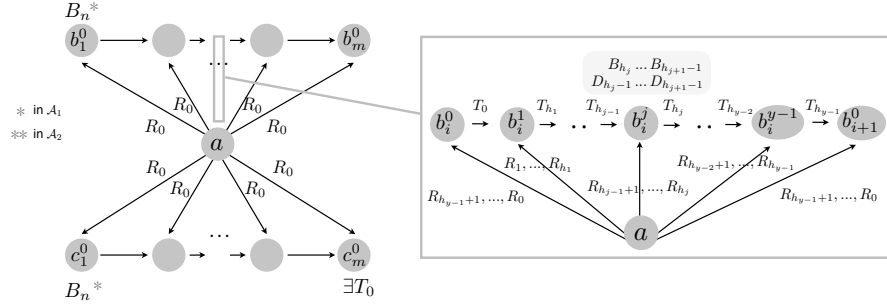


Fig. 4: Visual aid for the ABox used in the proof of Theorem 2.

The ABoxes  $\mathcal{A}_1$  and  $\mathcal{A}_2$  are based upon the following ABox  $\mathcal{A}$ :

$$\begin{aligned}
\mathcal{A} = & \{ R_j(a, b_i^p), R_j(a, c_i^p) \mid 1 \leq i \leq m-1, 1 \leq p \leq y-1, h_{p-1} < j \leq h_p \} \cup \\
& \{ R_j(a, b_i^0), R_j(a, c_i^0) \mid 2 \leq i \leq m \text{ and } j = 0 \text{ or } h_{y-1} < j < h_y \} \cup \\
& \{ R_0(a, b_1^0), R_0(a, c_1^0) \} \cup \\
& \{ T_{h_p}(b_i^p, b_i^{p+1}), T_{h_p}(c_i^p, c_i^{p+1}) \mid 1 \leq i \leq m-1, 0 \leq p \leq y-2 \} \cup \\
& \{ T_{h_p}(b_i^p, b_{i+1}^0), T_{h_p}(c_i^p, c_{i+1}^0) \mid 1 \leq i \leq m-1, p = y-1 \} \cup \\
& \{ D_j(b_i^p), D_j(c_i^p) \mid D_j \in \mathbf{N}_C \text{ and } 1 \leq i \leq m-1, 1 \leq p \leq y-1, h_{p-1} < j < h_p \\
& \quad \text{or } 2 \leq i \leq m, h_{y-1} < j < h_y \} \cup \\
& \{ S_j(b_i^p, w_j^D), S_j(c_i^p, w_j^D) \mid D_j = \exists S_j \text{ and } 1 \leq i \leq m-1, 1 \leq p \leq y-1, \\
& \quad h_{p-1} < j \leq h_p \text{ or } 2 \leq i \leq m \text{ and } h_{y-1} < j \leq h_y \} \cup \\
& \{ B_j(b_i^p), B_j(c_i^p) \mid B_j \in \mathbf{N}_C \text{ and } 1 \leq i \leq m-1, 1 \leq p \leq y-1, h_{p-1} < j \leq h_p \\
& \quad \text{or } 2 \leq i \leq m \text{ and } j = 0 \text{ or } h_{y-1} < j \leq h_y \} \cup \\
& \{ S_j(b_i^p, w_j^B), S_j(c_i^p, w_j^B) \mid B_j = \exists S_j, \text{ and } 1 \leq i \leq m-1, 1 \leq p \leq y-1, \\
& \quad h_{p-1} < j \leq h_p \text{ or } 2 \leq i \leq m \text{ and } h_{y-1} < j \leq h_y \}
\end{aligned}$$

The extension of  $\mathcal{A}$  to  $\mathcal{A}_1$  and  $\mathcal{A}_2$  depends on whether  $B_n$  is a concept name or existential concept. If  $B_n \in \mathbf{N}_C$ , then we have:

$$\mathcal{A}_1 = \mathcal{A} \cup \{ B_n(c_1^0), T_0(c_m^0, v) \} \quad \mathcal{A}_2 = \mathcal{A} \cup \{ B_n(b_1^0), T_0(c_m^0, v) \}$$

Otherwise, if  $B_n = \exists S_n$ , then we use a slightly different formulation:

$$\mathcal{A}_1 = \mathcal{A} \cup \{ S_n(c_1^0, w_n^B), T_0(c_m^0, v) \} \quad \mathcal{A}_2 = \mathcal{A} \cup \{ S_n(b_1^0, w_n^B), T_0(c_m^0, v) \}$$

Figure 4 provides a pictorial representation of  $\mathcal{A}_1$  and  $\mathcal{A}_2$ .

We first show that the Duplicator has a  $k$ -round winning strategy in the Ehrenfeucht-Fraïssé game based on interpretations  $\mathcal{I}_{\mathcal{A}_1}$  and  $\mathcal{I}_{\mathcal{A}_2}$ . More specifically, we show that Duplicator can play in such a way as to satisfy the following property after each round  $g$ .

Let  $\mathbf{d} = (d_{-2}, d_{-1}, d_0, d_1, d_2, \dots, d_g)$  be the vector composed of  $a, c_1^0, c_m^0$  followed by the sequence of  $g$  points played so far in  $\mathcal{I}_{\mathcal{A}_1}$ , and let  $\mathbf{e} = (e_{-2}, e_{-1}, e_0, e_1, e_2, \dots, e_g)$  be the vector starting by  $a, b_1^0, c_m^0$  and finishing with the  $g$  points played so far in  $\mathcal{I}_{\mathcal{A}_2}$ . Then we have the following:

1. if  $d_\ell \in \{b_i^p, c_i^p\}$ , then  $e_\ell \in \{b_i^p, c_i^p\}$
2. letting  $\lambda = \max(\{i \mid d_\ell \in \{b_i^p, c_i^p\}, d_\ell \neq e_\ell\})$  and  $\rho = \min(\{i \mid d_\ell \in \{b_i^p, c_i^p\}, d_\ell = e_\ell\})$ , we have  $\rho - \lambda > 2^{k-g}$
3.  $(\mathbf{d}, \mathbf{e})$  define a partial isomorphism between  $\mathcal{I}_{\mathcal{A}_1}$  and  $\mathcal{I}_{\mathcal{A}_2}$

We will proceed by induction on  $g$ . The base case is when  $g = 0$ , i.e. the game has not yet begun. In this case,  $\mathbf{d} = (a, c_1^0, c_m^0)$  and  $\mathbf{e} = (a, b_1^0, c_m^0)$ . The first and third conditions clearly hold, and for the second, we remark that  $\lambda = 1$  and  $\rho = m$ , hence  $\rho - \lambda = m - 1 = 2^k + 1$ . For the induction step, assume that Duplicator can play so as to satisfy the property for the first  $g$  rounds, and consider what happens in round  $g + 1$ . For simplicity, we will suppose that in round  $g + 1$  spoiler selects a point  $d_{g+1}$  from  $\mathcal{I}_{\mathcal{A}_1}$ , but the proof is analogous if spoiler picks a point  $e_{g+1}$  from  $\mathcal{I}_{\mathcal{A}_2}$ . There are four cases:

- $d_{g+1} = d_\ell$  for some  $-2 \leq \ell \leq g$ . In this case, Duplicator responds with  $e_\ell$ , and the property trivially holds for round  $g + 1$ .
- $d_{g+1} = v$  or  $d_{g+1} = w_j^B$  or  $d_{g+1} = w_j^D$ . In any of these cases, Duplicator plays  $e_{g+1} = d_{g+1}$ . The first and second properties are trivially satisfied. For the third property, suppose first that  $d_{g+1} = v$ . Then  $T_0(c_m^0, v)$  holds in both  $\mathcal{I}_{\mathcal{A}_1}$  and  $\mathcal{I}_{\mathcal{A}_2}$ , this is the only assertion involving  $v$ , and  $c_m^0$  appears in both  $\mathbf{d}$  and  $\mathbf{e}$ . Thus, adding  $v$  to  $\mathbf{d}$  and  $\mathbf{e}$  preserves the partial isomorphism. Suppose next that  $d_{g+1} = w_j^D$  (the case where  $d_{g+1} = w_j^B$  is similar). The only assertions that  $w_j^D$  may participate in are of the form  $S_j(c_i^p, w_j^D)$ . Moreover, we know that in both  $\mathcal{I}_{\mathcal{A}_1}$  and  $\mathcal{I}_{\mathcal{A}_2}$ ,  $S_j(b_i^p, w_j^D)$  holds if and only if  $S_j(c_i^p, w_j^D)$  does, and furthermore,  $S_j(b_i^p, w_j^D)$  holds in  $\mathcal{I}_{\mathcal{A}_1}$  if and only if it holds in  $\mathcal{I}_{\mathcal{A}_2}$ . Thus, using property 1 of the IH, we know that  $S_j(d_\ell, w_j^D)$  holds in  $\mathcal{I}_{\mathcal{A}_1}$  if and only if  $S_j(e_\ell, w_j^D)$  holds in  $\mathcal{I}_{\mathcal{A}_2}$ .
- $d_{g+1} = b_i^p$  and  $d_\ell = c_i^q$  for some  $-2 \leq \ell \leq g$ . Then Duplicator plays  $b_i^p$  if  $d_\ell = e_\ell$  and otherwise plays  $c_i^p$ . Clearly the first part of the property will hold. For the second part, we note that Duplicator's choice ensures that  $\lambda$  and  $\rho$  have the same values after rounds  $g$  and  $g + 1$ , and by the induction hypothesis, we had  $\rho - \lambda > 2^{k-g}$  following round  $g$ . Hence, after round  $g + 1$  we have  $\rho - \lambda > 2^{k-g} > 2^{k-(g+1)}$ . Now consider the third condition. We let  $f$  be such that  $f(d_\ell) = e_\ell$  for every  $-2 \leq \ell \leq g + 1$ . We know from the IH that the restriction of  $\mathcal{I}_{\mathcal{A}_1}$  to  $d_{-2}, \dots, d_g$  is isomorphic to the restriction of  $\mathcal{I}_{\mathcal{A}_2}$  to  $e_{-2}, \dots, e_g$ , and we aim to show this continues to hold when we add  $d_{g+1}$  and  $e_{g+1}$ . Proving this is straightforward but quite tedious as there are many cases to treat, so we give just one case to illustrate the main ideas. Suppose that  $T_j(d_{g+1}, d_\ell)$  holds in  $\mathcal{I}_{\mathcal{A}_1}$ . Then we need to show that  $T_j(f(d_{g+1}), f(d_\ell))$  holds in  $\mathcal{I}_{\mathcal{A}_2}$ . Given the definition of  $\mathcal{I}_{\mathcal{A}_1}$ , it must be the case that  $T_j(d_{g+1}, d_\ell)$  is of the form  $T_{h_p}(b_i^p, b_i^{p+1})$  (if  $p < y - 1$ ) or  $T_{h_p}(b_i^p, b_{i+1}^0)$  (if  $p = y - 1$ ). We consider only the latter case, as the former is even simpler. If Duplicator plays  $b_i^p$ , then we know that  $\rho \geq i$  after round  $g$ . Since  $\rho > \lambda$ , it follows that Duplicator played  $b_{i+1}^0$  in response to  $d_\ell = b_{i+1}^0$ . Since

$T_{h_p}(b_i^p, b_{i+1}^0) = T_{h_p}(f(b_i^p), f(b_{i+1}^0))$  holds in  $\mathcal{I}_{\mathcal{A}_2}$ , we are done. If instead plays  $c_i^p$  in response to  $d_{g+1} = b_i^p$ , then  $i \leq \lambda$ . Since  $\rho - \lambda > 2^{k-g} > 1$  after  $g$  rounds, we must have  $\rho \geq i + 2$ , and hence  $i + 1 \leq \lambda$ . It follows that Duplicator played  $c_{i+1}^0$  in response to  $d_\ell = b_{i+1}^0$ . We have  $T_{h_p}(c_i^p, c_{i+1}^0) = T_{h_p}(f(b_i^p), f(b_{i+1}^0))$  in  $\mathcal{I}_{\mathcal{A}_2}$ , as desired. The proof that the pre-images of  $T$ -assertions in the restriction of  $\mathcal{I}_{\mathcal{A}_2}$  to  $e_{-2}, \dots, e_{g+1}$  also hold in the restriction of  $\mathcal{I}_{\mathcal{A}_1}$  to  $d_{-2}, \dots, d_{g+1}$  proceeds analogously. For  $B$ -,  $D$ -, and  $S$ -assertions, the proofs are even simpler and quite similar to those in the previous bullet.

- $d_{g+1} = c_i^p$  and  $d_\ell = b_i^q$  for some  $-2 \leq \ell \leq g$ . Then Duplicator plays  $c_i^p$  if  $d_\ell = e_\ell$  and otherwise plays  $b_i^p$ . Satisfaction of the three properties is shown analogously to the previous case.
- $d_{g+1} \in \{b_i^p, c_i^p\}$  and  $d_\ell \notin \{b_i^q, c_i^q\}$  for all  $-2 \leq \ell \leq g$  and  $0 \leq q < n$ . We only consider the case where  $d_{g+1} = b_i^p$  (the case where  $d_{g+1} = c_i^p$  proceeds analogously). We start with the second condition, since the first condition will be obviously satisfied by the choices of Duplicator. We let  $\lambda$  and  $\rho$  be the values computed following round  $g$ . If  $i \leq \lambda$ , then Duplicator plays  $c_j^p$ , and if  $i \geq \rho$ , Duplicator plays  $b_i^p$ . Note that in this case the  $\lambda$  and  $\rho$  values will not change after round  $i + 1$ , and so condition 2 will be satisfied. The remaining case is when  $\lambda < i < \rho$ . If we have  $i - \lambda \leq \rho - i$ , then Duplicator chooses  $c_j^p$ , and if  $\rho - i < i - \lambda$ , Duplicator picks  $b_j^p$ . In the former case,  $i - \lambda \leq \rho - i$  together with  $\rho - \lambda > 2^{k-g}$  from the IH yield  $\rho - i > 2^{k-g-1}$ . This is exactly what we need since after round  $g + 1$  the value of  $\rho$  remains unchanged but  $\lambda$  changes to  $i$ , and so we get the desired  $\rho - \lambda > 2^{k-(g+1)}$ . In the second case,  $\rho - i < i - \lambda$  and  $\rho - \lambda > 2^{k-g}$  yields  $i - \lambda > 2^{k-g-1}$ . As after round  $g + 1$ ,  $\lambda$  remains as before but  $\rho$  is replaced by  $i$ , we get  $\rho - \lambda > 2^{k-(g+1)}$  as required. Thus, condition 2 is satisfied. Now let us consider the final condition. As before, we will not detail the entire proof as it involves a great many cases. Instead, we illustrate the main ideas by considering what happens with  $T$ -assertions when  $\lambda < i < \rho$  (the other cases are simpler). Suppose that  $T_j(d_\ell, d_{g+1})$  holds in  $\mathcal{I}_{\mathcal{A}_1}$  (the case with  $T_j(d_\ell, d_{g+1})$  or when we start in  $\mathcal{I}_{\mathcal{A}_2}$  proceed similarly). As  $\lambda < i < \rho$ , we know that  $d_\ell$  must not have index  $i$  (otherwise either  $i \leq \lambda$  or  $\rho \leq i$ , contradicting our earlier assumption). The only possibility is thus that  $d_\ell = b_{i-1}^{y-1}$  and  $d_{g+1} = b_i^0$  and  $\lambda = i - 1$ . It follows that  $j = y - 1$  and Duplicator plays  $e_\ell = c_{i-1}^{y-1}$ . Moreover,  $i - \lambda = 1$  and  $\rho - \lambda > 2^{k-i} \geq 2$ , so  $i - \lambda < \rho - i$ , hence  $e_{g+1} = c_i^0$ . This yields the desired assertion  $T_j(e_\ell, e_{g+1}) = T_{y-1}(c_{i-1}^{y-1}, c_i^0)$  in  $\mathcal{I}_{\mathcal{A}_2}$ .

To complete the proof, we must show that  $\mathcal{A}_1$  and  $\mathcal{A}_2$  disagree on  $q$ . More specifically we prove that  $\langle \mathcal{T}, \mathcal{A}_1 \rangle \models_{cons} A(a)$  and  $\langle \mathcal{T}, \mathcal{A}_2 \rangle \not\models_{cons} A(a)$ . For the first point, suppose for a contradiction that there is a repair  $\mathcal{B}$  of  $\mathcal{A}_1$  w.r.t.  $\mathcal{T}$  such that  $\langle \mathcal{T}, \mathcal{B} \rangle \not\models A(a)$ . Then there can be no assertions in  $\mathcal{B}$  of the forms  $R_j(a, b_i^p)$  or  $R_j(a, c_i^p)$ , since any such assertion would imply  $q$  because  $\mathcal{T} \models \exists R_j \sqsubseteq A$ . Since  $\mathcal{B}$  is a maximal subset of  $\mathcal{A}$  consistent with  $\mathcal{T}$ , we know that adding any assertion  $R_j(a, b_i^p)$  from  $\mathcal{A}_1$  to  $\mathcal{B}$  will cause a contradiction. Let us first consider the assertion  $R_0(a, b_1^0)$ . We know that the assertion  $T_0(b_1^0, b_1^1)$  (or  $T_0(b_1^0, b_2^0)$ , if  $y = 1$ ) contradicts  $R_0(a, b_1^0)$ , and moreover, it is the only such assertion since there are no other assertions in  $\mathcal{A}$  involving the constant  $b_1^0$  and all assertions in  $\mathcal{A}$  involving  $a$  are of the form  $R_j(a, b_i^p)$  or  $R_j(a, b_i^p)$ , and we know

these do not appear in  $\mathcal{B}$ . First consider the case where  $y = 1$  and  $T_0(b_1^0, b_2^0) \in \mathcal{B}$ . Then  $\mathcal{B}$  cannot contain the assertion  $B_1(b_2^0)$  (or  $S_1(b_2^0, w_1^B)$ , if  $B_1 = \exists S_1$ ), since  $\exists T_0^- \sqsubseteq \neg B_1$  (and  $B_1 = B_{h_y} = B_n$ ). Thus, in order for  $\mathcal{B} \cup \{R_0(a, b_2^0)\}$  to be inconsistent with  $\mathcal{T}$ ,  $\mathcal{B}$  must possess the assertion  $T_0(b_2^0, b_3^0)$ . Continuing this line of reasoning, we are forced to infer that  $\mathcal{B}$  contains all the assertions  $T_0(b_1^0, b_2^0), T_0(b_2^0, b_3^0), \dots, T_0(b_{m-1}^0, b_m^0)$ . However, including  $T_0(b_{m-1}^0, b_m^0)$  means excluding  $B_1(b_m^0) / S_1(b_m^0, w_1^B)$ . As the latter is the only assertion in  $\mathcal{A}$  which contradicts  $R_0(a, b_m^0)$ , maximality of  $\mathcal{B}$  means  $\mathcal{B}$  contains  $R_0(a, b_m^0)$ , and so  $\langle \mathcal{T}, \mathcal{B} \rangle \models A(a)$ , contradicting our earlier assumption.

We now consider the case where  $y > 1$ . We associate to each tuple  $(i, j) \in [1, \dots, m-1] \times [0, \dots, n-1]$  the unique assertion in  $\mathcal{A}$  of the form  $T_j(b_i^p, b_i^{p+1})$  (with  $j = h_p$  and  $p < y-1$ ),  $T_j(b_i^p, b_{i+1}^0)$  (with  $j = h_p$  and  $p = y-1$ ),  $D_j(b_i^p)$  (with  $h_p < j < h_{p+1}$ ), or  $S_j(b_i^p, w_j^D)$  (with  $h_p < j < h_{p+1}$ ). Then we show by induction on  $(i, j) \in [1, \dots, m-1] \times [0, \dots, n-1]$  (using the lexicographic ordering to compare tuples) that the assertion associated with  $(i, j)$  must belong to  $\mathcal{B}$ . The base case, when  $(i, j) = (1, 0)$ , was already treated above. For the induction step, suppose that we have treated all tuples up to but not including  $(i, j)$ , and let  $p$  be such that either  $h_p < j \leq h_{p+1}$  or  $j = 0$  and  $p = 0$ . We know that  $R_j(a, b_i^p)$  does not appear in  $\mathcal{B}$ , and so there must be some assertion in  $\mathcal{B}$  with which it conflicts. The assertion associated with  $(i, j)$  does conflict with  $R_j(a, b_i^p)$ , so all that is needed is to show that there are no other ways of obtaining a conflict. First we note that because of the IH, we know that the assertion associated with the preceding tuple belongs to  $\mathcal{B}$ . It is easily verified that this assertion conflicts with  $B_j(b_i^p)$ , and thus  $B_j(b_i^p)$  (which does conflict with  $R_j(a, b_i^p)$ ) cannot appear in  $\mathcal{B}$ . Let us now consider the other assertions in  $\mathcal{A}$  which involve the individual  $b_i^p$ . First remark that any assertion  $B_\ell(b_i^p)$  with  $\ell < j$  cannot conflict with  $R_j(a, b_i^p)$ , since then  $\mathcal{T} \models \exists R_j^- \sqsubseteq \neg B_\ell$ , and we would violate our earlier assumption that the cause-conflict chain cannot be shortened. Assertions of the form  $B_\ell(b_i^p)$  with  $\ell > j$  also cannot conflict with  $R_j(a, b_i^p)$ , since otherwise we would violate the backwards relevance condition of type-2 chains (C6). Now suppose we have an assertion  $D_\ell(b_i^p)$  (or  $S_\ell(b_i^p, w_\ell^D)$ ) with  $\ell < j$ . In this case, we cannot have a conflict with  $R_j(a, b_i^p)$ , or else  $\mathcal{T} \models \exists R_j^- \sqsubseteq \neg D_\ell$ , and we would violate the forward relevance condition of cause-conflict chains. If instead we have the same type of assertion but  $\ell > j$ , then a conflict with  $R_j(a, b_i^p)$  would imply that since then  $\mathcal{T} \models \exists R_j^- \sqsubseteq \neg D_\ell$ , which violates the assumption of no shorter type-2 cause-conflict chain. Next consider an assertion  $T_{h_{p-1}}(b_i^p)$ . If it conflicts with  $R_j(a, b_i^p)$ , then we must have  $\mathcal{T} \models \exists T_{h_{p-1}}^- \sqsubseteq \neg \exists R_j^-$ , which violates forward minimality condition. The last type of assertion which can involve  $b_i^p$  is  $T_{h_p}(b_i^p, b_i^{p+1})$  (or  $T_{h_p}(b_i^p, b_{i+1}^0)$  if  $p = y-1$ ). If such an assertion were to conflict with  $R_j(a, b_i^p)$ , this would mean that  $\mathcal{T} \models \exists T_{h_p} \sqsubseteq \neg \exists R_j^-$ , which would imply the existence of a shorter type-2 chain, contrary to our earlier assumption. Thus, we have shown that all assertions from  $\mathcal{A}$  which involve  $b_i^p$ , other than the assertion associated with  $(i, j)$ , either do not belong to  $\mathcal{B}$  or do not conflict with  $R_j(a, b_i^p)$ . From this, we can conclude that the assertion associated with  $(i, j)$  belongs to  $\mathcal{B}$  (and that  $B_j(b_i^p)$  does not). In particular,  $B_j(b_i^p)$  does not belong to  $\mathcal{B}$ . As a result, we find that there are no assertions in  $\mathcal{B}$  which conflict with  $R_0(a, b_m^0)$ , contradicting the assumption that  $\langle \mathcal{T}, \mathcal{B} \rangle \not\models A(a)$ .

To see why  $\langle \mathcal{T}, \mathcal{A}_2 \rangle \not\models_{cons} A(a)$ , consider the set  $\mathcal{B} \subseteq \mathcal{A}$  defined as follows:

$$\begin{aligned}
\mathcal{B} = & \{ T_{h_p}(c_i^p, c_i^{p+1}) \mid 1 \leq i \leq m-1, 0 \leq p \leq y-2 \} \cup \\
& \{ T_{h_p}(c_i^p, c_{i+1}^0) \mid 1 \leq i \leq m-1, p = y-1 \} \cup \\
& \{ D_j(c_i^p) \mid D_j \in \mathbf{N}_C \text{ and } 1 \leq i \leq m-1, 1 \leq p \leq y-1, h_{p-1} < j < h_p \\
& \quad \text{or } 2 \leq i \leq m, h_{y-1} < j < h_y \} \cup \\
& \{ S_j(c_i^p, w_j^D) \mid D_j = \exists S_j \text{ and } 1 \leq i \leq m-1, 1 \leq p \leq y-1, \\
& \quad h_{p-1} < j \leq h_p \text{ or } 2 \leq i \leq m \text{ and } h_{y-1} < j \leq h_y \} \\
& \{ B_j(b_i^p) \mid B_j \in \mathbf{N}_C \text{ and } 1 \leq i \leq m-1, 1 \leq p \leq y-1, h_{p-1} < j \leq h_p \cup \\
& \quad \text{or } 2 \leq i \leq m \text{ and } j = 0 \text{ or } h_{y-1} < j \leq h_y \} \cup \\
& \{ S_j(b_i^p, w_j^B) \mid B_j = \exists S_j, \text{ and } 1 \leq i \leq m-1, 1 \leq p \leq y-1, \\
& \quad h_{p-1} < j \leq h_p \text{ or } 2 \leq i \leq m \text{ and } h_{y-1} < j \leq h_y \} \cup \\
& \{ T_0(c_m^{y-1}, v) \} \cup (\mathcal{A}_2 \cap \{ B_n(b_1^0), S_n(b_1^0, w_n^B) \})
\end{aligned}$$

We claim that  $\mathcal{B}$  is a repair of  $\mathcal{A}_2$  w.r.t.  $\mathcal{T}$  such that  $\langle \mathcal{T}, \mathcal{B} \rangle \not\models A(a)$ . It is straightforwardly verified that  $\mathcal{B}$  is consistent with  $\mathcal{T}$ , using the forward consistency and backward consistency properties of the cause-conflict chain. The absence of any  $R_j$  assertions together with consistency yields  $\langle \mathcal{T}, \mathcal{B} \rangle \not\models A(a)$ . All that remains to be shown is the maximality of  $\mathcal{B}$ . First consider some assertion  $R_j(a, b_i^p) \in \mathcal{A}_2$ . Then we know that  $\mathcal{B} \models B_j(b_i^p)$ , and since  $\mathcal{T} \models \exists R_j^- \sqsubseteq \neg B_j$ , it is impossible to add  $R_j(a, b_i^p)$  without introducing a contradiction. For assertions in  $\mathcal{A}_2$  of the form  $R_j(a, c_i^p)$  ( $j \notin \text{SP}$ ), the argument is similar:  $\mathcal{B} \models D_j(c_i^p)$ , and since  $\mathcal{T} \models \exists R_j^- \sqsubseteq \neg D_j$ , we cannot consistently add  $R_j(a, c_i^p)$  to  $\mathcal{B}$ . For  $R_j(a, c_i^p)$  with  $j \in \text{SP}$ , we use  $\mathcal{B} \models \exists T_j(c_i^p)$  and  $\mathcal{T} \models \exists R_j^- \sqsubseteq \neg \exists T_j$ . We next must justify the absence of  $B_j$  (or  $S_j$ ) assertions for the  $c$ -individuals and the absence of  $D_j$  (or  $S_j$ ) and  $T_j$  assertions for  $b$ -individuals. For an assertion  $B_j(c_i^p)$  (or  $S_j(c_i^p, w_j^B)$ ), we simply use the fact that either  $T_{h_{p-1}}(c_i^{p-1}, c_i^p)$  (and  $j = h_{p-1}+1$ ) or  $D_{j-1}(c_i^p)$  or  $S_{j-1}(c_i^p, w_j^D)$  appears in  $\mathcal{B}$ . In the former case, we use  $\exists T_{h_{p-1}}^- \sqsubseteq \neg B_j$ , and in the latter cases, we use the inclusion  $D_{j-1} \sqsubseteq \neg B_j$ . The argument for the missing  $D_j / S_j / T_j$  assertions for  $b$ -individuals is similar. Thus, no assertion from  $\mathcal{A}_2 \setminus \mathcal{B}$  can be consistently added to  $\mathcal{B}$ .

### C Proof of Theorem 3

We prove the two statements of the theorem in separate lemmas:

**Lemma 2.** *If there exists a type-1 cause-conflict chain for  $\mathcal{T}$  and  $\alpha$ , then there also exists a type-2 cause-conflict chain.*

*Proof.* Suppose we have a type-2 cause-conflict chain of length  $n+1$ . To turn it into a type-1 cause-conflict chain, we perform the following steps. First, we create a second chain by truncating the chain at  $\mu_n$ , reversing it, and replacing each individual  $b$  which does not appear in  $\mu_n$  by a new individual  $b'$ . Then we take the original chain up to  $\mu_n$  and add to it the second chain (identifying  $\mu_n$  and its image at the start of the second

chain). We claim that the result is a type-1 cause-conflict chain. We start by showing that it is a valid cause-conflict chain. For forward consistency, we use the fact that the original chain satisfies forward and backward consistency (so both “halves” of the chain are forward consistent), plus the fact that the second half of the chain uses new individual names and so cannot conflict with the first half of the chain. For forward minimality, the reasoning is similar: forward and backward minimality of the original chain guarantee forward minimality for each half, and then we leverage the fact that the two halves cannot interact because of their lack of shared individuals. Condition C1 and C2 are obtained by using C1 and C2 of the original chain. Backward consistency and minimality (C4 and C5) can be shown in the same way as forward consistency and minimality.

**Lemma 3.** *Assuming  $P \neq NP$ , there exists  $\mathcal{T}, \alpha$  for which a type-2 cause-conflict chain exists but no type-1 cause-conflict chain exists.*

*Proof.* Let

$$\mathcal{T} = \{ \exists R \sqsubseteq A, \exists R^- \sqsubseteq \neg \exists S, \exists R^- \sqsubseteq \neg B, \exists S^- \sqsubseteq \neg B, \}$$

be the TBox used in Example 5. We show that deciding whether  $\langle \mathcal{T}, \mathcal{A} \rangle \models_{cons} A(a)$  can be decided in polynomial time in  $|\mathcal{A}|$ . The procedure is as follows. We first check whether the following condition holds:

( $\star$ ) there exists  $c$  such that  $R(a, c) \in \mathcal{A}$  and neither  $B(c)$  nor any  $S(c, d)$  belongs to  $\mathcal{A}$

If this condition holds, then there is no assertion in  $\mathcal{A}$  which conflicts with  $R(a, c)$ , so  $R(a, c)$  must belong to all repairs of  $\langle \mathcal{T}, \mathcal{A} \rangle$ . If it does not hold, we check to see whether the next condition holds:

( $\star\star$ ) for every  $R(a, c) \in \mathcal{A}$ , there is some assertion of the form  $S(c, d)$  in  $\mathcal{A}$

If ( $\star\star$ ) holds, then we can construct a set  $\mathcal{B}$  which is consistent with  $\mathcal{T}$  and inconsistent with every cause of  $A(a)$  by selecting for every  $R(a, c)$  of  $A(a)$  an assertion of the form  $S(c, d)$ . Thus, by Proposition 1, we have shown that  $A(a)$  is not consistently entailed, and so can return no. If neither ( $\star$ ) nor ( $\star\star$ ) holds, then there is some  $R(a, c) \in \mathcal{A}$  such that  $B(c) \in \mathcal{A}$  but no  $S(c, d)$  belongs to  $\mathcal{A}$ . Clearly any repair which contradicts all causes of  $A(a)$  must include  $B(c)$ , and hence cannot include any assertion of the form  $S(e, c)$ . We can thus delete all incoming  $S$ -edges to  $c$  without affecting the existence of a repair which does not entail  $A(a)$ . We now repeat this procedure until we either satisfy ( $\star$ ) and return a positive answer, or we satisfy ( $\star\star$ ) and return a negative answer. As the size of  $\mathcal{A}$  decreases with each iteration, we can be sure that eventually one of these criteria will be satisfied, and the procedure will return the correct result. It is easy to see that the procedure runs in polynomial time in  $|\mathcal{A}|$ .

## D Proof of Theorem 4

Termination: first we note that there are only finitely many pairs  $(\mathcal{C}, \mathcal{D})$  in  $\mathcal{G}$ , and hence only finitely many iterations of the for-loop. Thus, the algorithm terminates if each of

the calls to `auxRewrite` terminates. To see why this is the case, notice that each branch in the recursion tree for `auxRewrite` constructs a cause-conflict chain (assuming we treat variables as individuals). This is because we always choose a new cause (the assertion  $\alpha$ ) which is not already contradicted by one of the assertions selected so far. Moreover, we know that the current cause has a conflict which contradicts the selected conflict of the previous cause. Finally, we always make sure to choose a conflict which is consistent with the set of assertions selected so far. Also note that by construction, the cause-conflict chain we construct will be such that  $\text{cpos}(\Sigma_i) = \text{bpos}(\Omega_i)$  for all  $i \geq 1$ . Thus, by condition **No Loops**, we cannot have  $\mathcal{Y}_i = \mathcal{Y}_j$  where  $i < j$  except in the case where  $\mathcal{Y}_j$  is the final inclusion in the chain. This means the recursion depth for `auxRewrite` cannot exceed  $|\text{CauseT}(A)| + 1$ .

Correctness: suppose that  $\mathcal{T}, A(a)$  satisfy conditions **Ordering** and **No Loops**, and let  $\neg\varphi$  be the output of `Rewrite` on input  $\mathcal{T}, A(a)$ . We aim to show that for all ABoxes  $\mathcal{A}$ , we have  $\langle \mathcal{T}, \mathcal{A} \rangle \not\models_{\text{cons}} A(a)$  if and only if  $\mathcal{I}_{\mathcal{A}} \models \varphi$ . For the first direction, suppose  $\langle \mathcal{T}, \mathcal{A} \rangle \not\models_{\text{cons}} A(a)$ , and let  $\mathcal{B}$  be a repair of  $\mathcal{A}$  w.r.t.  $\mathcal{T}$  such that  $\langle \mathcal{T}, \mathcal{B} \rangle \not\models A(a)$ . Define the sets  $\mathcal{C}$  and  $\mathcal{D}$  as follows:

$$\begin{aligned} \mathcal{C} &= \{C \in \text{CauseT}(A) \mid \exists \gamma \in \mathcal{A} \text{ with } \{\gamma\} \models C(a) \text{ and } \exists D \in \text{ConflT}(C) \text{ such that} \\ &\quad \exists \beta \in \mathcal{B} \text{ with } \{\beta\} \models D(a)\} \\ \mathcal{D} &= \{D \mid \exists C \in \mathcal{C} \text{ such that } D \in \text{ConflT}(C) \text{ and } \exists \beta \in \mathcal{B} \text{ with } \{\beta\} \models D(a)\} \end{aligned}$$

We remark that the pair  $(\mathcal{C}, \mathcal{D})$  satisfies conditions (a), (b), and (c), and so there will be a disjunct  $\psi$  of  $\varphi$  which is associated with this pair. We intend to show that the disjunct  $\psi$  is satisfied by  $\mathcal{I}_{\mathcal{A}}$ . We know that  $\psi$  has the form

$$\exists w_1 \dots w_m \bigwedge_{\beta \in S} \beta \wedge \bigwedge_{\gamma \in I} \gamma \wedge \bigwedge_{C \in U} (\forall x \text{auxRewrite}(\mathcal{T}, A(a), C, x, S))$$

where

$$\begin{aligned} \mathcal{D} &= \{B_1, \dots, B_k, \exists P_1, \dots, \exists P_\ell, \exists P_{\ell+1}^-, \dots, \exists P_m^-\} \\ S &= \{B_i(a)\}_{i=1}^k \cup \{P_i(a, w_i)\}_{i=1}^\ell \cup \{P_i(w_i, a)\}_{i=\ell+1}^m \\ I &= \{v_i \neq v_j \mid v_i, v_j \in \{a, w_1, \dots, w_m\} \text{ and } \mathcal{T}, S \cup \{v_i = v_j\} \models \perp\} \\ U &= \text{CauseT}(A) \setminus \mathcal{C} \end{aligned}$$

Because of the way we constructed  $(\mathcal{C}, \mathcal{D})$ , we know that each of the assertions  $B_i(a)$  belongs to  $\mathcal{B}$ . We further know that we can find individuals  $c_i$  such that  $P_i(a, c_i) \in \mathcal{B}$  for every  $1 \leq i \leq \ell$  and  $P_i(c_i, a) \in \mathcal{B}$  for every  $\ell + 1 \leq i \leq m$ . Define a valuation  $\sigma$  by  $\sigma(w_i) = c_i$ . For convenience, we will apply valuations not only to variables but also to formulas, with the obvious meaning. We aim to show that the formula  $\sigma(\bigwedge_{\beta \in S} \beta \wedge \bigwedge_{\gamma \in I} \gamma \wedge \bigwedge_{C \in U} (\forall x \text{auxRewrite}(\mathcal{T}, A(a), C, x, S)))$  holds in  $\mathcal{I}_{\mathcal{A}}$ . We have already shown that  $\sigma(\beta) \in \mathcal{B} \subseteq \mathcal{A}$  for every  $\beta \in S$ , and hence  $\mathcal{I}_{\mathcal{A}} \models \sigma(\bigwedge_{\beta \in S} \beta)$ . We also know that all of the inequalities in  $I$  must be satisfied by the valuation  $\sigma$  since otherwise this would imply that  $\mathcal{B}$  is not consistent with  $\mathcal{T}$ . Thus, all that remains to be shown is that each formula  $\sigma(\forall x \text{auxRewrite}(\mathcal{T}, A(a), C, x, S))$  is satisfied by  $\mathcal{I}_{\mathcal{A}}$ . In other words, we must show that for every  $\sigma'$  which extends  $\sigma$  to  $x$ , the formula

$\sigma(\forall x \text{auxRewrite}(T, A(a), C, x, S))$  is satisfied by  $\mathcal{I}_A$ . We will prove this by induction on subformulas. Specifically, we will show the following statement holds

if  $\sigma$  is a valuation of  $\text{Vars}(G) \cup \{v\}$  satisfying  $\sigma(G) \subseteq \mathcal{B}$ , then  
 $\sigma(\text{auxRewrite}(T, A(a), C, v, G))$  holds in  $\mathcal{I}_A$ .

for all subformulas  $\text{auxRewrite}(T, A(a), C, v, G)$  of  $\psi$ . The base case is when a subformula  $\text{auxRewrite}(T, A(a), C, v, G)$  does not have any proper  $\text{auxRewrite}$  subformulas, i.e. there are no subcalls to  $\text{auxRewrite}$ . If  $\sigma(\alpha)$  does not appear in  $\mathcal{A}$ , then the first disjunct of the formula is satisfied by  $\mathcal{I}_A$ . Otherwise,  $\sigma(\alpha)$  and  $C \in \text{CauseT}(A)$  implies that  $\sigma(\alpha) \notin \mathcal{B}$ . Using the maximality of  $\mathcal{B}$ , we can find an assertion  $\gamma \in \mathcal{B}$  which conflicts with  $\sigma(\alpha)$ . Note that if  $\gamma \models D(a)$  where  $D \in \text{ConflT}(C)$ , then we would have  $C \in \mathcal{C}$  and  $D \in \mathcal{D}$ , and so  $C$  would never appear as an argument to  $\text{auxRewrite}$ . Hence, it must be the case that  $C = \exists R$  for some basic role  $R$ ,  $\sigma(\alpha) = \sigma(R(a, x)) = R(a, b)$  for some  $b$ , and there is  $D \in \text{ConflT}(\exists R^-)$  such that  $\gamma \models D(b)$ . If  $D \in \mathcal{N}_C$ , then  $\sigma(\text{auxRewrite}(T, A(a), C, v, G))$  contains a disjunct  $\sigma(D(x) \wedge \epsilon)$ . We know that the first part of this disjunct will be satisfied by  $\mathcal{I}_A$  because  $\sigma(D(x)) = D(b) = \gamma \in \mathcal{B} \subseteq \mathcal{A}$ . For the second part ( $\sigma(\epsilon)$ ), we note that  $\sigma(T \cup \{\beta\}) \subseteq \mathcal{B}$ , and so consistency of  $\mathcal{B}$  ensures all inequalities in  $\epsilon$  must be satisfied by  $\sigma$ . Next consider the case where  $D = \exists T$ . In this case, there is a disjunct  $\sigma(\exists y T(x, y) \wedge \epsilon)$  in  $\text{auxRewrite}(T, A(a), C, v, G)$ . To see why it is satisfied, let  $d$  be such that  $\gamma = T(b, d)$ , and define  $\sigma'$  as the extension of  $\sigma$  to the new variable  $y$  which assigns  $\sigma'(y) = d$ . It is then easy to see that  $\sigma'(T(x, y) \wedge \epsilon)$  is satisfied in  $\mathcal{I}_A$  because of the presence of  $\gamma$  in  $\mathcal{B} \subseteq \mathcal{A}$  and because of the consistency of  $\mathcal{B}$  (which guarantees that the inequalities  $\epsilon$  hold). Thus, we have shown that there is always some disjunct of  $\text{auxRewrite}(T, A(a), C, v, G)$  which is satisfied in  $\mathcal{I}_A$ , and hence  $\text{auxRewrite}(T, A(a), C, v, G)$  holds in  $\mathcal{I}_A$ .

For the induction step, suppose that the statement holds for subformulas having at most  $k$  nested  $\text{auxRewrite}$  subformulas. Let  $\text{auxRewrite}(T, A(a), C, v, G)$  be a subformula of  $\psi$  having  $k+1$  nested  $\text{auxRewrite}$  subformulas. In the simplest case,  $\sigma(\alpha)$  does not appear in  $\mathcal{A}$ , and so the first disjunct of the formula is satisfied by  $\mathcal{I}_A$ . Let us then consider the more interesting case where  $\sigma(\alpha) \in \mathcal{A}$ . Then there must be some assertion  $\gamma \in \mathcal{B}$  which conflicts with  $\sigma(\alpha)$ . Using the same reasoning as in the base case, we deduce that there is a basic role  $R$ , an individual  $b$ , and a concept  $D$  such that  $C = \exists R$ ,  $\sigma(\alpha) = \sigma(R(a, x)) = R(a, b)$ ,  $D \in \text{ConflT}(\exists R^-)$ , and  $\gamma \models D(b)$ . We have already seen above that there is a valuation  $\sigma'$  over  $\text{Vars}(G) \cup \{v\}$  such that  $\sigma'(G \cup \{\beta\}) \subseteq \mathcal{B}$  and  $\sigma'(\beta \wedge \epsilon)$  holds in  $\mathcal{I}_A$ . We then notice that each formula subformula  $\text{auxRewrite}(T, A(a), H, u, G \cup \{\beta\})$  with  $(H, u) \in \Delta$  has at most  $k$  nested  $\text{auxRewrite}$  subformulas, and so we can apply the induction hypothesis to these formulas using the valuation  $\sigma'$ . This allows us to infer  $\sigma'(\text{auxRewrite}(T, A(a), H, u, G \cup \{\beta\}))$  holds in  $\mathcal{I}_A$  for every  $(H, u) \in \Delta$ . This concludes our proof that the formula  $\text{auxRewrite}(T, A(a), C, v, G)$  is satisfied in  $\mathcal{I}_A$ .

For the second direction, suppose  $\mathcal{I}_A \models \varphi$ . We aim to construct a set  $\mathcal{B} \subseteq \mathcal{A}$  which is consistent with  $T$  and satisfies  $\langle T, \mathcal{B} \rangle \not\models A(a)$ . As  $\mathcal{I}_A \models \varphi$ , there must be some disjunct  $\psi$  of  $\varphi$  such that  $\mathcal{I}_A \models \delta$ . Let  $(\mathcal{C}, \mathcal{D})$  be the pair of sets which were used to



produce  $\psi$ . We know that  $\psi$  has the following form:

$$\psi = \exists w_1 \dots w_m \bigwedge_{\beta \in S} \beta \wedge \bigwedge_{\gamma \in I} \gamma \wedge \bigwedge_{C \in U} (\forall x \text{auxRewrite}(\mathcal{T}, A(a), C, x, S))$$

We can thus find a valuation  $\sigma_0$  mapping variables  $w_1, \dots, w_m$  to individuals  $c_1, \dots, c_m$  such that

$$\sigma_0 \left( \bigwedge_{\beta \in S} \beta \wedge \bigwedge_{\gamma \in I} \gamma \wedge \bigwedge_{C \in U} (\forall x \text{auxRewrite}(\mathcal{T}, A(a), C, x, S)) \right)$$

holds in  $\mathcal{I}_{\mathcal{A}}$ . We define  $\mathcal{B}_0$  as the set  $\{\sigma_0(\beta) \mid \beta \in S\}$ . Note that we can be sure that  $\mathcal{B}_0$  is consistent with  $\mathcal{T}$  because  $\mathcal{D}$  is consistent with  $\mathcal{T}$  and  $\sigma$  satisfies the inequalities in  $I$ . Also note that every cause  $C(a) \in \mathcal{A}$  of  $A(a)$  with  $C \in \mathbb{N}_C$  must conflict with some assertion in  $\mathcal{B}_0$ , since otherwise  $C \notin \mathcal{C}$  would imply that there is a conjunct  $\neg C(a)$  in  $\psi$ , and such a conjunct could not be satisfied by  $\mathcal{I}_{\mathcal{A}}$ . At this point, however, there may be many role assertions which cause  $A(a)$  which do not conflict with  $\mathcal{B}_0$ . We thus need to add assertions to  $\mathcal{B}_0$  to obtain a larger set  $\mathcal{B}$  which conflicts with every cause in  $\mathcal{A}$ . In order to do so, we will construct an ordered labelled tree which will make explicit how the `auxRewrite` conjuncts are satisfied in  $\mathcal{I}_{\mathcal{A}}$  (i.e. how are variables instantiated and which disjuncts are made true), and then we will show how a traversal of this tree allows us to consistently extend  $\mathcal{B}$  so as to contradict all causes in  $\mathcal{A}$ . The nodes of the tree will correspond to pairs  $(\text{auxRewrite}(\mathcal{T}, A(a), \exists T, x, S), \sigma)$  of `auxRewrite` subformulas coupled with a valuation such that  $\sigma(\text{auxRewrite}(\mathcal{T}, A(a), \exists T, x, S))$  holds in  $\mathcal{I}_{\mathcal{A}}$ . Each such node is naturally associated with a cause for  $A(a)$ , namely  $\sigma(T(a, x))$ . Each node in the tree, excepting the root, will be labelled with an assertion which conflicts with the node's associated cause. Formally, the tree is defined as follows:

- the children of the root are the pairs  $(\text{auxRewrite}(\mathcal{T}, A(a), \exists T, x, S), \sigma)$  such that:
  - $\forall x \text{auxRewrite}(\mathcal{T}, A(a), \exists T, x, S)$  is a conjunct of  $\psi$
  - $\sigma$  extends  $\sigma_0$  to variable  $x$
  - $\sigma(T(a, x)) \in \mathcal{A}$
- the children of the root are ordered in such a way that a node  $(\text{auxRewrite}(\mathcal{T}, A(a), C_1, x, S), \sigma)$  appears to the left of a node  $(\text{auxRewrite}(\mathcal{T}, A(a), C_2, x, S), \sigma')$  whenever  $C_1 < C_2$
- for every non-root node  $n = (\text{auxRewrite}(\mathcal{T}, A(a), \exists T, x, S), \sigma)$ , let

$$\delta = (\exists y)(\beta \wedge \epsilon \wedge \bigwedge_{(H, v) \in \Delta} \text{auxRewrite}(\mathcal{T}, A(a), H, v, S \cup \{\beta\}))$$

be a disjunct of  $\text{auxRewrite}(\mathcal{T}, A(a), \exists T, x, S)$  such that  $\sigma(\delta)$  holds in  $\mathcal{I}_{\mathcal{A}}$ . If  $\delta$  is preceded by a quantifier  $\exists y$ , then let  $c$  be a constant which can be substituted for  $y$  to make  $\sigma(\delta)$  hold, and define  $\sigma'$  as the extension of  $\sigma$  to  $y$  with  $\sigma'(y) = c$ . If the quantifier was omitted, set  $\sigma' = \sigma$ . Then:

- we label node  $n$  with the assertion  $\sigma'(\beta)$
- the children of  $n$  are all pairs  $(\text{auxRewrite}(\mathcal{T}, A(a), \exists R, v, S \cup \{\beta\}), \sigma')$  such that  $(\exists R, v) \in \Delta$  and  $\sigma'(R(a, v)) \in \mathcal{A}$

The finiteness of the formula  $\psi$  ensures the finiteness of the constructed tree. A simple inductive argument shows that for every node  $(\kappa, \sigma)$ , the formula  $\kappa(\sigma)$  holds in  $\mathcal{I}_{\mathcal{A}}$  and the cause associated with  $(\kappa, \sigma)$  is present in  $\mathcal{A}$ . Finally, we note that because of the way  $\beta$  is selected, we can be sure that the label  $\sigma'(\beta)$  of a node conflicts with its associated cause.

It is now straightforward to use the tree to construct the desired  $\mathcal{B}$ . We start with  $\mathcal{B} = \mathcal{B}_0$ , and then we perform a depth-first left-to-right traversal of the tree. Whenever we are at a node whose associated cause is not yet contradicted by  $\mathcal{B}$ , we add its label to  $\mathcal{B}$  and continue our depth-first traversal. If we reach a cause which is already contradicted by  $\mathcal{B}$ , we simply skip the subtree under the current cause and pursue the traversal as if we had visited the subtree. When we return to the root, we continue on to the next unexamined child node. We aim to show that at the end of our traversal the set  $\mathcal{B}$  is consistent with  $\mathcal{T}$  and  $\langle \mathcal{T}, \mathcal{B} \cup \{\alpha\} \rangle \models \perp$  for every cause  $\alpha \in \mathcal{A}$  of the IQ  $A(a)$ . For the latter property, take some cause  $\alpha \in \mathcal{A}$ . If  $\alpha$  corresponds to a cause-type  $C \in \mathcal{C}$ , then our construction ensures that the original  $\mathcal{B}$  contains a conflict for it. Otherwise,  $\alpha$  must correspond to a cause-type  $C \in \mathcal{U}$ , which means there is child of the root whose associated cause is  $\alpha$ . It follows that at least once during the depth-first traversal we will visit a node with associated cause  $\alpha$ , and so at the first such occurrence, a conflict for  $\alpha$  will be added to  $\mathcal{B}$ .

Now we turn to consistency of  $\mathcal{B}$  with  $\mathcal{T}$ . Suppose for a contradiction that  $\mathcal{B}$  is inconsistent, and let  $\gamma_1, \gamma_2 \in \mathcal{B}$  be such that  $\langle \mathcal{T}, \{\gamma_1, \gamma_2\} \rangle \models \perp$ . We know that the original  $\mathcal{B}$ , before tree traversal, must be consistent, so at least one of  $\gamma_1$  and  $\gamma_2$  must have been added during the tree traversal. Suppose then that  $\gamma_1$  was added to  $\mathcal{B}$  before  $\gamma_2$ . We suppose without loss of generality that there is no  $\gamma'_1$  which was added before  $\gamma_1$  which conflicts with  $\gamma_2$ . First suppose that  $\gamma_1$  belongs to the original  $\mathcal{B}$  and  $\gamma_2$  was added when visiting node  $(\text{auxRewrite}(\mathcal{T}, A(a), \exists T, x, S), \sigma)$ . Then by construction,  $\gamma_1 = \sigma(\beta_1)$  for some  $\beta_1 \in S$  and  $\gamma_2 = \sigma'(\beta_2)$  where  $\beta_2$  is the first conjunct of  $\text{auxRewrite}(\mathcal{T}, A(a), \exists T, x, S)$ . The conjunct  $\epsilon$  of  $\text{auxRewrite}(\mathcal{T}, A(a), \exists T, x, S)$  will contain all inequalities that are needed to ensure that  $S \cup \{\beta_2\}$  is consistent with  $\mathcal{T}$ . As  $\sigma'$  was chosen so as to make  $\text{auxRewrite}(\mathcal{T}, A(a), \exists T, x, S)$  hold,  $\sigma'$  must verify  $\epsilon$ , and so  $\gamma_1$  and  $\gamma_2$  must be consistent, contradicting our assumption to the contrary. Exactly the same argument applies when  $\gamma_2$  was added at a node which is a successor of the node where  $\gamma_1$  was added, since we still have that  $\gamma_1 = \sigma(\beta_1)$  for some  $\beta_1 \in S$ .

Let us now consider the more interesting case where  $\gamma_1$  was added at node

$$n_1 = (\text{auxRewrite}(\mathcal{T}, A(a), \exists T_1, x_1, S_1), \sigma_1),$$

$\gamma_2$  was added at node

$$n_2 = (\text{auxRewrite}(\mathcal{T}, A(a), \exists T_2, x_2, S_2), \sigma_2),$$

and  $n_2$  is not a successor of  $n_1$ . For  $i \in \{1, 2\}$ , we let  $D_i$  be the concept in  $\text{ConfIT}(\exists T_i)$  which was selected at node  $n_i$ ;  $\delta_i$  be the disjunct of  $\text{auxRewrite}(\mathcal{T}, A(a), \exists T_i, x_i, S_i)$  which is associated with  $D_i$ ;  $\beta_i$  be the first conjunct of  $\delta_i$ ; and  $\sigma'_i$  the extension of  $\sigma_i$  to any new variable in  $\gamma_i$ . We know that  $\sigma'_i(\beta_i) = \gamma_i$ .

First consider the case where  $\gamma_1 \bullet \xrightarrow{D_1 \sqsubseteq \neg D_2} \gamma_2$ . Then we must have  $\sigma_1(x_1) = \sigma_2(x_2)$  and  $D_2 \in \text{ConflT}(\exists T_2^-) \cap \text{ConflT}(D_1)$ . We must also have  $\langle \mathcal{T}, S_1 \cup \{\beta_1\} \cup \{T_2(a, x_1)\} \rangle \not\models \perp$  since otherwise  $\sigma_2(T_2(a, x_2)) = \sigma_1(T_2(a, x_1))$  would already be contradicted by some assertion  $\mathcal{B}$  by the time we visit  $n_2$ , in which case  $\gamma_2$  would not have been added. It follows that the pair  $(\exists T_2, x_1)$  will be added to  $\Delta$ , and so  $n_1$  will have a child  $(\text{auxRewrite}(\mathcal{T}, A(a), \exists T_2, x_1, S_1 \cup \{\beta_1\}), \sigma'_1)$ . This is a contradiction since this would again imply that  $\mathcal{B}$  contains a conflict for  $\gamma_2$  before the node  $n_2$  is visited. A similar argument applies in the case where  $D_1 = \exists X$  and  $\gamma_1 \bullet \xrightarrow{D_1 \sqsubseteq \neg \exists X^-} \gamma_2$ , except this time we will have  $(\exists T_2, x_2) \in \Delta$ , yielding a child node  $(\text{auxRewrite}(\mathcal{T}, A(a), \exists T_2, x_2, S_1 \cup \{\beta_1\}), \sigma'_1)$ .

The remaining case is when we have  $D_2 = \exists Y$  and either  $\gamma_1 \bullet \xrightarrow{D_1 \sqsubseteq \neg \exists Y^-} \gamma_2$  or  $\gamma_1 \bullet \xrightarrow{\exists X^- \sqsubseteq \neg \exists Y^-} \gamma_2$  (where  $D_1 = \exists X$ ). In this case, we show that we can construct a cause-conflict chain which satisfies conditions C1 and C3. To build the chain, first let  $p_0 \dots p_k$  be the sequence of nodes on the path from the root to  $n_1$ , excluding the root. I.e.  $p_k = n_1$  and  $p_0$  is the unique child of the root which is an ancestor of  $n_1$ . For each  $1 \leq i \leq k$ , we let  $R_i$  be the cause-type associated with the node  $p_i$ , and let  $v_i$  be the variable associated with it (i.e.  $R_i$  and  $v_i$  are the third and fourth arguments of the  $\text{auxRewrite}$  expression of  $p_i$ ). We can then set  $\mu_i = R_i(a, v_i)$  and  $\Upsilon_i = \exists R_i \sqsubseteq A$ . Let  $\chi_i$  be the selected disjunct of node  $p_i$ 's associated formula. We set  $\rho_i$  equal to the first conjunct of  $\chi_i$ , and we use the inclusion  $\Gamma_i = \exists R_i^- \sqsubseteq \neg D_i$ , where  $D_i$  is the concept associated with disjunct  $\chi_i$ . For the assertion  $\lambda_i$ , we choose any assertion  $\lambda_i$  such that either  $\rho_{i-1} \bullet \xrightarrow{D_i \sqsubseteq E} \lambda_i$  (or  $\rho_{i-1} \bullet \xrightarrow{\exists V_i^- \sqsubseteq E} \lambda_i$  if  $D_i = \exists V_i$ ) and  $\lambda_i \bullet \xrightarrow{E \sqsubseteq \neg \exists R_i^-} \mu_i$  for some concept  $E$ . We know such an assertion must exist because of the way children of nodes are defined. Finally, for the end of the chain, we set  $\lambda_{k+1} = \beta_2$ ,  $\mu_{k+1} = \sigma_1(T_2(a, x_1))$ ,  $\Sigma_{k+1}$  is either  $D_1 \sqsubseteq \neg \exists Y^-$  or  $\exists X^- \sqsubseteq \neg \exists Y^-$  (where  $D_1 = \exists X$ ), and  $\Omega_{k+1} =$ . Now we can instantiate the variables with individuals in any way we like as long as we map distinct variables to distinct individuals and we ensure  $\rho_k = \gamma_1$  is mapped to  $\beta_1$ . It can be verified that we obtain in this manner a proper cause-conflict chain. The conditions defining the children of a node ensure that  $\rho_i$  are always chosen so as to not contradict  $\rho_j$  selected earlier on the path from the root (giving forward consistency) and that we only allow as child nodes causes which are not already contradicted by some previous  $\rho_j$  (ensuring forward minimality). Conditions C1 and C3 are satisfied by construction. We have thus shown that there is a cause-conflict chain satisfying C1 and C3 which starts with  $\exists R_0 \sqsubseteq A$  and ends with  $\exists T_2 \sqsubseteq A$ . Because of the property **Ordering**, we must have  $\exists T_2 < \exists R_0$ . But this is a contradiction since it implies that we would have already visited a node with associated cause  $\sigma_2(T_2(a, x_2))$  and added a conflict for it to  $\mathcal{B}$  before beginning the path to  $n_1$ , and a fortiori before visiting the node  $n_2$ .

## E Proof of Theorem 5

Given a TBox  $\mathcal{T}$  and a CQ  $q$ , we first compute (in the standard manner) a UCQ  $q' = q_1 \vee \dots \vee q_n$  such that for all ABoxes  $\mathcal{A}$ , we have  $\langle \mathcal{T}, \mathcal{A} \rangle \models q$  if and only if  $\mathcal{I}_{\mathcal{A}} \models q'$ .

Then to each disjunct  $q_i$  we add the conjunction of the negations of all atomic queries which could contradict one of the atoms in the disjunct (e.g. if a disjunct contains  $A(x)$ , and the TBox entails  $A \sqsubseteq \neg B$ , we will add  $\neg B(x)$  to the disjunct). This yields a new first-order query  $q''$ . We claim that  $q''$  is satisfied by an ABox  $\mathcal{A}$  if and only if the original CQ  $q$  is cautiously entailed from  $\mathcal{T}, \mathcal{A}$ . For the first direction, suppose that  $q$  is cautiously entailed from  $\mathcal{T}, \mathcal{A}$ , and let  $\mathcal{A}^r \subseteq \mathcal{A}$  denote the intersection of the repairs of  $\mathcal{A}$  w.r.t.  $\mathcal{T}$ . Then since  $\langle \mathcal{T}, \mathcal{A}^r \rangle \models q$ , we must also have  $\mathcal{I}_{\mathcal{A}^r} \models q'$ . There must thus be a disjunct  $\delta$  of  $q'$  which is satisfied by  $\mathcal{A}^r$ , and so we can find a match  $\pi$  for  $\delta$  in  $\mathcal{I}_{\mathcal{A}^r}$ . Now let  $\delta^*$  be the disjunct of  $q''$  which was obtained from  $\delta$ . We claim that  $\pi$  is a match for  $\delta^*$  in  $\mathcal{I}_{\mathcal{A}}$ , and hence that  $\mathcal{I}_{\mathcal{A}} \models q''$ . First note that each conjunct of  $\delta^*$  which also appears in  $\delta$  is a positive atom  $\alpha(\mathbf{v})$  for which we know  $\alpha(\pi(\mathbf{v}))$  holds in  $\mathcal{I}_{\mathcal{A}^r}$ . As  $\mathcal{A}^r \subseteq \mathcal{A}$ , the assertion  $\alpha(\pi(\mathbf{v}))$  must also hold in  $\mathcal{I}_{\mathcal{A}}$ . Consider next a new conjunct of  $\delta^*$ , which must be of one of the following forms:  $\neg A(x)$ ,  $\neg \exists y S(x, y)$ , or  $\neg \exists y S(y, x)$ . Suppose for a contradiction that a conjunct  $\neg \exists y S(x, y)$  is not satisfied. Then there must exist an individual  $c$  such that  $S(\pi(x), c)$  appears in  $\mathcal{I}_{\mathcal{A}}$ . But we know from the construction of  $q''$  that the assertion  $S(\pi(x), c)$  conflicts with assertion  $\alpha(\pi(\mathbf{v}))$  for some atom  $\alpha(\mathbf{v}) \in \delta$ . It follows that  $\alpha(\pi(\mathbf{v}))$  does not appear in  $\mathcal{I}_{\mathcal{A}^r}$ , contradicting our earlier assumption to the contrary. The argument is analogous for the two other types of new conjuncts.

For the second direction, suppose that  $\mathcal{I}_{\mathcal{A}} \models q''$ , and let  $\pi$  be a match which makes some disjunct  $\delta^*$  of  $q''$  hold true. Consider some atom  $\alpha(\mathbf{v})$  which is a conjunct of  $\delta^*$  and the corresponding disjunct  $\delta$  in  $q'$ . We know that  $\alpha(\pi(\mathbf{v}))$  holds in  $\mathcal{I}_{\mathcal{A}}$ , and we wish to show that  $\alpha(\pi(\mathbf{v}))$  holds also in  $\mathcal{I}_{\mathcal{A}^r}$ . Suppose this is not the case. Then the maximality of  $\mathcal{A}^r$  implies that there must be some assertion in  $\mathcal{A}$  which conflicts with  $\alpha(\pi(\mathbf{v}))$  given  $\mathcal{T}$ . By construction, such an assertion must conflict with one of the negative conjuncts of  $\delta^*$ , but this contradicts the fact that  $\pi$  is a match which makes  $\delta^*$  hold in  $\mathcal{I}_{\mathcal{A}}$ . Thus, it must be the case that  $\pi$  makes all conjuncts of  $\delta$  hold in  $\mathcal{I}_{\mathcal{A}^r}$ , and so  $\langle \mathcal{T}, \mathcal{A}^r \rangle \models q$ , hence  $\langle \mathcal{T}, \mathcal{A} \rangle \models_{\text{caut}} q$ .