A non-monotonic Description Logic for reasoning about typicality

L. Giordano [a,1], V. Gliozzi [b], N. Olivetti [c], G.L. Pozzato [b,1,*]

[a] *DISIT, Università del Piemonte Orientale "A. Avogadro", viale Teresa Michel, 11, 15121 Alessandria, Italy*
[b] *Dipartimento di Informatica, Università degli Studi di Torino, C.So Svizzera, 185, 10149 Torino, Italy*
[c] *Aix-Marseille Université, CNRS, LSIS UMR 7296, Campus de Saint Jérôme, Avenue Escadrille Normandie-Niemen, 13397 Marseille Cedex 20, France*

ABSTRACT

In this paper we propose a non-monotonic extension of the Description Logic $\mathcal{ALC}$ for reasoning about prototypical properties and inheritance with exceptions. The resulting logic, called $\mathcal{ALC} + \mathbf{T}_{min}$, is built upon a previously introduced (monotonic) logic $\mathcal{ALC} + \mathbf{T}$ that is obtained by adding a typicality operator $\mathbf{T}$ to $\mathcal{ALC}$. The operator $\mathbf{T}$ is intended to select the "most normal" or "most typical" instances of a concept, so that knowledge bases may contain subsumption relations of the form $\mathbf{T}(C) \sqsubseteq D$ ("$\mathbf{T}(C)$ is subsumed by D"), expressing that typical C-members are instances of concept D. From a knowledge representation point of view, the monotonic logic $\mathcal{ALC} + \mathbf{T}$ is too weak to perform inheritance reasoning. In $\mathcal{ALC} + \mathbf{T}_{min}$, in order to perform non-monotonic inferences, we define a "minimal model" semantics over $\mathcal{ALC} + \mathbf{T}$. The intuition is that preferred or minimal models are those that maximize typical instances of concepts. By means of $\mathcal{ALC} + \mathbf{T}_{min}$ we are able to infer defeasible properties of (explicit or implicit) individuals. We also present a tableau calculus for deciding $\mathcal{ALC} + \mathbf{T}_{min}$ entailment that allows to give a complexity upper bound for the logic, namely that query entailment is in co-NExp^{NP}.

© 2012 Elsevier B.V. All rights reserved.

1. Introduction

The family of Description Logics (for short: DLs) is one of the most important formalisms of knowledge representation. They have a well-defined semantics based on first-order logic and offer a good trade-off between expressivity and complexity. DLs have been successfully implemented by a range of systems and they are at the basis of languages for the semantic web such as OWL.

A DL knowledge base (KB) comprises two components: the TBox, containing the definition of concepts (and possibly roles) and a specification of inclusion relations among them, and the ABox containing instances of concepts and roles. Since the very objective of the TBox is to build a taxonomy of concepts, the need of representing prototypical properties and of reasoning about defeasible inheritance of such properties naturally arises. The traditional approach is to handle defeasible inheritance by integrating some kind of non-monotonic reasoning mechanism. This has led to study non-monotonic extensions of DLs [3,4,10,19,21,22,52,17]. However, finding a suitable non-monotonic extension for inheritance with exceptions is far from being obvious.

To give a brief account,[2] [3] proposes the extension of DL with Reiter's default logic. However, the same authors have pointed out that this integration may lead to both semantic and computational difficulties. Indeed, the unsatisfactory

---

* Corresponding author.

E-mail addresses: laura@mfn.unipmn.it (L. Giordano), gliozzi@di.unito.it (V. Gliozzi), nicola.olivetti@univ-cezanne.fr, nicola.olivetti@univ-amu.fr (N. Olivetti), pozzato@di.unito.it (G.L. Pozzato).

[1] The author is supported by the project MIUR PRIN08 "LoDeN: Logiche Descrittive Nonmonotone: Complessitá e implementazioni".

[2] A more detailed discussion on other non-monotonic extensions of DLs can be found in Section 7.

treatment of open defaults via Skolemization may lead to an undecidable default consequence relation. For this reason, [3] proposes a restricted semantics for open default theories, in which default rules are only applied to individuals explicitly mentioned in the ABox. Furthermore, Reiter's default logic does not provide a direct way of modeling inheritance with exceptions. This has motivated the study of extensions of DLs with prioritized defaults [52,4]. A more general approach is undertaken in [21], where it is proposed an extension of DL with two epistemic operators. This extension allows to encode Reiter's default logic as well as to express epistemic concepts and procedural rules.

In [10] the authors propose an extension of DL with circumscription. One of the motivating applications of circumscription is indeed to express prototypical properties with exceptions, and this is done by introducing "abnormality" predicates, whose extension is minimized. The authors provide decidability and complexity results based on theoretical analysis. A tableau calculus for circumscriptive $\mathcal{ALCO}$ is presented in [38].

In [17,18] a non-monotonic extension of $\mathcal{ALC}$ based on the application of Lehmann and Magidor's *rational closure* [43] to $\mathcal{ALC}$ is proposed. The approach is based on the introduction of a consequence relation $\vdash$ among concepts and of a consequence relation $\Vdash$ among an unfoldable KB and assertions. The authors show that such consequence relations are *rational*. It is also shown that such relations inherit the same computational complexity of the underlying DL.

Recent works discuss the combination of open and closed world reasoning in DLs. In particular, formalisms have been defined for combining DLs with logic programming rules (see, for instance, [22] and [47]). A grounded circumscription approach for DLs with local closed world capabilities has been defined in [42].

In this work, we propose a new non-monotonic logic $\mathcal{ALC} + \mathbf{T}_{min}$ for defeasible reasoning in Description Logics. The logic $\mathcal{ALC} + \mathbf{T}_{min}$ extends the monotonic logic $\mathcal{ALC} + \mathbf{T}$ introduced in [26], obtained by adding a typicality operator $\mathbf{T}$ to $\mathcal{ALC}$. Both $\mathcal{ALC} + \mathbf{T}$ and $\mathcal{ALC} + \mathbf{T}_{min}$ are based on the approach to non-monotonic reasoning pioneered by Kraus, Lehmann and Magidor (for short: KLM). KLM axiomatic systems provide a terse and well-established analysis of the core properties of non-monotonic reasoning [41]. $\mathcal{ALC} + \mathbf{T}$ is an extension of Description Logics with a semantics strongly related to the KLM preferential semantics for non-monotonic reasoning. As a further step, $\mathcal{ALC} + \mathbf{T}_{min}$ is a non-monotonic extension of $\mathcal{ALC} + \mathbf{T}$ with a minimal model semantics that allows to capture useful non-monotonic inferences that $\mathcal{ALC} + \mathbf{T}$ in itself cannot perform. We apply here our approach to the basic Description Logic $\mathcal{ALC}$. However, our approach, which consists in defining a typicality extension of DL together with its preferential semantics, is a general one: by its semantical nature it can be applied to other Description Logics. For instance, we have applied it to *low complexity Description Logics*, some results are contained in [28,29,35,33,34], and they show the feasibility of our typicality extension. Although we have not yet investigated it in details, our approach can be applied equally well to more expressive Description Logics including some combinations of number restrictions, qualified number restrictions, inverse roles, and role hierarchies, provided they have the finite model property in order to ensure both decidability and the existence of minimal models.

The operator $\mathbf{T}$ that characterizes $\mathcal{ALC} + \mathbf{T}$ provides a natural way of expressing prototypical properties, and its intended meaning is that for any concept C, $\mathbf{T}(C)$ singles out the instances of C that are considered as "typical" or "normal". Thus an assertion as

"normally, a member of the Department has lunch at the restaurant"

is represented by

$$\mathbf{T}(DepartmentMember) \sqsubseteq LunchAtRestaurant$$

As shown in [26], the operator $\mathbf{T}$ is characterized by a set of postulates that are essentially a reformulation of KLM axioms of preferential logic $\mathbf{P}$, namely the assertion $\mathbf{T}(C) \sqsubseteq D$ is equivalent to the conditional assertion $C \vdash D$ of $\mathbf{P}$. The operator $\mathbf{T}$ is non-monotonic, in the sense that from $C \sqsubseteq D$ (C is subsumed by D) we cannot infer that $\mathbf{T}(C)$ is subsumed by $\mathbf{T}(D)$: even if $C \sqsubseteq D$, $\mathbf{T}(C)$ and $\mathbf{T}(D)$ can have different properties, and we can consistently say that for some P, $\mathbf{T}(C) \sqsubseteq P$ whereas $\mathbf{T}(D) \sqsubseteq \neg P$. The semantics of the typicality operator $\mathbf{T}$ can be specified by enriching with a preference relation "$<$" standard $\mathcal{ALC}$ models. Intuitively, the domain elements that belong to the extension of $\mathbf{T}(C)$ (i.e., the typical instances of C) are elements that (i) belong to the extension of C and (ii) are minimal with respect to $<$. This semantics can be seen as a modal semantics. Indeed, the preference relation $<$ works as an accessibility relation R (with $R(x, y) \equiv y < x$) of a modality $\square$, so that we can define $\mathbf{T}(C)$ as $C \sqcap \square \neg C$. We shall see that $\square$ satisfies the properties of Gödel–Löb modal logic G.

Observe that $<$ is an "absolute" preference relation that does not take into account different aspects of a class. For instance $<$ does not allow to express the fact that x is more preferred than y with respect to aspect P_1 but not with respect to aspect P_2. We can think of extending our approach in order to deal with several preference relations (whence typicality operators) $<_{P_i}$ dependent on different aspects P_i. This might increase the expressive power. We expect however that this extension will have a price: priorities will be then needed in order to constrain the behavior of different $<_{P_1}$ and $<_{P_2}$ in particular – but not exclusively – when P_1 and P_2 are logically related. The extension with multiple preference relations will be the object of future work.

We assume that a KB comprises, in addition to the standard TBox and ABox, a set of assertions of the type $\mathbf{T}(C) \sqsubseteq D$ where D is a concept not mentioning $\mathbf{T}$. For instance, let the TBox contain:

$$\mathbf{T}(DepartmentMember) \sqsubseteq LunchAtRestaurant$$

T(*DepartmentMember* ⊓ *TemporaryWorker*) ⊑ ¬*LunchAtRestaurant*

T(*DepartmentMember* ⊓ *TemporaryWorker* ⊓ ∃*Owns.RestaurantTicket*) ⊑ *LunchAtRestaurant*

corresponding to the assertions: typical members of the Department spend their lunch break at a restaurant, whereas normally a temporary member does not have lunch at the restaurant (in order to save money), but normally a temporary member having restaurant tickets eats at the restaurant.

Suppose further that the ABox contains alternatively one of the following facts about *greg*:

1. *DepartmentMember*(*greg*)
2. *DepartmentMember*(*greg*), *TemporaryWorker*(*greg*)
3. *DepartmentMember*(*greg*), *TemporaryWorker*(*greg*), ∃*Owns.RestaurantTicket*(*greg*)

From the different combinations of TBox and one of the above ABox assertions (either 1 or 2 or 3), we would like to infer the expected (defeasible) conclusions about *greg*. These are, respectively:

1. *LunchAtRestaurant*(*greg*)
2. ¬*LunchAtRestaurant*(*greg*)
3. *LunchAtRestaurant*(*greg*)

Moreover, we would also like to infer (defeasible) properties of individuals implicitly introduced by existential restrictions, for instance, if the ABox contains

∃*HasChild.*(*DepartmentMember* ⊓ *TemporaryWorker*)(*paul*)

we would like to infer that:

∃*HasChild.*(¬*LunchAtRestaurant*)(*paul*)

Finally, adding irrelevant information should not affect the conclusions. From the TBox above, one should be able to infer as well

T(*DepartmentMember* ⊓ *Tall*) ⊑ *LunchAtRestaurant*

T(*DepartmentMember* ⊓ *TemporaryWorker* ⊓ *Tall*) ⊑ ¬*LunchAtRestaurant*

T(*DepartmentMember* ⊓ *TemporaryWorker* ⊓ ∃*Owns.RestaurantTicket* ⊓ *Tall*) ⊑ *LunchAtRestaurant*

as *Tall* is irrelevant with respect to having lunch at the restaurant or not. For the same reason, the conclusion about *greg* being an instance of *LunchAtRestaurant* or not should not be influenced by adding *Tall*(*greg*) to the ABox.

From a knowledge representation point of view, the monotonic logic $\mathcal{ALC} + \mathbf{T}$ is not sufficient to perform inheritance reasoning of the kind described above. Indeed, in order to derive the expected conclusion about *greg* from the above TBox and ABox containing, for instance, the facts *DepartmentMember*(*greg*) and *TemporaryWorker*(*greg*), we should know that *greg* is a *typical* temporary member of the Department, but we do not have this information. Similarly, in order to derive that also a typical tall member of the Department must have lunch at a restaurant, we must be able to infer or assume that a "typical tall member of the Department" is also a "typical member of the Department", since there is no reason why it should not be the case; this cannot be derived by the logic itself given the non-monotonic nature of **T**. The basic monotonic logic $\mathcal{ALC} + \mathbf{T}$ is then too weak to enforce these extra assumptions, so that we need an additional mechanism to perform defeasible inferences.

In order to perform the inferences described above, two different approaches are discussed in [26], namely:

- we can define a *completion* of an ABox: the idea is that each individual is assumed to be a typical member of the most specific concept to which it belongs. Such a completion allows to perform inferences as 1, 2, 3 above;
- we can strengthen the semantics of $\mathcal{ALC} + \mathbf{T}$ by proposing a *minimal model semantics*. Intuitively, the idea is to restrict our consideration to models that maximize typical instances of a concept.

The first proposal is computationally easier, but it presents the following difficulties:

- it is not clear how to take into account implicit individuals. The approach of completion has indeed the same limitations, concerning the treatment of implicit individuals, as default extensions of DLs (see Section 7 for details);
- the completion might be inconsistent even if the initial KB is consistent;
- it is not clear whether and how the completion has to take into account concept instances that are inferred from previous typicality assumptions introduced by the completion itself (this would require a kind of fixpoint definition).

Given the above difficulties, in this work we investigate in detail the second proposal, which is computationally more expensive, but is more powerful for inheritance reasoning. Rather than defining an ad-hoc mechanism to perform defeasible inferences or making non-monotonic assumptions, we strengthen the semantics of the logic $\mathcal{ALC} + \mathbf{T}$ by proposing a minimal model semantics. Intuitively, the idea is to restrict our consideration to models that minimize the atypical instances of a concept. In order to define the preference relation on models we take advantage of the modal semantics of $\mathcal{ALC} + \mathbf{T}$: the preference relation on models (with the same domain) is defined by comparing, for each individual, the set of modal concepts (occurring in the scope of $\square$) containing the individual in the two models. Similarly to circumscription, where we must specify a set of minimized predicates, here we must specify a set of concepts $\mathcal{L}_\mathbf{T}$ of which we want to minimize the atypical instances (it may just be the set of all concepts occurring in the knowledge base). We call the new logic $\mathcal{ALC} + \mathbf{T}_{min}$ and we denote by $\models_{min}^{\mathcal{L}_\mathbf{T}}$ semantic entailment determined by minimal models.

Taking the KB of the examples above, and letting

$$\mathcal{L}_\mathbf{T} = \{DepartmentMember, DepartmentMember \sqcap TemporaryWorker,$$

$$DepartmentMember \sqcap TemporaryWorker \sqcap \exists Owns.RestaurantTicket,$$

$$DepartmentMember \sqcap TemporaryWorker \sqcap \exists Owns.RestaurantTicket \sqcap Tall\}$$

we obtain, for instance:

1. KB $\cup$ {$DepartmentMember(greg), TemporaryWorker(greg)$} $\models_{min}^{\mathcal{L}_\mathbf{T}} \neg LunchAtRestaurant(greg)$
2. KB $\cup$ {$\exists HasChild.(DepartmentMember \sqcap TemporaryWorker)(paul)$} $\models_{min}^{\mathcal{L}_\mathbf{T}} \exists HasChild.\neg LunchAtRestaurant(paul)$
3. KB $\models_{min}^{\mathcal{L}_\mathbf{T}} \mathbf{T}(DepartmentMember \sqcap Tall) \sqsubseteq LunchAtRestaurant$

As the second example shows, we are able to infer the intended conclusion also for the implicit individuals.

Our semantic approach is seemingly close to non-monotonic extensions of DL based on *circumscription*. For this reason, we discuss in detail the relationships between our approach and the one introduced in [10], based on an extension of DLs with circumscription. We point out differences and similarities, as well as a formal relation between the two approaches. Moreover, we provide a polynomial reduction of satisfiability in concept circumscribed KBs (for which it is known [10] that satisfiability in $\mathcal{ALC}$ is NExp^{NP}-hard) to satisfiability in $\mathcal{ALC} + \mathbf{T}_{min}$ with *nominals*: by this reduction, we obtain the same hardness result for $\mathcal{ALC} + \mathbf{T}_{min}$ with nominals.

We also provide a decision procedure for checking minimal entailment in $\mathcal{ALC} + \mathbf{T}_{min}$. Our decision procedure has the form of tableau calculus, with a two-step tableau construction. The idea is that the top level construction generates open branches that are candidates to represent minimal models, whereas the auxiliary construction checks whether a candidate branch indeed represents a minimal model. Termination is ensured by means of a standard blocking mechanism. Our procedure can be used to determine constructively an upper bound of the complexity of $\mathcal{ALC} + \mathbf{T}_{min}$. Namely we obtain that checking query entailment for $\mathcal{ALC} + \mathbf{T}_{min}$ is in co-NExp^{NP}. We also show how to reduce standard reasoning problems in DLs to query entailment, obtaining complexity upper bounds for them. In detail, we show that the complexity of instance checking and of subsumption for $\mathcal{ALC} + \mathbf{T}_{min}$ is in co-NExp^{NP}, whereas the complexity of concept satisfiability for $\mathcal{ALC} + \mathbf{T}_{min}$ is in NExp^{NP}. Finally, we consider the problem of checking the satisfiability of KB (alone), and we show that its complexity for $\mathcal{ALC} + \mathbf{T}_{min}$ is EXPTIME complete.

The plan of the paper is as follows. In Section 2 we recall the monotonic logic $\mathcal{ALC} + \mathbf{T}$ introduced in [26]. In Section 3 we observe that $\mathcal{ALC} + \mathbf{T}$ is too weak to reason about typicality, then we introduce the stronger non-monotonic logic $\mathcal{ALC} + \mathbf{T}_{min}$. In Section 4 we discuss the relationships between the extension of DLs based on the $\mathbf{T}$ operator and the one based on circumscription. In Section 5 we present a tableau calculus for checking entailment in $\mathcal{ALC} + \mathbf{T}_{min}$ and study an upper bound of its complexity. In Section 6 we consider other well-known reasoning problems for $\mathcal{ALC} + \mathbf{T}_{min}$, namely *instance checking, subsumption, concept satisfiability* and *KB satisfiability*. Sections 7 and 8 concludes this work with a discussion on existing approaches to non-monotonic extensions of DLs and with some pointers to future issues. Preliminary results of this paper have been presented in [25].

2. The logic $\mathcal{ALC} + \mathbf{T}$

In this section, we recall the original $\mathcal{ALC} + \mathbf{T}$, which is an extension of $\mathcal{ALC}$ by a typicality operator $\mathbf{T}$ introduced in [26]. Given an alphabet of concept names $\mathcal{C}$, of role names $\mathcal{R}$, and of individual constants $\mathcal{O}$, the language $\mathcal{L}$ of the logic $\mathcal{ALC} + \mathbf{T}$ is defined by distinguishing *concepts* and *extended concepts* as follows:

- (Concepts)
  - $A \in \mathcal{C}$, $\top$ and $\bot$ are *concepts* of $\mathcal{L}$;
  - if $C, D \in \mathcal{L}$ and $R \in \mathcal{R}$, then $C \sqcap D$, $C \sqcup D$, $\neg C$, $\forall R.C$, $\exists R.C$ are *concepts* of $\mathcal{L}$.
- (Extended concepts)
  - if C is a concept of $\mathcal{L}$, then C and $\mathbf{T}(C)$ are *extended concepts* of $\mathcal{L}$;
  - boolean combinations of extended concepts are extended concepts of $\mathcal{L}$.

A knowledge base is a pair (TBox, ABox). TBox contains subsumptions $C \sqsubseteq D$, where $C \in \mathcal{L}$ is an extended concept of the form either C' or $\mathbf{T}(C')$, and $C', D \in \mathcal{L}$ are concepts. ABox contains expressions of the form $C(a)$ and aRb where $C \in \mathcal{L}$ is an extended concept, $R \in \mathcal{R}$, and $a, b \in \mathcal{O}$.

In order to provide a semantics to the operator $\mathbf{T}$, we extend the definition of a model used in "standard" terminological logic $\mathcal{ALC}$[3]:

Definition 1 *(Semantics of $\mathbf{T}$ with selection function).* A model is any structure

$$\langle \Delta, I, f_{\mathbf{T}} \rangle$$

where:

- Δ is the domain, whose elements are denoted with $x, y, z, \ldots$;
- I is the extension function that maps each extended concept C to $C^I \subseteq \Delta$, and each role R to an $R^I \subseteq \Delta \times \Delta$. I assigns to each atomic concept $A \in \mathcal{C}$ a set $A^I \subseteq \Delta$ and it is extended to arbitrary extended concepts as follows:
  - $\top^I = \Delta$;
  - $\bot^I = \emptyset$;
  - $(\neg C)^I = \Delta \setminus C^I$;
  - $(C \sqcap D)^I = C^I \cap D^I$;
  - $(C \sqcup D)^I = C^I \cup D^I$;
  - $(\forall R.C)^I = \{x \in \Delta \mid \forall y.(x, y) \in R^I \rightarrow y \in C^I\}$;
  - $(\exists R.C)^I = \{x \in \Delta \mid \exists y.(x, y) \in R^I \text{ and } y \in C^I\}$;
  - $(\mathbf{T}(C))^I = f_{\mathbf{T}}(C^I)$.
- Given $S \subseteq \Delta$, $f_{\mathbf{T}}$ is a function $f_{\mathbf{T}} : Pow(\Delta) \rightarrow Pow(\Delta)$ satisfying the following properties:
  - $(f_{\mathbf{T}} - 1)$ $f_{\mathbf{T}}(S) \subseteq S$;
  - $(f_{\mathbf{T}} - 2)$ if $S \neq \emptyset$, then also $f_{\mathbf{T}}(S) \neq \emptyset$;
  - $(f_{\mathbf{T}} - 3)$ if $f_{\mathbf{T}}(S) \subseteq R$, then $f_{\mathbf{T}}(S) = f_{\mathbf{T}}(S \cap R)$;
  - $(f_{\mathbf{T}} - 4)$ $f_{\mathbf{T}}(\bigcup S_i) \subseteq \bigcup f_{\mathbf{T}}(S_i)$;
  - $(f_{\mathbf{T}} - 5)$ $\bigcap f_{\mathbf{T}}(S_i) \subseteq f_{\mathbf{T}}(\bigcup S_i)$.

Intuitively, given the extension of some concept C, the selection function $f_{\mathbf{T}}$ selects the *typical* instances of C. $(f_{\mathbf{T}} - 1)$ requests that typical elements of S belong to S. $(f_{\mathbf{T}} - 2)$ requests that if there are elements in S, then there are also *typical* such elements. The following properties constrain the behavior of $f_{\mathbf{T}}$ with respect to $\cap$ and $\cup$ in such a way that they do not entail monotonicity. According to $(f_{\mathbf{T}} - 3)$, if the typical elements of S are in R, then they coincide with the typical elements of $S \cap R$, thus expressing a weak form of monotonicity (namely, *cautious monotonicity*). $(f_{\mathbf{T}} - 4)$ corresponds to one direction of the equivalence $f_{\mathbf{T}}(\bigcup S_i) = \bigcup f_{\mathbf{T}}(S_i)$, so that it does not entail monotonicity. Similar considerations apply to the equation $f_{\mathbf{T}}(\bigcap S_i) = \bigcap f_{\mathbf{T}}(S_i)$, of which only the inclusion $\bigcap f_{\mathbf{T}}(S_i) \subseteq f_{\mathbf{T}}(\bigcap S_i)$ holds. $(f_{\mathbf{T}} - 5)$ is a further constraint on the behavior of $f_{\mathbf{T}}$ with respect to arbitrary unions and intersections; it would be derivable if $f_{\mathbf{T}}$ were monotonic.

In [26], we have shown that one can give an equivalent, alternative semantics for $\mathbf{T}$ based on a *preference relation* semantics rather than on a selection function semantics. The idea is that there is a global, irreflexive and transitive relation among individuals and that the typical members of a concept C (i.e., those selected by $f_{\mathbf{T}}(C^I)$) are the minimal elements of C with respect to this relation. Observe that this notion is *global*, that is to say, it does not compare individuals with respect to a specific concept. For this reason, we cannot express the fact that y is more typical than x with respect to concept C, whereas x is more typical than y with respect to another concept D. All what we can say is that either x is incomparable with y or x is more typical than y or y is more typical than x. In this framework, an element $x \in \Delta$ is a *typical instance* of some concept C if $x \in C^I$ and there is no C-element in Δ *more typical* than x. The typicality preference relation is partial since it is not always possible to establish given two element which one of the two is more typical. Following KLM, the preference relation also satisfies a *Smoothness Condition*, which is related to the well-known *Limit Assumption* in Conditional Logics [48][4]; this condition ensures that, if the extension C^I of a concept C is not empty, then there is at least one *minimal* element of C^I. This is stated in a rigorous manner in the following definition:

Definition 2. Given an irreflexive and transitive relation $<$ over a domain Δ, called *preference relation*, for all $S \subseteq \Delta$, we define

$$Min_<(S) = \{x \in S \mid \nexists y \in S \text{ s.t. } y < x\}$$

We say that $<$ satisfies the *Smoothness Condition* if for all $S \subseteq \Delta$, for all $x \in S$, either $x \in Min_<(S)$ or $\exists y \in Min_<(S)$ such that $y < x$.

---

[3] We refer to [2] for a detailed description of the standard Description Logic $\mathcal{ALC}$.

[4] More precisely, the Limit Assumption entails the Smoothness Condition (i.e. that there are no infinite $<$ descending chains). Both properties come for free in finite models.

The following representation theorem is proved in [26]:

Theorem 1 *(Theorem 2.1 in [26]). Given any model $\langle \Delta, I, f_{\mathbf{T}} \rangle$, $f_{\mathbf{T}}$ satisfies postulates $(f_{\mathbf{T}} - 1)$ to $(f_{\mathbf{T}} - 5)$ above iff there exists an irreflexive and transitive relation $<$ on Δ, satisfying the Smoothness Condition, such that for all $S \subseteq \Delta$, $f_{\mathbf{T}}(S) = Min_<(S)$.*

Having the above Representation Theorem, from now on, we will refer to the following semantics:

Definition 3 *(Semantics of $\mathcal{ALC} + \mathbf{T}$).* A model $\mathcal{M}$ of $\mathcal{ALC} + \mathbf{T}$ is any structure

$$\langle \Delta, I, < \rangle$$

where:

- Δ is the domain;
- $<$ is an irreflexive and transitive relation over Δ satisfying the Smoothness Condition (Definition 2);
- I is the extension function that maps each extended concept C to $C^I \subseteq \Delta$, and each role R to a $R^I \subseteq \Delta \times \Delta$. I assigns to each atomic concept $A \in \mathcal{C}$ a set $A^I \subseteq \Delta$. Furthermore, I is extended as in Definition 1 with the exception of $(\mathbf{T}(C))^I$, which is defined as

$$\left(\mathbf{T}(C) \right)^I = Min_< \left(C^I \right)$$

Let us now introduce the notion of satisfiability of an $\mathcal{ALC} + \mathbf{T}$ knowledge base. In order to define the semantics of the assertions of the ABox, we extend the function I to individual constants; we assign to each individual constant $a \in \mathcal{O}$ a *distinct* domain element $a^I \in \Delta$, that is to say we enforce the *unique name assumption*. As usual, the adoption of the unique name assumption greatly simplifies reasoning about prototypical properties of individuals denoted by different individual constants. Considering the example of department staff having lunches, if (in addition to the TBox) the ABox only contains the following facts about Greg and Sara:

DepartmentMember(*greg*)

DepartmentMember(*sara*), *TemporaryWorker*(*sara*)

we would like to infer that Greg takes his lunches at the restaurant, whereas Sara does not; but without the unique name hypothesis, we cannot get this conclusion since Greg and Sara might be the same individual. To perform useful reasoning we would need to extend the language with equality and make a case analysis according to possible identities of individuals. While this is technically possible, we prefer to keep the things simple here by adopting the unique name assumption.

Definition 4 *(Model satisfying a knowledge base).* Consider a model $\mathcal{M}$, as defined in Definition 3. We extend I so that it assigns to each individual constant a of $\mathcal{O}$ an element $a^I \in \Delta$, and I satisfies the unique name assumption. Given a KB (TBox, ABox), we say that:

- $\mathcal{M}$ satisfies TBox iff for all inclusions $C \sqsubseteq D$ in TBox, $C^I \subseteq D^I$;
- $\mathcal{M}$ satisfies ABox iff: (i) for all $C(a)$ in ABox, we have that $a^I \in C^I$, (ii) for all aRb in ABox, we have that $(a^I, b^I) \in R^I$.

$\mathcal{M}$ satisfies a knowledge base if it satisfies both its TBox and its ABox. Last, a query F is entailed by KB in $\mathcal{ALC} + \mathbf{T}$ if it holds in all models satisfying KB. In this case we write $KB \models_{\mathcal{ALC}+\mathbf{T}} F$.

Notice that the meaning of $\mathbf{T}$ can be split into two parts: for any x of the domain Δ, $x \in (\mathbf{T}(C))^I$ just in case (i) $x \in C^I$, and (ii) there is no $y \in C^I$ such that $y < x$. As already mentioned in the Introduction, in order to isolate the second part of the meaning of $\mathbf{T}$ (for the purpose of the calculus that we will present in Section 5), we introduce a new modality $\square$. The basic idea is simply to interpret the preference relation $<$ as an accessibility relation. By the Smoothness Condition, it turns out that $\square$ has the properties as in Gödel–Löb modal logic of provability G. The Smoothness Condition ensures that typical elements of C^I exist whenever $C^I \neq \emptyset$, by avoiding infinitely descending chains of elements. This condition therefore corresponds to the finite-chain condition on the accessibility relation (as in G). The interpretation of $\square$ in $\mathcal{M}$ is as follows:

Definition 5. Given a model $\mathcal{M}$ as in Definition 3, we extend the definition of I with the following clause:

$$(\square C)^I = \left\{ x \in \Delta \mid \text{for every } y \in \Delta, \text{ if } y < x \text{ then } y \in C^I \right\}$$

It is easy to observe that x is a typical instance of C if and only if it is an instance of C and $\square \neg C$, that is to say:

Proposition 1. *Given a model $\mathcal{M}$ as in Definition 3, given a concept C and an element $x \in \Delta$, we have that*

$$x \in \left(\mathbf{T}(C)\right)^I \quad iff \quad x \in (C \sqcap \Box\neg C)^I$$

Since we only use $\Box$ to capture the meaning of **T**, in the following we will always use the modality $\Box$ followed by a negated concept, as in $\Box\neg C$.

The Smoothness Condition, together with the transitivity of $<$, ensures the following lemma:

Lemma 1. *Given an $\mathcal{ALC} + \mathbf{T}$ model as in Definition 3, an extended concept C, and an element $x \in \Delta$, if there exists $y < x$ such that $y \in C^I$, then either $y \in Min_<(C^I)$ or there is $z < x$ such that $z \in Min_<(C^I)$.*

Proof. Since $y \in C^I$, by the Smoothness Condition we have that either (i) $y \in Min_<(C^I)$ or (ii) there is $z < y$ such that $z \in Min_<(C^I)$. In case (i) we are done. In case (ii), since $<$ is transitive, we have also that $z < x$ and we are done. $\quad\Box$

Last, we state a theorem which will be used in the following:

Theorem 2 *(Finite model property of $\mathcal{ALC} + \mathbf{T}$). The logic $\mathcal{ALC} + \mathbf{T}$ has the finite model property.*

Proof. The theorem is a consequence of Theorems 3.1 and 3.2 in [26], which prove the soundness, the completeness and the termination of a tableau calculus for $\mathcal{ALC} + \mathbf{T}$. Indeed, if a KB is satisfiable in an $\mathcal{ALC} + \mathbf{T}$ model, then there is a tableau with a finite open branch. With a construction similar to the one used in the proof of Theorem 3.1, from this branch we can build a finite model satisfying KB. $\quad\Box$

3. The logic $\mathcal{ALC} + \mathbf{T}_{min}$

As mentioned in the Introduction, the logic $\mathcal{ALC} + \mathbf{T}$ presented in [26] allows to reason about typicality. As a difference with respect to standard $\mathcal{ALC}$, in $\mathcal{ALC} + \mathbf{T}$ we can consistently express, for instance, the fact that three different concepts, like *Department member*, *Temporary Department Member* and *Temporary Department member having restaurant tickets*, have a different status with respect to *Have lunch at a restaurant*. This can be consistently expressed by including in a knowledge base the three formulas:

$\mathbf{T}(DepartmentMember) \sqsubseteq LunchAtRestaurant$

$\mathbf{T}(DepartmentMember \sqcap TemporaryResearcher) \sqsubseteq \neg LunchAtRestaurant$

$\mathbf{T}(DepartmentMember \sqcap TemporaryResearcher \sqcap \exists Owns.RestaurantTicket) \sqsubseteq LunchAtRestaurant$

Assume that *greg* is an instance of the concept $DepartmentMember \sqcap TemporaryResearcher \sqcap \exists Owns.RestaurantTicket$. What can we conclude about *greg*? We have already mentioned that if the ABox explicitly points out that *greg* is a *typical* instance of the concept, and it contains the assertion that:

$\mathbf{T}(DepartmentMember \sqcap TemporaryResearcher \sqcap \exists Owns.RestaurantTicket)(greg)$ $\qquad\qquad(*)$

then, in $\mathcal{ALC} + \mathbf{T}$, we can conclude that

$LunchAtRestaurant(greg)$

However, if $(*)$ is replaced by the weaker

$(DepartmentMember \sqcap TemporaryResearcher \sqcap \exists Owns.RestaurantTicket)(greg)$ $\qquad\qquad(**)$

in which there is no information about the typicality of *greg*, in $\mathcal{ALC} + \mathbf{T}$ we can no longer draw this conclusion, and indeed we cannot make any inference about whether *greg* spends its lunch time at a restaurant or not. The limitation here lies in the fact that $\mathcal{ALC} + \mathbf{T}$ is *monotonic*, whereas we would like to make a non-monotonic inference. Indeed, we would like to non-monotonically assume, in the absence of information to the contrary, that *greg* is a typical instance of the concept. In general, we would like to infer that individuals are typical instances of the concepts they belong to, if this is consistent with the KB.

As a difference with respect to $\mathcal{ALC} + \mathbf{T}$, $\mathcal{ALC} + \mathbf{T}_{min}$ is *non-monotonic*, and it allows to make this kind of inference. Indeed, in $\mathcal{ALC} + \mathbf{T}_{min}$ if $(**)$ is all the information about *greg* present in the ABox, we can derive that *greg* is a typical instance of the concept, and from the inclusions above we conclude that $LunchAtRestaurant(greg)$. We have already mentioned that we obtain this non-monotonic behavior by restricting our attention to the minimal $\mathcal{ALC} + \mathbf{T}$ models. As a difference with respect to $\mathcal{ALC} + \mathbf{T}$, in order to determine what is entailed by a given knowledge base KB, we do not consider *all* models of KB but only the *minimal* ones. These are the models that minimize the number of atypical instances of concepts.

Given a KB, we consider a finite set $\mathcal{L}_{\mathbf{T}}$ of concepts occurring in the KB: these are the concepts for which we want to minimize the atypical instances. The minimization of the set of atypical instances will apply to individuals explicitly occurring in the ABox as well as to implicit individuals. We assume that the set $\mathcal{L}_{\mathbf{T}}$ contains at least all concepts C such that $\mathbf{T}(C)$ occurs in the KB. Notice that in case $\mathcal{L}_{\mathbf{T}}$ contains more concepts than those occurring in the scope of $\mathbf{T}$ in KB, the atypical instances of these concepts will be minimized but no extra properties will be inferred for the typical instances of the concepts, since the KB does not say anything about these instances.

We have seen that $(\mathbf{T}(C))^I = (C \sqcap \Box \neg C)^I$: x is a typical instance of a concept C ($x \in (\mathbf{T}(C))^I$) when it is an instance of C and there is no other instance of C preferred to x, i.e. $x \in (C \sqcap \Box \neg C)^I$. By contraposition an instance of C is atypical if $x \in (\neg\Box\neg C)^I$ therefore in order to minimize the atypical instances of C, we minimize the instances of $\neg\Box\neg C$. Notice that this is different from maximizing the instances of $\mathbf{T}(C)$. We have adopted this solution since it allows to maximize the set of typical instances of C without affecting the extension C^I of C (whereas maximizing the extension of $\mathbf{T}(C)$ would imply maximizing also the extension of C).

We define the set $\mathcal{M}^{\Box^-}_{\mathcal{L}_{\mathbf{T}}}$ of negated boxed formulas holding in a model, relative to the concepts in $\mathcal{L}_{\mathbf{T}}$:

Definition 6. Given a model $\mathcal{M} = \langle \Delta, I, < \rangle$ and a set of concepts $\mathcal{L}_{\mathbf{T}}$, we define

$$\mathcal{M}^{\Box^-}_{\mathcal{L}_{\mathbf{T}}} = \left\{ (x, \neg\Box\neg C) \mid x \in (\neg\Box\neg C)^I, \text{ with } x \in \Delta, \ C \in \mathcal{L}_{\mathbf{T}} \right\}$$

Let KB be a knowledge base and let $\mathcal{L}_{\mathbf{T}}$ be a set of concepts occurring in KB.

Definition 7 *(Preferred and minimal models).* Given a model $\mathcal{M} = \langle \Delta_{\mathcal{M}}, I_{\mathcal{M}}, <_{\mathcal{M}} \rangle$ of KB and a model $\mathcal{N} = \langle \Delta_{\mathcal{N}}, I_{\mathcal{N}}, <_{\mathcal{N}} \rangle$ of KB, we say that $\mathcal{M}$ is preferred to $\mathcal{N}$ with respect to $\mathcal{L}_{\mathbf{T}}$, and we write $\mathcal{M} <_{\mathcal{L}_{\mathbf{T}}} \mathcal{N}$, if the following conditions hold:

- $\Delta_{\mathcal{M}} = \Delta_{\mathcal{N}}$;
- $a^{I_{\mathcal{M}}} = a^{I_{\mathcal{N}}}$ for all individual constants $a \in \mathcal{O}$;
- $\mathcal{M}^{\Box^-}_{\mathcal{L}_{\mathbf{T}}} \subset \mathcal{N}^{\Box^-}_{\mathcal{L}_{\mathbf{T}}}$.

A model $\mathcal{M}$ is a *minimal model* for KB (with respect to $\mathcal{L}_{\mathbf{T}}$) if it is a model of KB and there is no a model $\mathcal{M}'$ of KB such that $\mathcal{M}' <_{\mathcal{L}_{\mathbf{T}}} \mathcal{M}$.

Given the notion of preferred and minimal models above, we introduce a notion of *minimal entailment*, that is to say we restrict our consideration to minimal models only. First of all, we introduce the notion of *query*, which can be minimally entailed from a given KB. A query F is a formula of the form $C(a)$ where C is an extended concept and $a \in \mathcal{O}$. We assume that, for all $\mathbf{T}(C')$ occurring in F, $C' \in \mathcal{L}_{\mathbf{T}}$. Given a KB and a model $\mathcal{M} = \langle \Delta, I, < \rangle$ satisfying it, we say that a query $C(a)$ holds in $\mathcal{M}$ if $a^I \in C^I$.

Let us now define minimal entailment of a query in $\mathcal{ALC} + \mathbf{T}_{min}$. In Section 6 we will reduce the other standard reasoning tasks to minimal entailment.

Definition 8 *(Minimal entailment in $\mathcal{ALC} + \mathbf{T}_{min}$).* A query F is minimally entailed from a knowledge base KB with respect to $\mathcal{L}_{\mathbf{T}}$ if it holds in all models of KB that are minimal with respect to $\mathcal{L}_{\mathbf{T}}$. We write $\text{KB} \models^{\mathcal{L}_{\mathbf{T}}}_{min} F$.

The non-monotonic character of $\mathcal{ALC} + \mathbf{T}_{min}$ also allows to deal with the following examples.

Example 1. Consider the following KB:

$$\text{KB} = \left\{ \mathbf{T}(Athlet) \sqsubseteq Confident, Athlet(john), Finnish(john) \right\}$$

and $\mathcal{L}_{\mathbf{T}} = \{Athlet, Finnish\}$. We have

$$\text{KB} \models^{\mathcal{L}_{\mathbf{T}}}_{min} Confident(john)$$

Indeed, there is no minimal model of KB that contains a non-typical instance of some concept (indeed in all minimal models of KB the relation $<$ is empty). Hence *john* is an instance of $\mathbf{T}(Athlet)$ (it can be easily verified that any model in which *john* is not an instance of $\mathbf{T}(Athlet)$ is not minimal). By KB, in all these models, *john* is an instance of *Confident*. Observe that *Confident(john)* is obtained, in spite of the presence of the irrelevant assertion *Finnish(john)*.

Example 2. Consider now the knowledge base KB$'$ obtained by adding to KB the formula $\mathbf{T}(Athlet \sqcap Finnish) \sqsubseteq \neg Confident$, that is to say:

$$\text{KB}' = \left\{ \mathbf{T}(Athlet) \sqsubseteq Confident, \mathbf{T}(Athlet \sqcap Finnish) \sqsubseteq \neg Confident, Athlet(john), Finnish(john) \right\}$$

and to $\mathcal{L}_{\mathbf{T}}$ the concept *Athlet ⊓ Finnish*. From KB′, *Confident(john)* is no longer derivable. Instead, we have that

$$\text{KB}' \models^{\mathcal{L}_{\mathbf{T}}}_{min} \neg Confident(john)$$

Indeed, by reasoning as above it can be shown that in all the minimal models of KB′, *john* is an instance of $\mathbf{T}(Athlet \sqcap Finnish)$, and it is no longer an instance of $\mathbf{T}(Athlet)$. This example shows that, in case of conflict (here, *john* cannot be both a typical instance of *Athlet* and of *Athlet ⊓ Finnish*), typicality in the more specific concept is preferred.

In general, a knowledge base KB may have no minimal model or more than one minimal model, with respect to a given $\mathcal{L}_{\mathbf{T}}$. The following properties hold.

Proposition 2. *If KB has a model, then KB has a minimal model with respect to any $\mathcal{L}_{\mathbf{T}}$.*

The above fact is a consequence of the *finite model property* of the logic $\mathcal{ALC} + \mathbf{T}$ (Theorem 2).

Proposition 3. *Given a knowledge base KB and a query F, let us replace all occurrences of $\mathbf{T}(C)$ in KB and in F with C. We call KB′ the resulting knowledge base and F′ the resulting query. If $\text{KB} \models^{\mathcal{L}_{\mathbf{T}}}_{min} F$ then $\text{KB}' \models_{\mathcal{ALC}+\mathbf{T}} F'$.*

Proof. We show the contrapositive that if $\text{KB}' \not\models_{\mathcal{ALC}+\mathbf{T}} F'$ then $\text{KB} \not\models^{\mathcal{L}_{\mathbf{T}}}_{min} F$. Let $\mathcal{M}$ be an $\mathcal{ALC} + \mathbf{T}$ model satisfying KB′ and not satisfying F′. Since neither KB′ nor F′ contain any occurrence of $\mathbf{T}$, the relation $<$ does not play any role in $\mathcal{M}$ and we can assume that $<$ is empty. Notice that in $\mathcal{M}$, for all C, we have that $\mathbf{T}(C)^I = C^I$. Therefore it can be shown by induction on the complexity of formulas in KB and in F that $\mathcal{M}$ is also a model of KB that does not satisfy F.

Furthermore, by Definition 5, for all C: $(\neg\Box\neg C)^I = \emptyset$, hence $\mathcal{M}$ is a minimal model of KB. We therefore conclude that $\text{KB} \not\models^{\mathcal{L}_{\mathbf{T}}}_{min} F$, and the proposition follows by contraposition. □

The above proposition shows that the inferences allowed by $\mathcal{ALC} + \mathbf{T}_{min}$ have as upper approximation the consequences that can be drawn classically from the knowledge base KB′ obtained by transforming $\mathbf{T}(C) \sqsubseteq C'$ into the trivial $C \sqsubseteq C'$, what corresponds to assume that all individuals are typical. Obviously the KB′ may be inconsistent or degenerated (all concepts are empty), whereas the original KB is not. For this reason the inverse of the proposition obviously does not hold.

4. $\mathcal{ALC} + \mathbf{T}_{min}$ and circumscribed knowledge bases

Among the approaches to non-monotonic DL, the one based on circumscription is perhaps the closest to ours as both are based on the idea of minimizing atypical, or "abnormal" members of a concept. For this reason it is worthwhile to investigate the relations between the two approaches. In [10] the authors propose an extension of $\mathcal{ALC}$ with circumscription. One of the motivating applications of circumscription is to express prototypical properties that can have exceptions, and this is done by introducing "abnormality" predicates, whose extension is minimized. In order to express that "Typical Cs are Ds", that in $\mathcal{ALC} + \mathbf{T}_{min}$ we express as $\mathbf{T}(C) \sqsubseteq D$, the authors introduce the inclusion $C \sqsubseteq D \sqcup Ab_C$ (or equivalently $C \sqcap \neg Ab_C \sqsubseteq D$) where Ab_C ("abnormal C") is a predicate to be minimized[5]: roughly speaking the attention will be restricted to models in which the extension of Ab_C will be as little as possible. More precisely, circumscribed knowledge bases as defined in [10] are knowledge bases equipped with a so-called circumscription pattern: a tuple $CP = (\prec, M, Fix, V)$ where M are the predicates and roles to be minimized, *Fix* are the predicates to be kept fixed, V are the predicates that vary, and $\prec$ is a strict partial order over M that allows to express priorities among predicates to be minimized. As pointed out by the authors, these priorities usually reflect the taxonomy described by the TBox and since the subsumption hierarchy is a partial order, priorities are assumed to form a partial order too, as a difference with standard prioritized circumscription which assumes a total ordering. In order to establish a formal relation between $\mathcal{ALC} + \mathbf{T}_{min}$ and circumscribed knowledge bases, in this section we restrict our attention to simple circumscribed knowledge bases in which: (1) there are no fixed predicates ($Fix = \emptyset$), (2) there are no priorities over M ($\prec = \emptyset$), and (3) M only contains abnormality predicates with form Ab_C. Restrictions (1), (2) and (3) are motivated by the fact that this is the case which is closer to our approach. Notice however that all the following considerations and results still hold for $\prec$ that imposes specificity: more specific abnormality predicates must be minimized before more general ones (and for instance $Ab_{A\sqcap B} \prec Ab_A$). A simplified version of circumscribed knowledge bases without priorities is considered in [9]. As for $\mathcal{ALC} + \mathbf{T}_{min}$, for circumscribed knowledge bases one only considers *minimal* models w.r.t. the preference relation $<_{CP}$ over models which, in the restricted case we are considering, can be defined as: given two models I and J, we define $I <_{CP} J$ if (i) $\Delta^I = \Delta^J$, (ii) for all individual constants a $a^I = a^J$, (iii) for all $Ab_A \in M$,

---

[5] Notice that this is one of the possible uses of circumscription to formalize commonsense reasoning. A more sophisticated way suggested in [46] and used in [5] represents the information that an individual a is an abnormal instance of a class C with respect to a given aspect P by means of binary abnormality predicates such as $Ab(P, x)$. We consider here the simplified version of circumscription because it is the closer to our approach. As already mentioned in the Introduction, in future work we will consider the problem of parameterizing the preference relations to distinct aspects by considering a family of preference relations $<_P$ for each aspect P rather than the single preference relation $<$.

$Ab_A{}^I \subseteq Ab_A{}^J$, and (iv) there is an $Ab_A \in M$ s.t. $Ab_A{}^I \subset Ab_A{}^J$. We write $KB \models_{CIRC}^{CP} F$ if F holds in all minimal models of KB w.r.t. $<_{CP}$.

The question now is: do the inferences that can be done in circumscribed knowledge bases with the above restrictions 1, 2, and 3 coincide with those that can be done in $\mathcal{ALC} + \mathbf{T}_{min}$?

We will see that although there are major similarities between the two approaches, there are also some differences. First we provide some examples of the similarities and differences in the two formalisms. Then in Lemmas 2, 3, and Theorem 3, we formally analyze under what conditions the two formalisms coincide.

4.1. Similarities

Both formalisms are based on the idea of minimizing atypical, or abnormal instances of concepts. By this fact, many inferences coincide in the two logics.

We consider an adaptation of a well-known example provided by [24], and used by [23]. Consider a knowledge base saying that typical molluscs are shell-bearers, and that Fred is a mollusc. In the two formalisms this knowledge base would be formalized as

$$KB = \big\{ \mathbf{T}(Mollusc) \sqsubseteq ShellBearer, Mollusc(fred) \big\}$$

and

$$KB = \big\{ Mollusc \sqcap \neg Ab_{Mollusc} \sqsubseteq ShellBearer, Mollusc(fred) \big\}$$

respectively.

Both logics would infer that *Fred* is a *typical, not abnormal* mollusc ($\mathbf{T}(Mollusc)(fred)$ and $\neg Ab_{Mollusc}(fred)$), and therefore *ShellBearer*(*fred*), although this is not explicitly said in KB.

Suppose now we added to the knowledge base above the information that cephalopods are exceptional molluscs because they typically do not have a shell. This addition would result in the addition to KB of the following sets formulas: in $\mathcal{ALC} + \mathbf{T}_{min}$

$$\big\{ Cephalopod \sqsubseteq Mollusc, \mathbf{T}(Cephalopod) \sqsubseteq \neg ShellBearer \big\}$$

and, in circumscribed knowledge bases

$$\{ Cephalopod \sqsubseteq Mollusc, Cephalopod \sqcap \neg Ab_{Cephalopod} \sqsubseteq \neg ShellBearer \}.$$

Both logics infer (in the absence of information to the contrary) that there are no cephalopods ($Cephalopod \sqsubseteq \bot$), since these are atypical molluscs. In general, in both logics if there is a concept C that is exceptional with respect to the typical properties of a more general concept D, then C is assumed to be empty, in the absence of information to the contrary.

What happens if we added to the KBs above the information that *Jim* is a cephalopod ($Cephalopod(jim)$)? First of all, obviously in neither of the two logics it would be inferred that $Cephalopod \sqsubseteq \bot$ anymore.[6] However, in $\mathcal{ALC} + \mathbf{T}_{min}$ we would derive that $\mathbf{T}(Cephalopod)(jim)$ and $\neg ShellBearer(jim)$, whereas in circumscribed knowledge bases (without priorities) we would not make this inference. This is the *Specificity Principle*, that ensures that an individual is assumed to be a typical instance of the most specific concept it belongs to. The Specificity Principle holds in $\mathcal{ALC} + \mathbf{T}_{min}$ as a consequence of the fact that the semantics of $\mathbf{T}$ is based on the properties of KLM logic $\mathbf{P}$, whereas it does not hold in circumscribed knowledge bases simplified as above (with $\prec = \emptyset$). In prioritized circumscribed knowledge bases in which $Ab_{Cephalopod} \prec Ab_{Mollusc}$ one could infer that $\neg Ab_{Cephalopod}(jim)$ and therefore $\neg ShellBearer(jim)$, as in $\mathcal{ALC} + \mathbf{T}_{min}$. Notice that this behavior holds for free in $\mathcal{ALC} + \mathbf{T}_{min}$ whereas in circumscribed knowledge bases it requires to specify a priority between abnormality predicates.

4.2. Differences

We here provide two examples of the different inferences that can be drawn in the two formalisms. In both examples the differences are due to the fact that in $\mathcal{ALC} + \mathbf{T}_{min}$ the typicality operator $\mathbf{T}$ has some properties (such as the Smoothness Condition or the constraints on the possible combinations of typicality assumptions) that do not hold in circumscribed knowledge bases.

Example 1. Suppose we added to the knowledge base in the example above the information that Jim is *not* a typical cephalopod. Do we want to conclude that Jim is a typical mollusc or not? The correct answer is unclear but probably we would not want to draw this conclusion since cephalopods (also atypical ones) are usually atypical molluscs (recall that

---

[6] In circumscription an alternative way to block inferences such as the one that $Cephalopod \sqsubseteq \bot$ is to fix predicates. Our semantics does not treat fixed predicates. Handling fixed predicates will be the object of future research.

typicality is here an absolute notion and we cannot distinguish between typicality with respect to the concept of being a shell bearer and typicality with respect to other concepts). $\mathcal{ALC} + \mathbf{T}_{min}$ and circumscribed knowledge bases draw opposite conclusions. In $\mathcal{ALC} + \mathbf{T}_{min}$ we would derive that Jim is not a typical mollusc either (i.e., $\neg\mathbf{T}(Mollusc)(jim)$), whereas in circumscription (also in the prioritized version) we would derive that he is a typical mollusc $((\neg Ab_{Mollusc})(jim))$. Indeed, in $\mathcal{ALC} + \mathbf{T}_{min}$ we have strong constraints on the possible combinations of typicality throughout subclasses: a cephalopod cannot be a typical mollusc without being also a typical cephalopod. This is due to the fact that in $\mathcal{ALC} + \mathbf{T}_{min}$ we have that $(\mathbf{T}(M) \sqcap C)^I \subseteq \mathbf{T}(M \sqcap C)^I$.

Another example of the different behavior of the two formalisms comes from another property of our $\mathbf{T}$ operator that is not enforced in circumscribed DLs: the Smoothness Condition.

Example 2. In the knowledge base of the previous example there is no information that forces us to conclude that typical cephalopods exist: the only cephalopod we are aware of is Jim who is not a typical cephalopod. Can we conclude that typical cephalopods don't exists, i.e. $\mathbf{T}(Cephalopod) \sqsubseteq \bot$ (or equivalently $Cephalopod \sqcap \neg Ab_{Cephalopod} \sqsubseteq \bot$)? The answer again is different in the two formalisms. In circumscription (also in the prioritized version) the answer is yes: we conclude that $Cephalopod \sqcap \neg Ab_{Cephalopod} \sqsubseteq \bot$. On the other hand, in $\mathcal{ALC} + \mathbf{T}_{min}$ by the Smoothness Condition we have that if cephalopods exist then also typical ones exist and therefore we conclude that $\mathbf{T}(Cephalopod) \not\sqsubseteq \bot$.

4.3. Formal relation between circumscribed KBs and $\mathcal{ALC} + \mathbf{T}_{min}$

First of all we define the following natural translation of formulas from the language of $\mathcal{ALC} + \mathbf{T}_{min}$ to the language of circumscribed knowledge bases.

Definition 9. The translation of an $\mathcal{ALC} + \mathbf{T}_{min}$ formula into a circumscribed KB formula is obtained by replacing each occurrence of $\mathbf{T}(A)$ with $A \sqcap \neg Ab_A$.

Since $\mathbf{T}(A)$ can be equivalently defined as $A \sqcap \square \neg A$, the above Definition 9 entails that we translate $\square \neg A$ with $\neg Ab_A$. This is in accordance with the intuition underlying the two formalisms.

As we said above in $\mathcal{ALC} + \mathbf{T}_{min}$ there are strong constraints on the possible combinations of typicality, and of boxed formulas whereas in circumscribed knowledge bases there are no equivalent constraints on the abnormality operator. We therefore impose these constraints in a circumscribed KB (call it KB_{CIRC}) in order to obtain a KB which is equivalent, and leads to the same inferences as the starting $\mathcal{ALC} + \mathbf{T}_{min}$ KB (call it $KB_{\mathcal{ALC}+\mathbf{T}_{min}}$). The equivalence between $KB_{\mathcal{ALC}+\mathbf{T}_{min}}$ and the KB resulting from the addition of these constraints to KB_{CIRC} is formally stated in Theorem 3 below. Examples of constraints that hold for typicality and boxed formulas in $\mathcal{ALC} + \mathbf{T}_{min}$ but do not hold for abnormality predicates in circumscribed KBs and must therefore be explicitly added are the following:

(i) In $\mathcal{ALC} + \mathbf{T}_{min}$ it holds that $(\mathbf{T}(A) \sqcap \mathbf{T}(B))^I \subseteq \mathbf{T}(A \sqcup B)^I$, and it holds that $(\square \neg A \sqcap \square \neg B)^I = (\square \neg(A \sqcup B))^I$: this gives a relation between typical $(A \sqcup B)$s, typical As and typical Bs. No such relation exists for abnormality predicates in circumscription. If we want an equivalent relation for abnormality predicates we have to require that if $x \in Ab_{A \sqcup B}{}^I$, then $x \in Ab_A{}^I$ or $x \in Ab_B{}^I$, and vice versa.

(ii) Similarly, in $\mathcal{ALC} + \mathbf{T}_{min}$ we have that $\mathbf{T}(A)^I \cap B^I \subseteq \mathbf{T}(A \sqcap B)^I$, and $(\square \neg A \sqcup \square \neg B)^I \subseteq \square \neg(A \sqcap B)$, which establish a relation between typical $A \sqcap B$s and typical As. Translated into circumscription terms this corresponds to enforcing that if $x \in Ab_{A \sqcap B}$, then $x \in Ab_A$ and $x \in Ab_B$.

(iii) By the Smoothness Condition, in $\mathcal{ALC} + \mathbf{T}_{min}$ if there is an atypical A, then there is also a typical A. In circumscribed knowledge bases this amounts to requiring that if $(A \sqcap Ab_A)^I \neq \emptyset$, then $(A \sqcap \neg Ab_A)^I \neq \emptyset$.

Is there a simple way of imposing these constraints to abnormality predicates, in order to obtain a correspondence with the typicality operator in $\mathcal{ALC} + \mathbf{T}_{min}$? The answer is constraint (Constr) below. It can be verified that constraint (Constr) enforces (i), (ii), (iii) above, and indeed ensures that the behavior of abnormality predicates corresponds to the properties of $\mathbf{T}$.

$$Ab_B \sqcap \neg Ab_{A_1} \sqcap \cdots \sqcap \neg Ab_{A_n} \sqsubseteq \exists R_{B, A_1, \ldots, A_n}.(B \sqcap \neg A_1 \sqcap \cdots \sqcap \neg A_n \sqcap \neg Ab_B \sqcap \neg Ab_{A_1} \sqcap \cdots \sqcap \neg Ab_{A_n}) \qquad \text{(Constr)}$$

where role $R_{B, A_1, \ldots, A_n}$ is new for each instance of (Constr).

We consider a constraint (Constr) for each subset $\{B, A_1, \ldots, A_n\}$ of $\mathcal{L}_{\mathbf{T}}$.

(Constr) is better understood if we separately consider its components. Let us first consider: $Ab_B \sqcap \neg Ab_{A_1} \sqcap \cdots \sqcap \neg Ab_{A_n} \sqsubseteq \exists R_{B, A_1, \ldots, A_n}.B \sqcap \neg A_1 \sqcap \cdots \sqcap \neg A_n$. Only if abnormality predicates satisfy this property there is a hope that they behave as typicality assertions of $\mathcal{ALC} + \mathbf{T}_{min}$. Indeed, the property requires that if x is an abnormal, atypical instance of B, whereas it is a typical $A_1, \ldots, A_n$ element, then there must be a y preferred to x which is a B element but a $\neg A_1, \ldots, \neg A_n$ element (otherwise it would conflict with the typicality of x w.r.t. $A_1, \ldots, A_n$). Furthermore, by the Smoothness Condition, there must be a *typical B* element y preferred to x such that all the elements preferred to it do not satisfy B nor $A_1, \ldots, A_n$

(otherwise these elements would contradict that x is a typical $A_1, \ldots, A_n$ element). This is why on the right hand side of (Constr) it appears that $\neg Ab_B \sqcap \neg Ab_{A_1} \sqcap \cdots \sqcap \neg Ab_{A_n}$ (the fact that y is an instance of $\neg Ab_B \sqcap \neg Ab_{A_1} \sqcap \cdots \sqcap \neg Ab_{A_n}$ guarantees that all elements preferred to y are instances of $\neg B, \neg A_1, \ldots, \neg A_n$).

We call KB^*_{CIRC} the circumscribed KB resulting from the addition to KB_{CIRC} of all instances of (Constr). We show that $KB_{\mathcal{ALC}+\mathbf{T}_{min}} \models^{\mathcal{L}_\mathbf{T}}_{min} F$ iff $KB^*_{CIRC} \models^{CP}_{CIRC} F'$, where F' is the translation of F (see Theorem 3 below), where the circumscription pattern $CP = (\prec, M, Fix, V)$ is defined as: $\prec = \emptyset$, $Fix = \emptyset$, $M = \{Ab_A: A \in \mathcal{L}_\mathbf{T}\}$ (see Definition 8), and V contains all the predicates occurring in the KB^*_{CIRC} except from those in M.

In the following lemma we consider the model $\mathcal{M}_{CIRC}$ for which we define $\mathcal{M}_{CIRC}{}^{Ab} = \{(x, Ab_A) \text{ s.t. } Ab_A \in M \text{ and } x \in Ab_A{}^I$ in $\mathcal{M}_{CIRC}\}$.

Lemma 2. *Given any $\mathcal{ALC} + \mathbf{T}_{min}$ model $\mathcal{M}_{\mathcal{ALC}+\mathbf{T}_{min}}$ satisfying $KB_{\mathcal{ALC}+\mathbf{T}_{min}}$, there is a model $\mathcal{M}_{CIRC}$ satisfying KB^*_{CIRC} such that the domain of $\mathcal{M}_{\mathcal{ALC}+\mathbf{T}_{min}}$ and $\mathcal{M}_{CIRC}$ coincide and for all Ab_A in M, $(x, Ab_A) \in \mathcal{M}_{CIRC}{}^{Ab}$ iff $(x, \neg\Box\neg A) \in \mathcal{M}^{\Box-}_{\mathcal{ALC}+\mathbf{T}_{min}}$. The other direction from $\mathcal{M}_{CIRC}$ to $\mathcal{M}_{\mathcal{ALC}+\mathbf{T}_{min}}$ also holds.*

Proof. $\Rightarrow$ Consider $\mathcal{M}_{\mathcal{ALC}+\mathbf{T}_{min}} = \langle \Delta, I, < \rangle$ satisfying $KB_{\mathcal{ALC}} + \mathbf{T}_{min}$. We can build $\mathcal{M}_{CIRC} = \langle \Delta, I' \rangle$ satisfying KB^*_{CIRC} such that the domains Δ of the two models coincide. I' of $\mathcal{M}_{CIRC}$ is defined as I of $\mathcal{M}_{\mathcal{ALC}+\mathbf{T}_{min}}$ except from the fact that for all Ab_A in M

(i) $Ab_A{}^{I'} = (\neg\Box\neg A)^I$;
(ii) $R^{I'}_{B, A_1, \ldots, A_n} = \{(x, y): x \in (Ab_B \sqcap \neg Ab_{A_1} \sqcap \cdots \sqcap \neg Ab_{A_n})^{I'}$ and $y \in (B \sqcap \neg A_1 \sqcap \cdots \sqcap A_n \sqcap \neg Ab_B \sqcap \neg Ab_{A_1} \sqcap \cdots \sqcap \neg Ab_{An})^{I'}\}$.

By (i) we have that $\neg Ab_A{}^{I'} = (\Box\neg A)^I$ and since $\mathbf{T}(A)^I = (A \sqcap \Box\neg A)^I$, for each $\mathcal{ALC} + \mathbf{T}_{min}$ formula F and its translation F', $x \in F^I$ in $\mathcal{M}_{\mathcal{ALC}+\mathbf{T}_{min}}$ iff $x \in F^{I'}$ in $\mathcal{M}_{CIRC}$. It follows that the model obtained is a model of KB_{CIRC}. Furthermore, it is also a model of KB^*_{CIRC} since it satisfies all instances of (Constr). Indeed if $x \in (Ab_B \sqcap \neg Ab_{A_1} \sqcap \cdots \sqcap \neg Ab_{A_n})^{I'}$, then $x \in (\neg\Box\neg B \sqcap \Box\neg A_1 \sqcap \cdots \sqcap \Box\neg A_n)^I$. It follows from the semantics of $\mathcal{ALC} + \mathbf{T}_{min}$ that there is a $y < x$ such that $y \in (B \sqcap \neg A_1 \sqcap \cdots \sqcap \neg A_n \sqcap \Box\neg B \sqcap \Box\neg A_1 \sqcap \cdots \sqcap \Box\neg A_n)^I$. By definition of I' (i) $y \in (B \sqcap \neg A_1 \sqcap \cdots \sqcap \neg Ab_B \sqcap \neg Ab_{A_1} \sqcap \cdots \sqcap \neg Ab_{An})^{I'}$, hence by (ii) $(x, y) \in R^{I'}_{B, A_1, \ldots, A_n}$, and $x \in (\exists R_{B, A_1, \ldots, A_n}(B \sqcap \neg A_1 \sqcap \cdots \sqcap \neg Ab_B \sqcap \neg Ab_{A_1} \sqcap \cdots \sqcap \neg Ab_{An}))^I$. Last by what said above $\neg Ab_A{}^{I'} = \Box\neg A^I$, hence $(x, \neg\Box\neg A) \in \mathcal{M}_{\mathcal{ALC}+\mathbf{T}_{min}}{}^{\Box-}$ iff $(x, Ab_A) \in \mathcal{M}_{CIRC}{}^{Ab}$.

$\Leftarrow$ Let $\mathcal{M}_{CIRC} = \langle \Delta, I \rangle$ of KB^*_{CIRC}. We build an $\mathcal{ALC} + \mathbf{T}_{min}$ model $\mathcal{M}_{\mathcal{ALC}+\mathbf{T}_{min}} = \langle \Delta, I', < \rangle$ of $KB_{\mathcal{ALC}} + \mathbf{T}_{min}$ as follows. The domain Δ of the two models coincides. We say that $y < x$ if:

(a) either for all Ab_A in M we have $x \in Ab_A{}^I$, whereas there is an Ab_A in M s.t. $y \in (A \sqcap \neg Ab_A)^I$;
(b) or there is an Ab_A in M s.t. $x \in \neg Ab_A{}^I$ and for all Ab_A s.t. $x \in \neg Ab_A{}^I$: $y \notin A^I$ and there exists B in M s.t. $A^I \subseteq B^I$ and $y \in (B \sqcap \neg Ab_B)^I$.

The two cases (a) and (b) above are mutually exclusive. I' is defined as I of $\mathcal{M}_{CIRC}$ (except from the predicates Ab_A that do not belong to the language of $\mathcal{ALC} + \mathbf{T}_{min}$).

We now show that $\mathcal{M}_{\mathcal{ALC}+\mathbf{T}_{min}}$ is a model of $KB_{\mathcal{ALC}+\mathbf{T}_{min}}$.

1. $<$ is irreflexive and transitive. Both properties can be easily proven.
2. $x \in \Box\neg A^{I'}$ iff $x \in \neg Ab_A^I$. This results from 2a and 2b below.
2a. $\Leftarrow$ Let $x \in \neg Ab_A^I$. Consider $y < x$. Clearly $y < x$ has been inserted by (b). Hence $y \notin A^I$ and by definition of I', $y \notin A^{I'}$, and $x \in \Box\neg A^{I'}$.
2b. $\Rightarrow$ If $x \in \Box\neg A^{I'}$. For a contradiction suppose $x \in Ab_A^I$. We distinguish two cases: case (a) of the definition of $<$. Consider $y \in A \sqcap \neg Ab_A^I$. This y exists by constraint (Constr). By (a) $y < x$ and $y \in A^I$, i.e. by definition of I', $y \in A^{I'}$, which contradicts $x \in \Box\neg A^{I'}$, therefore impossible. (b) There are $B_1 \ldots B_n$ s.t. $x \in (\neg Ab_{B_1} \sqcap \cdots \sqcap \neg Ab_{B_n})^I$. Since we are reasoning under the assumption that $x \in Ab_A^I$, by the constraint (Constr) there is a $y \in (A \sqcap \neg B_1 \sqcap \cdots \sqcap \neg B_n \sqcap \neg Ab_A \sqcap \neg Ab_{B_1} \sqcap \cdots \sqcap \neg Ab_{Bn})^I$. By (b) $y < x$. Furthermore, $y \in A^I$, and by definition of I' $y \in A^{I'}$ which contradicts that $x \in \Box\neg A^{I'}$. Both in case (a) and in case (b) the assumption that $x \in Ab_A^I$ leads to contradiction. We therefore conclude that $x \in \neg Ab_A^I$.
3. $<$ satisfies the Smoothness Condition. Consider $x \in A^{I'}$. If $x \in \neg Ab_A^I$, by 2 $x \in (\Box\neg A)^{I'}$, hence $Min_<(A^{I'}) \neq \emptyset$. On the other hand, if $x \in Ab_A^I$, by (Constr) there is a $y \in (A \sqcap \neg Ab_A \sqcap \neg Ab_{B_1} \sqcap \cdots \sqcap \neg Ab_{Bn})^I$ for $B_1 \cdots B_n$: $x \in \neg Ab_{B_1}{}^I \cdots \neg Ab_{B_n}{}^I$. By definition of I', $y \in A^{I'}$, and by 2a $y \in (\Box\neg A)^{I'}$ and $y \in Min_<(A^{I'})$. Therefore $Min_<(A^{I'}) \neq \emptyset$.
4. By definition of I' and by 2 we have that for each $\mathcal{ALC} + \mathbf{T}_{min}$ formula F and its translation F', $x \in F^{I'}$ in $\mathcal{M}_{\mathcal{ALC}+\mathbf{T}_{min}}$ iff $x \in F'^I$ in $\mathcal{M}_{CIRC}$. Since $\mathcal{M}_{CIRC}$ is a model of KB_{CIRC} which is obtained by substituting each formula in $KB_{\mathcal{ALC}} + \mathbf{T}_{min}$ with its translation it follows that $\mathcal{M}_{\mathcal{ALC}+\mathbf{T}_{min}}$ is a model of the corresponding $KB_{\mathcal{ALC}} + \mathbf{T}_{min}$.
5. Last, by 2 we immediately conclude that $(x, \neg\Box\neg A) \in \mathcal{M}_{\mathcal{ALC}+\mathbf{T}_{min}}{}^{\Box-}$ iff $(x, Ab_A) \in \mathcal{M}_{CIRC}{}^{Ab}$. $\quad\Box$

Lemma 3. *Consider any model $\mathcal{M}_{CIRC}$ and its equivalent $\mathcal{M}_{\mathcal{ALC}+\mathbf{T}_{min}}$ of the previous lemma. $\mathcal{M}_{CIRC}$ is a minimal model of KB^*_{CIRC} iff $\mathcal{M}_{\mathcal{ALC}+\mathbf{T}_{min}}$ is a minimal model of $KB_{\mathcal{ALC}+\mathbf{T}_{min}}$.*

Proof. $\Rightarrow$ If $\mathcal{M}_{CIRC}$ is a minimal model of KB^*_{CIRC}, suppose for a contradiction that $\mathcal{M}_{\mathcal{ALC}+\mathbf{T}_{min}}$ is not minimal. Then there is an $\mathcal{M}'_{\mathcal{ALC}}+\mathbf{T}_{min}$ such that $\mathcal{M}'_{\mathcal{ALC}}+\mathbf{T}_{min}$ is preferred to $\mathcal{M}_{\mathcal{ALC}+\mathbf{T}_{min}}$, i.e., it has the same domain however $\mathcal{M}'_{\mathcal{ALC}}+\mathbf{T}_{min}{}^{\Box-} \subset \mathcal{M}_{\mathcal{ALC}+\mathbf{T}_{min}}{}^{\Box-}$. By Lemma 2 above, from $\mathcal{M}'_{\mathcal{ALC}}+\mathbf{T}_{min}$ we can build a model $\mathcal{M}'_{CIRC}$ that has the same domain as $\mathcal{M}_{CIRC}$ but such that $\mathcal{M}'_{CIRC}{}^{Ab} \subset \mathcal{M}_{CIRC}{}^{Ab}$, against the minimality of $\mathcal{M}_{CIRC}$. We are therefore forced to conclude that also $\mathcal{M}_{\mathcal{ALC}+\mathbf{T}_{min}}$ must be a minimal model of $KB_{\mathcal{ALC}}+\mathbf{T}_{min}$.

$\Leftarrow$ If $\mathcal{M}_{\mathcal{ALC}+\mathbf{T}_{min}}$ is a minimal model of $KB_{\mathcal{ALC}}+\mathbf{T}_{min}$ we reason as above to conclude that also $\mathcal{M}_{CIRC}$ must be a minimal model of KB^*_{CIRC}. $\quad\Box$

Theorem 3. *Given a query F, and its translation F′, $KB_{\mathcal{ALC}+\mathbf{T}_{min}} \models^{\mathcal{L}_\mathbf{T}}_{min} F$ iff $KB^*_{CIRC} \models^{CP}_{CIRC} F'$.*

Proof. $\Rightarrow$ By contraposition, let $KB^*_{CIRC} \not\models^{CP}_{CIRC} F'$. Then there is a minimal model $\mathcal{M}_{CIRC} = \langle \Delta, I \rangle$ of KB^*_{CIRC} and an x such that $x \notin F'^I$ in $\mathcal{M}_{CIRC}$. By Lemmas 2 and 3 above there is a minimal $\mathcal{ALC}+\mathbf{T}_{min} = \langle \Delta, I', < \rangle$ model of $KB_{\mathcal{ALC}+\mathbf{T}_{min}}$ such that $x \notin F^{I'}$. Hence $KB_{\mathcal{ALC}+\mathbf{T}_{min}} \not\models^{\mathcal{L}_\mathbf{T}}_{min} F$. $\Leftarrow$ By reasoning as just done we can prove the other direction: if $KB_{\mathcal{ALC}+\mathbf{T}_{min}} \not\models^{\mathcal{L}_\mathbf{T}}_{min} F$ then $KB^*_{CIRC} \not\models^{CP}_{CIRC} F'$. $\quad\Box$

Observe that the mapping above does not provide a polynomial encoding of $\mathcal{ALC}+\mathbf{T}_{min}$ KBs into circumscribed KBs. In fact, the number of instances of (Constr) added in KB^*_{CIRC} is exponential in the number of atomic concepts in $\mathcal{L}_\mathbf{T}$. Finding a polynomial encoding is at present an open problem and it will be addressed in future research.

4.4. A mapping between circumscribed DLs and $\mathcal{ALC}+\mathbf{T}_{min}$ with nominals

In [10] Bonatti, Lutz and Wolter prove that, in $\mathcal{ALC}$, satisfiability with respect to concept-circumscribed KBs is NExp^{NP}-hard even if the TBox is acyclic, the ABox and preference relations are empty, and there are no fixed predicates.

In this section we show that the same hardness result can be proved for $\mathcal{ALC}+\mathbf{T}_{min}$, with *nominals*. In the following we call $\mathcal{ALCO}+\mathbf{T}_{min}$ the logic $\mathcal{ALC}+\mathbf{T}_{min}$ with nominals, and we provide a polynomial reduction of satisfiability in concept circumscribed KBs to satisfiability in $\mathcal{ALCO}+\mathbf{T}_{min}$.

Unfortunately, this result does not provide a lower bound for $\mathcal{ALC}+\mathbf{T}_{min}$, since nominals cannot be defined in $\mathcal{ALC}+\mathbf{T}_{min}$. These issues will be discussed at the end of this section, after the proof of the following Theorem 4.

Theorem 4. *Satisfiability in $\mathcal{ALCO}+\mathbf{T}_{min}$ is NExp^{NP}-hard.*

In oder to prove the theorem, we define a polynomial reduction of satisfiability in concept-circumscribed KBs to satisfiability in $\mathcal{ALCO}+\mathbf{T}_{min}$. The theorem is an immediate consequence of the lemmas below.

Let us consider an $\mathcal{ALC}$ concept-circumscribed KB $Circ_{CP}(T, A)$, where T is a TBox, $A = \emptyset$ an empty ABox, and where $CP = (\prec, M, F, V)$ is a circumscription pattern, with $\prec = \emptyset$ (the preference relations are empty), M, F and V respectively the minimized, fixed and varying predicates. We assume $F = \emptyset$. As in [10] (Theorem 15 and Corollary 16) we assume that there is no bound on the number of minimized predicates.

Let C_0 be an $\mathcal{ALC}$ concept in the language of $Circ_{CP}(T, A)$. We construct an $\mathcal{ALCO}+\mathbf{T}_{min}$ knowledge base $KB' = (T', A')$, as follows. Let M_{KB} be the concept names $p \in M$ such that p occurs either in the KB or in C_0. We define the language of KB' as the language containing the atomic concepts $M_{KB} \cup V \cup \{D, G\}$ where D and G are two new concept names. Furthermore, we introduce in the language of KB' some new individual names: c_1 and c_2 and, for each proposition $p \in M_{KB}$, an individual name c_p. We let

$$\mathcal{L}_\mathbf{T} = \{p\colon p \in M_{KB}\} \cup \{G\}$$

To define KB', we translate each $\mathcal{ALC}$ concept C into C^* as follows:

- $C^* = C$, if C is a concept name;
- $C^* = \neg C_1^*$ if $C = \neg C_1$;
- $C^* = C_1^* \sqcap C_2^*$ if $C = C_1 \sqcap C_2$;
- $C^* = \exists R.(C_1^* \sqcap D)$, if $C = \exists R.C_1$;
- $C^* = \forall R.(C_1^* \sqcap D)$, if $C = \forall R.C_1$.

The TBox T' contains the following inclusions:

$$D \sqcap C_1^* \sqsubseteq C_2^*, \text{ for all the inclusions } C_1 \sqsubseteq C_2 \in T;$$

$$D \sqcap p \sqsubseteq \neg\mathbf{T}(p), \text{ for each } p \in M_{KB};$$

$$D \sqcap G \sqsubseteq \neg\mathbf{T}(G).$$

The ABox A' contains:

- the assertion $D(c_1)$;
- for each $p \in M$ occurring in T, the assertion

$$\big(\mathbf{T}(p) \sqcap \neg D \sqcap \neg p_1 \sqcap \cdots \sqcap \neg p_n\big)(c_p)$$

  where $p_1, \ldots, p_n$ are all the concept names in M_{KB}, different from p;
- the assertion

$$(\neg D \sqcap G \sqcap \neg p_1 \sqcap \cdots \sqcap \neg p_n)(c_2)$$

  where $p_1, \ldots, p_n$ are all the concept names in M_{KB}.

To state that the domain elements corresponding to the individuals $c_2, c_{p_1}, \ldots, c_{p_n}$ are the only $\neg D$-elements, we add in the TBox T' the inclusion:

$$\neg D \sqsubseteq \{c_2\} \sqcup \{c_{p_1}\} \sqcup \cdots \sqcup \{c_{p_n}\} \tag{1}$$

where $p_1, \ldots, p_n$ are all the concept names in M_{KB}.

Finally, the concept C_0' is defined as $C_0' = C_0^* \sqcap D$.

By the above construction of KB′, we immediately get:

Lemma 4. *The size of* KB′ *is polynomial in the size of* KB.

Let us consider an atomic concept $p \in M$, which occurs in the KB or in C_0. The idea is that minimizing p in $Circ_{CP}(T, A)$ corresponds to maximizing $\mathbf{T}(p)$ (and thus minimizing $\neg \Box \neg p$) in KB′ according to $\mathcal{ALCO} + \mathbf{T}_{min}$. The D elements of the domain in a model of KB′ define the domain of a model of $Circ_{CP}(T, A)$. If a D-element of the domain satisfies p, by the inclusion above, it also satisfies $\neg \mathbf{T}(p)$, and hence it satisfies $\neg \Box \neg p$. By minimizing $\neg \Box \neg p$ in KB′, we also minimize p. The introduction of an individual name c_p, for each minimized concept name p, guarantees that the model contains at least a $\mathbf{T}(p) \sqcap \neg D$ element in the domain that satisfies only p and no other concept names in M_{KB}.

For each $q \in M - M_{KB}$ (i.e., for each $q \in M$ such that q neither occurs in KB nor in C_0), we encode the atomic concept q with the concept $G \sqcap \neg \mathbf{T}(G)$. By minimizing $\neg \Box \neg G$, we globally minimize all the $q \in M - M_{KB}$.

The idea is that each model M of the KB′ in $\mathcal{ALCO} + \mathbf{T}_{min}$ corresponds to a model $\mathcal{I}$ of the circumscribed KB, where $\mathcal{I}$ is obtained from M by: taking the set of D-elements of M as the domain $\Delta^{\mathcal{I}}$ and defining the interpretation of concepts and roles as in M, except for the concept names $q \in M - M_{KB}$, whose interpretation in $\mathcal{I}$ is defined as the interpretation of G in M.

What we want to prove is that the minimal $\mathcal{ALCO} + \mathbf{T}_{min}$ models of KB′ correspond to models of $Circ_{CP}(T, A)$ in which the $p \in M$ are minimized. We prove the following lemma:

Lemma 5. *The concept C_0 is satisfiable in $Circ_{CP}(T, A)$ if and only if the concept C_0' is satisfiable with respect to* KB′ *in $\mathcal{ALCO} + \mathbf{T}_{min}$.*

Proof. For the "⇒" direction, let us assume that the concept C_0 is satisfiable in $Circ_{CP}(T, A)$, that is, there is a model $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ of $Circ_{CP}(T, A)$ such that $x_0 \in C_0^{\mathcal{I}}$, for some $x_0 \in \Delta$. We show that we can construct a model $\mathcal{M}' = \langle \Delta', <', I' \rangle$ of KB′ which is minimal w.r.t. $\mathcal{ALCO} + \mathbf{T}_{min}$ and is such that $x_0 \in \Delta'$ and $x_0 \in (C_0')^{I'}$. Δ' contains the domain elements in $\Delta^{\mathcal{I}}$ plus a new domain elements u_p, for each $p \in M$ such that p occurs in KB or in C_0, and a further element u, i.e.,

$$\Delta' = \Delta^{\mathcal{I}} \cup \{u_p \colon p \in M_{KB}\} \cup \{u\}$$

The idea is that, in the model $\mathcal{M}'$, each domain element u_p is defined as a typical p-element (for $p \in M_{KB}$), and that u is defined as a typical G element, which is a $\neg p$-element for all the $p \in M_{KB}$. We interpret individual constants as follows:

$$c_1^{I'} \in \Delta^{\mathcal{I}} \ (c_1^{I'} \text{ is any element of } \Delta^{\mathcal{I}})$$

$$c_p^{I'} = u_p, \text{ for all } p \in M_{KB}$$

$$c_2^{I'} = u$$

On the elements of $x \in \Delta^{\mathcal{I}}$ the interpretation I' is defined as follows:

- $x \in D^{I'}$;
- $x \in G^{I'}$ iff there is a $q \in M - M_{KB}$ such that $x \in q^{\mathcal{I}}$;
- $x \in p^{I'}$ iff $x \in p^{\mathcal{I}}$, for all concept names $p \in (M_{KB} \cup F \cup V)$;
- $(x, z) \in r^{I'}$ iff $(x, z) \in r^{\mathcal{I}}$, for all role names r, for all $x, z \in \Delta^{\mathcal{I}}$.

Let us define the interpretation I' on the new domain elements of Δ'. For all $u_p \in \Delta'$ (where $p \in M_{KB}$):

- $u_p \notin D^{I'}$;
- $u \notin G^{I'}$;
- $u_p \in p^{I'}$;
- $u_p \notin q^{I'}$, for all concept names $q \in M_{KB}$ with $q \neq p$;
- $(u_p, z) \notin r^{I'}$ and $(z, u_p) \notin r^{I'}$, for all role names r and for all $z \in \Delta'$.

For element $u \in \Delta'$:

- $u \notin D^{I'}$;
- $u \in G^{I'}$;
- $u \notin p^{I'}$, for all concept names $p \in M_{KB}$;
- $(u, z) \notin r^{I'}$ and $(z, u) \notin r^{I'}$, for all role names r and for all $z \in \Delta'$.

The relation $<'$ is defined as follows:

- for all $x \in \Delta^{\mathcal{I}}$ such that $x \in p^{\mathcal{I}}$ for some $p \in M_{KB}$, we let $u_p <' x$;
- for all $x \in \Delta^{\mathcal{I}}$ such that $x \in G^{\mathcal{I}}$, we let $u <' x$;
- no other pairs of elements of Δ' belong to the relation $<'$.

Observe that, for all $p \in M_{KB}$, $u_p \in (\mathbf{T}(p))^{I'}$. Moreover, for all $x \in \Delta^{\mathcal{I}}$, for all $p \in M_{KB}$:

$$x \in p^{\mathcal{I}} \quad \text{iff} \quad x \in (\neg\Box\neg p)^{I'}$$

$$x \notin p^{\mathcal{I}} \quad \text{iff} \quad x \in (\Box\neg p)^{I'}$$

and, for all $x \in \Delta^{\mathcal{I}}$, $x \in G^{I'}$ and $x \in (\neg\Box\neg G)^{I'}$ iff there is a $q \in M - M_{KB}$ such that $x \in q^{\mathcal{I}}$.

It can be shown that, for all $x \in \Delta^{\mathcal{I}}$ and for all concepts C on the language of $Circ_{CP}(T, A)$ which do not contain any atomic concept in $M - M_{KB}$:

$$x \in (C)^{\mathcal{I}} \quad \text{iff} \quad x \in \left(C^*\right)^{I'} \tag{2}$$

The proof is by induction on the structure of C. Observe that the atomic concepts in $M_{KB} \cup V$ have the same interpretation in $\mathcal{I}$ and in I' on the elements $x \in \Delta^{\mathcal{I}}$.

It is easy to see that $\mathcal{M}'$ is a model of KB$'$ (observe that, each inclusion $D \sqcap C_1^* \sqsubseteq C_2^*$ in KB$'$ is true for all D-elements, as well as (trivially) for $\neg D$-elements).

It can be proved that, $\mathcal{M}'$ is minimal model of KB$'$ w.r.t. $\mathcal{ALCO} + \mathbf{T}_{min}$. If it were not, there would be a model $\mathcal{M}''$ of KB$'$ preferred to $\mathcal{M}'$. Suppose that, for some $p \in M_{KB}$ and $x \in \Delta'$, $x \in (\Box\neg p)^{I''}$ while $x \in (\neg\Box\neg p)^{I'}$. By construction, it must be that $x \in \Delta^{\mathcal{I}}$ and hence $x \in D^{I'}$. Moreover, it must be that $x \in p^{\mathcal{I}}$ and $x \in p^{I'}$. As $x \in \Delta^{\mathcal{I}}$, it must be that $x \in D^{I''}$ (otherwise, by the inclusion (1), x would be in $\{u, u_{p_1}, \ldots, u_{p_n}\}$). Then, it must be $x \notin p^{I''}$ as, otherwise, we would have $x \in (\neg\Box\neg p)^{I''}$ by axiom $D \sqcap p \sqsubseteq \neg\mathbf{T}(p)$. From $\mathcal{M}''$ we are able to construct a new model $\mathcal{I}''$ of T and A (on the same domain $\Delta^{\mathcal{I}}$) such that $\mathcal{I}'' <_{CP} \mathcal{I}$, by taking the interpretation of the atomic concepts in $M_{KB} \cup V$ as in $\mathcal{M}''$ and the interpretation of the concepts in $M - M_{KB}$ as in $\mathcal{I}$. This contradicts the hypothesis that $\mathcal{I}$ is a model of $Circ_{CP}(T, A)$.

Suppose instead that the model $\mathcal{M}''$ of KB$'$ is preferred to $\mathcal{M}'$ since for some $x \in \Delta'$, $x \in \Box\neg G^{I''}$ while $x \in \neg\Box\neg G^{I'}$. By construction, it must be that $x \in D^{I'}$ and that there is at least a $q \in M - M_{KB}$ such that $x \in q^{\mathcal{I}}$. Then, we can construct a model $\mathcal{J}$ preferred to $\mathcal{I}$ by defining the interpretation in $\mathcal{J}$ as in $\mathcal{I}$ apart from taking $x \notin q^{\mathcal{J}}$. Again, this contradicts the hypothesis that $\mathcal{I}$ is a model of $Circ_{CP}(T, A)$. Hence, $\mathcal{M}'$ is a minimal model of KB$'$ w.r.t. $\mathcal{ALCO} + \mathbf{T}_{min}$.

Finally, as from the hypothesis $x_0 \in C_0^{\mathcal{I}}$, for some $x_0 \in \Delta^{\mathcal{I}}$, by (2), we have that $x_0 \in (C_0^*)^{I'}$ and, thus, $x_0 \in (C_0')^{I'}$. This concludes the proof of the $\Rightarrow$ direction of Lemma 5.

Let us prove the $\Leftarrow$ direction of Lemma 5. Assume that the concept C_0' is satisfiable in KB$'$ in $\mathcal{ALCO} + \mathbf{T}_{min}$, that is, there is a model $\mathcal{M}' = \langle \Delta', <', I' \rangle$ of KB$'$ which is a minimal model of KB$'$ in $\mathcal{ALCO} + \mathbf{T}_{min}$ and is such that $x_0 \in (C_0')^{I'}$, for some $x_0 \in \Delta'$. We show that we can construct a model $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ of $Circ_{CP}(T, A)$ such that $x_0 \in C_0^{\mathcal{I}}$.

We construct the model $\mathcal{M}$ as follows. $\Delta^{\mathcal{I}}$ is defined as the set of D-elements of Δ', i.e., $\Delta^{\mathcal{I}} = \Delta' \cap D^{I'}$. The interpretation $\cdot^{\mathcal{I}}$ on the domain $\Delta^{\mathcal{I}}$ is defined as I' on the language of $Circ_{CP}(T, A)$, except for the atomic concepts $q \in M - M_{KB}$. For all $x, y \in \Delta^{\mathcal{I}}$:

- $x \in p^{\mathcal{I}}$ iff $x \in p^{I'}$, for all concept names $p \in (M_{KB} \cup V)$;
- $x \in q^{\mathcal{I}}$ iff $x \in (G \sqcap \neg\mathbf{T}(G))^{I'}$, for all concept names $q \in M - M_{KB}$;
- $(x, y) \in r^{\mathcal{I}}$ iff $(x, y) \in r^{I'}$, for all role names r.

It can be shown that: For all $x \in \Delta$ and for all concepts C on the language of $Circ_{CP}(T, A)$, which do not contain concept names $q \in M - M_{KB}$:

$$x \in (C)^{\mathcal{I}} \quad \text{iff} \quad x \in \left(C^*\right)^{I'} \tag{3}$$

The proof can be done by induction on the structure of C.

For C concept name, it holds by construction. Observe that, if $x \in (C)^{\mathcal{I}}$, then it must be the case that $x \in C^{I'}$. Also, as $x \in \Delta^{\mathcal{I}}$, then $x \in D^{I'}$. Hence, $x \in (C^*)^{I'}$. Vice versa, if $x \in (C^*)^{I'}$, then $x \in C^{I'}$. By construction, $x \in (C)^{I}$.

For the inductive case, let us consider the case $C = \exists R.C_1$. If $x \in (\exists R.C_1)^{\mathcal{I}}$, then there is a $y \in \Delta^{\mathcal{I}}$ such that $(x, y) \in r^{\mathcal{I}}$ and $y \in C_1^{\mathcal{I}}$. We can show that $x \in ((\exists R.C_1)^*)^{I'}$. As $y \in \Delta^{\mathcal{I}}$, by construction, $y \in D^{I'}$. As $(x, y) \in r^{\mathcal{I}}$, by construction $(x, y) \in r^{I'}$. From $y \in C_1^{\mathcal{I}}$, we get, by inductive hypothesis, $y \in (C_1^*)^{I'}$. Hence, $y \in (C_1^* \sqcap D)^{I'}$. Thus, $x \in (\exists R.(C_1^* \sqcap D))^{I'}$, i.e., $x \in ((\exists R.C_1)^*)^{I'}$.

From the hypothesis, since $x_0 \in (C_0')^{I'}$, then $x_0 \in C_0^{I'}$, and $x_0 \in D^{I'}$. Hence, by construction, $x_0 \in \Delta^{\mathcal{I}}$ and, by (3), $x_0 \in C_0^{\mathcal{I}}$.

$\mathcal{I}$ is a model of T and A (remember that A is empty). In fact, it can be shown that $\mathcal{I}$ satisfies all the inclusions in T. Let $C_1 \sqsubseteq C_2 \in T$. Then, there is an inclusion $D \sqcap C_1^* \sqsubseteq C_2^* \in T'$. Assume that $x \in C_1^{\mathcal{I}}$. Then, by (3), we have $x \in (C_1^*)^{I'}$. Since $x \in \Delta^{\mathcal{I}}$, by construction, $x \in D^{I'}$. By the inclusion in T', $x \in (C_2^*)^{I'}$. Again, by (3), we have $x \in C_2^{\mathcal{I}}$.

It is easy to see that there is no model $\mathcal{J}$ of T and A, such that $\mathcal{J}$ is preferred to $\mathcal{I}$. If there were a model $\mathcal{J} = \langle \Delta^{\mathcal{J}}, \cdot^{\mathcal{J}} \rangle$ of T and A (where $\Delta^{\mathcal{I}} = \Delta^{\mathcal{J}}$) preferred to $\mathcal{I}$, we would have that: for all $q \in M$, $q^{\mathcal{J}} \subseteq q^{\mathcal{I}}$ and for some $p \in M$, $p^{\mathcal{J}} \subset p^{\mathcal{I}}$. In such a case, it would be possible to construct a model $\mathcal{M}'' = \langle \Delta'', <'', I'' \rangle$ of KB$'$ preferred to $\mathcal{M}'$. $\mathcal{M}''$ would have the same domain $\Delta'' = \Delta'$ as in $\mathcal{M}'$, the same interpretation of concepts and roles on the elements of $\Delta^{\mathcal{I}}$ (the D-elements) as in $\mathcal{J}$, and the same interpretation of concepts and roles as in $\mathcal{M}'$ for the elements of $\Delta' - \Delta^{\mathcal{I}}$ (the $\neg D$-elements). For all $x, y \in \Delta^{\mathcal{I}}$:

- $x \in p^{I''}$ iff $x \in p^{\mathcal{J}}$, for all concept names $p \in (M_{KB} \cup V)$;
- $x \in G^{I''}$ iff, for some concept name $q \in M - M_{KB}$, $x \in q^{\mathcal{J}}$;
- $(x, y) \in r^{I''}$ iff $(x, y) \in r^{\mathcal{J}}$, for all role names r.

Observe that, for each $p \in M_{KB}$, given the assertions in A', there must be an element of the domain $\Delta''(= \Delta')$, let us call it u_p, which is a p-element and is a $\neg p_i$-element, for all $p_i \in M_{KB}$ such that $p_i \neq p$. Also, there must be an element of the domain Δ'', let us call it u, which is a G-element and is a $\neg p$-element, for all $p \in M_{KB}$. The relation $<''$ is defined as follows. For all $x \in \Delta^{\mathcal{I}}$:

for all $p \in M_{KB}$, $u_p <'' x$ iff $x \in p^{\mathcal{J}}$
$u <'' x$ iff, for some $q \in M - M_{KB}$, $x \in q^{\mathcal{J}}$

No other pairs of elements of Δ'' are in $<''$.

As there is at least a $p \in M$ such that $p^{\mathcal{J}} \subset p^{\mathcal{I}}$ and, for all $q \in M$, $q^{\mathcal{J}} \subseteq q^{\mathcal{I}}$, $\mathcal{M}''$ would then be preferred to $\mathcal{M}'$, against the assumption that $\mathcal{M}'$ is minimal. Hence, $\mathcal{I}$ must be a model of $Circ_{CP}(T, A)$. □

Observe that the use of nominals in the proof above prevents that, in the minimization of $\neg\square$ formulas, some D-element becomes a $\neg D$-element. By inclusion (1) a model of KB$'$ must contain at most $n + 1$ $\neg D$-elements, and, by the assertions in A', it must contain exactly $n + 1$ $\neg D$-elements.

The same reduction does not work for $\mathcal{ALC} + \mathbf{T}_{min}$. One may hope that nominals can be modeled to some extent in $\mathcal{ALC} + \mathbf{T}_{min}$ by making use of minimization. In particular, to define the nominal $\{o\}$, let us introduce an atomic concept L representing it. Let us also introduce the auxiliary atomic concepts L' and F and the role R and let the KB contain the following inclusions:

$L \sqsubseteq L'$

$\mathbf{T}(L) \sqsubseteq F$

$\mathbf{T}(L') \sqsubseteq \neg F$

Let the KB contain the assertion $L(o)$. In the minimal $\mathcal{ALC} + \mathbf{T}_{min}$ models $\mathcal{M} = \langle \Delta, <, I \rangle$ of the KB, there is a single individual satisfying L, namely o^I. Observe, however, that the model $\mathcal{M}$ must contain an individual u, such that $u < o^I$ and u is a typical L'-element. This makes the solution above not general enough for defining nominals. Suppose we want to define a KB in which exactly one individual is a C and exactly one individual is a $\neg C$. Using nominals we can simply write:

$\{o_1\} \sqsubseteq C$

$C \sqsubseteq \{o_1\}$

$\{o_2\} \sqsubseteq \neg C$

$\neg C \sqsubseteq \{o_2\}$

However, with the construction above, we would introduce two atomic concepts L_{o_1} and L_{o_2} to represent the nominals $\{o_1\}$ and $\{o_2\}$. The models of the resulting KB should contain also a typical L'_{o_1} element and a typical L'_{o_2} element. But, are these C or $\neg C$ elements?

For a similar reason, in the proof of Theorem 4, we are unable to encode inclusion (1), i.e. $\neg D \sqsubseteq \{c_2\} \sqcup \{c_{p_1}\} \sqcup \cdots \sqcup \{c_{p_n}\}$. In order to state that $\{c_2^I, c_{p_1}^I, \ldots, c_{p_n}^I\}$ are the only $\neg D$-elements, a new element u, preferred to all of them, must be introduced in any $\mathcal{ALC} + \mathbf{T}_{min}$ model. But should u be a D or a $\neg D$ element?

In the definition of $\mathcal{ALC} + \mathbf{T}_{min}$ in Section 3, we have assumed that the typical instances of the concepts in $\mathcal{L}_\mathbf{T}$ are maximized, essentially, by minimizing the instances of $\neg\square\neg C$, for all $C \in \mathcal{L}_\mathbf{T}$. All the other predicates are allowed to vary. Similarly to what is done for circumscribed KBs [10], we could consider the case where the interpretation of some predicates is kept fixed during the minimization.

Observe that, when fixed predicates are allowed a polynomial reduction of satisfiability in concept circumscribed KBs to satisfiability in $\mathcal{ALC} + \mathbf{T}_{min}$ can be defined. In essence, we can adapt the proof of Theorem 4, by requiring that the atomic concept D is a fixed predicate, so that its interpretation remains unaltered during minimization. This prevents that, in the minimization of $\neg\square$ formulas, some D-element becomes a $\neg D$-element.

Unfortunately, this result again does not provide a lower bound for $\mathcal{ALC} + \mathbf{T}_{min}$ without fixed predicates. In fact, while in circumscribed KBs, the fixed predicates can be eliminated by introducing new predicates to be minimized and new inclusions in the TBox, a similar construction cannot be used to eliminate fixed predicates in $\mathcal{ALC} + \mathbf{T}_{min}$.

In [10] the idea was that, to fix an atomic predicate p, we both minimize p and its complement $\neg p$. In $\mathcal{ALC} + \mathbf{T}_{min}$, fixing an atomic concept p requires not only the introduction of new symbols in the language but also of new preferred elements in the semantics. The problem is that, the preferred elements must be either p or $\neg p$ elements, and this becomes incompatible with the finite chain property of the semantics.

5. A tableaux calculus for $\mathcal{ALC} + \mathbf{T}_{min}$

In this section we present a tableau calculus for deciding whether a query F is minimally entailed by a knowledge base (TBox, ABox). We introduce a labeled tableau calculus called $\mathcal{TAB}_{min}^{\mathcal{ALC}+\mathbf{T}}$, which extends the calculus $T^{\mathcal{ALC}+\mathbf{T}}$ presented in [26], and allows to reason about minimal models.

$\mathcal{TAB}_{min}^{\mathcal{ALC}+\mathbf{T}}$ performs a two-phase computation in order to check whether a query F is minimally entailed from the initial KB. In particular, the procedure tries to build an open branch representing a minimal model satisfying KB $\cup \{\neg F\}$.

In the first phase, a tableau calculus, called $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$, simply verifies whether KB $\cup \{\neg F\}$ is satisfiable in an $\mathcal{ALC} + \mathbf{T}$ model, building candidate models. In the second phase another tableau calculus, called $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$, checks whether the candidate models found in the first phase are *minimal* models of KB. To this purpose for each open branch of the first phase, $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$ tries to build a "smaller" model of KB, i.e. a model whose individuals satisfy less formulas $\neg\square\neg C$ than the corresponding candidate model. The whole procedure $\mathcal{TAB}_{min}^{\mathcal{ALC}+\mathbf{T}}$ is formally defined at the end of this section (Definition 22).

$\mathcal{TAB}_{min}^{\mathcal{ALC}+\mathbf{T}}$ is based on the notion of a *constraint system*. We consider a set of *variables* drawn from a denumerable set $\mathcal{V}$. Variables are used to represent individuals not explicitly mentioned in the ABox, that is to say implicitly expressed by existential as well as universal restrictions.

$\mathcal{TAB}_{min}^{\mathcal{ALC}+\mathbf{T}}$ makes use of labels, which are denoted with $x, y, z, \ldots$. A label represents either a variable or an individual constant occurring in the ABox, that is to say an element of $\mathcal{O} \cup \mathcal{V}$.

Definition 10 (*Constraint*). A *constraint* (or *labeled* formula) is a syntactic entity of the form either $x \xrightarrow{R} y$ or $y < x$ or $x : C$, where x, y are labels, R is a role and C is either an extended concept or has the form $\square\neg D$ or $\neg\square\neg D$, where D is a concept.

Intuitively, a constraint of the form $x \xrightarrow{R} y$ says that the individual represented by label x is related to the one denoted by y by means of role R; a constraint $y < x$ says that the individual denoted by y is "preferred" to the individual represented by x with respect to the relation $<$; a constraint $x : C$ says that the individual denoted by x is an instance of the concept C, i.e. it belongs to the extension C^I. As we will define in Definition 13, the ABox of a knowledge base can be translated into a set of constraints by replacing every membership assertion $C(a)$ with the constraint $a : C$ and every role aRb with the constraint $a \xrightarrow{R} b$.

Let us now separately analyze the two components of the calculus $\mathcal{TAB}_{min}^{\mathcal{ALC}+\mathbf{T}}$, starting with $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$.

5.1. The tableau calculus $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$

Let us first define the basic notions of a tableau system in $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$:

Definition 11 (*Tableau of $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$*). A tableau of $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ is a tree whose nodes are constraint systems, i.e., pairs $\langle S | U \rangle$, where S is a set of constraints, whereas U contains formulas of the form $C \sqsubseteq D^L$, representing subsumption relations

$\langle S, x : C, x : \neg C \mid U \rangle$ (Clash)

$\langle S, x : \neg\top \mid U \rangle$ (Clash)$_\top$

$\langle S, x : \bot \mid U \rangle$ (Clash)$_\bot$

$$\frac{\langle S, x : C \sqcap D \mid U \rangle}{\langle S, x : C \sqcap D, x : C, x : D \mid U \rangle} \ (\sqcap^+) \quad \text{if } \{x:C, x:D\} \not\subseteq S$$

$$\frac{\langle S, x : \neg(C \sqcap D) \mid U \rangle}{\langle S, x : \neg(C \sqcap D), x : \neg C \mid U \rangle \quad \langle S, x : \neg(C \sqcap D), x : \neg D \mid U \rangle} \ (\sqcap^-) \quad \text{if } x : \neg C \notin S \text{ and } x : \neg D \notin S$$

$$\frac{\langle S, x : C \sqcup D \mid U \rangle}{\langle S, x : C \sqcup D, x : C \mid U \rangle \quad \langle S, x : C \sqcup D, x : D \mid U \rangle} \ (\sqcup^+) \quad \text{if } x : C \notin S \text{ and } x : D \notin S$$

$$\frac{\langle S, x : \neg(C \sqcup D) \mid U \rangle}{\langle S, x : \neg(C \sqcup D), x : \neg C, x : \neg D \mid U \rangle} \ (\sqcup^-) \quad \text{if } \{x:\neg C, x:\neg D\} \not\subseteq S$$

$$\frac{\langle S, x : \neg\neg C \mid U \rangle}{\langle S, x : \neg\neg C, x : C \mid U \rangle} \ (\neg) \quad \text{if } x : C \notin S$$

$$\frac{\langle S, x : \mathbf{T}(C) \mid U \rangle}{\langle S, x : \mathbf{T}(C), x : C, x : \Box\neg C \mid U \rangle} \ (\mathbf{T}^+) \quad \text{if } \{x:C, x:\Box\neg C\} \not\subseteq S$$

$$\frac{\langle S, x : \neg\mathbf{T}(C) \mid U \rangle}{\langle S, x : \neg\mathbf{T}(C), x : \neg C \mid U \rangle \quad \langle S, x : \neg\mathbf{T}(C), x : \neg\Box\neg C \mid U \rangle} \ (\mathbf{T}^-) \quad \text{if } x : \neg C \notin S \text{ and } x : \neg\Box\neg C \notin S$$

$$\frac{\langle S \mid U \rangle}{\langle S, x : \Box\neg C \mid U \rangle \quad \langle S, x : \neg\Box\neg C \mid U \rangle} \ (cut) \quad \begin{array}{l} \text{if } x : \neg\Box\neg C \notin S \text{ and } x : \Box\neg C \notin S \\ C \in \mathcal{L}_\mathbf{T} \\ x \text{ occurs in } S \end{array}$$

$$\frac{\langle S \mid U, C \sqsubseteq D^L \rangle}{\langle S, x : \neg C \sqcup D \mid U, C \sqsubseteq D^{L,x} \rangle} \ (\sqsubseteq) \quad \text{if } x \text{ occurs in } S \text{ and } x \notin L$$

$$\frac{\langle S, x : \forall R.C, x \xrightarrow{R} y \mid U \rangle}{\langle S, x : \forall R.C, x \xrightarrow{R} y, y : C \mid U \rangle} \ (\forall^+) \quad \text{if } y : C \notin S$$

$$\frac{\langle S, x : \exists R.C \mid U \rangle}{\langle S, x : \exists R.C, x \xrightarrow{R} y, y : C \mid U \rangle \ \ \langle S, x : \exists R.C, x \xrightarrow{R} v_1, v_1 : C \mid U \rangle \ \ \langle S, x : \exists R.C, x \xrightarrow{R} v_2, v_2 : C \mid U \rangle \cdots \langle S, x : \exists R.C, x \xrightarrow{R} v_n, v_n : C \mid U \rangle} \ (\exists^+)$$

y new, if $\nexists z \prec x$ s.t. $z \equiv_{S, x:\exists R.C} x$ and $\nexists u$ s.t. $x \xrightarrow{R} u \in S$ and $u : C \in S$, $\forall v_i$ occurring in S

$$\frac{\langle S, x : \neg\Box\neg C \mid U \rangle}{\langle S, x : \neg\Box\neg C, y < x, y : C, y : \Box\neg C, S^M_{x \to y} \mid U \rangle \ \ \langle S, x : \neg\Box\neg C, v_1 < x, v_1 : C, v_1 : \Box\neg C, S^M_{x \to v_1} \mid U \rangle \cdots \langle S, x : \neg\Box\neg C, v_n < x, v_n : C, v_n : \Box\neg C, S^M_{x \to v_n} \mid U \rangle} \ (\Box^-)$$

y new, if $\nexists z \prec x$ s.t. $z \equiv_{S, x:\neg\Box\neg C} x$ and $\nexists u$ s.t. $\{u < x, u : C, u : \Box\neg C, S^M_{x \to u}\} \subseteq S$, $\forall v_i$ occurring in $S, x \neq v_i$

Fig. 1. The calculus $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$. To save space, we omit the rules $(\forall^-)$ and $(\exists^-)$, dual to $(\exists^+)$ and $(\forall^+)$, respectively.

$C \sqsubseteq D$ of the TBox. L is a list of labels.[7] A branch is a sequence of nodes $\langle S_1|U_1 \rangle, \langle S_2|U_2 \rangle, \ldots, \langle S_n|U_n \rangle, \ldots$, where each node $\langle S_i|U_i \rangle$ is obtained from its immediate predecessor $\langle S_{i-1}|U_{i-1} \rangle$ by applying a rule of $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$ (see Fig. 1), having $\langle S_{i-1}|U_{i-1} \rangle$ as the premise and $\langle S_i|U_i \rangle$ as one of its conclusions. A branch is closed if one of its nodes is an instance of *clash* (either (Clash) or (Clash)$_\top$ or (Clash)$_\bot$), otherwise it is open. A tableau is closed if all its branches are closed.

In the following, we will often refer to the *height* of a tableau: intuitively, the height of a tableau corresponds to the height of the tree of Definition 11. This is formally stated as follows:

Definition 12 (*Height of a tableau*). Given a tableau of $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$ having $\langle S|U \rangle$ as a root, we define its height h as follows:

- $h = 0$ if no rule is applied to $\langle S|U \rangle$[8];
- $h = 1 + max\{h_1, h_2, \ldots, h_n\}$ if a rule (R) ia applied to $\langle S|U \rangle$ and $h_1, h_2, \ldots, h_n$ are the heights of the tableaux whose roots are the conclusions of (R).

In order to check the satisfiability of a KB, we build the corresponding constraint system $\langle S|U \rangle$, and we check its satisfiability.

---

[7] As we will discuss later, this list is used in order to ensure the termination of the tableau calculus.

[8] In case of a closed tableau, this corresponds to the case in which $\langle S|U \rangle$ is an instance either of (Clash) or of (Clash)$_\bot$ or of (Clash)$_\top$.

Definition 13 *(Corresponding constraint system).* Given a knowledge base KB = (TBox, ABox), we define its *corresponding constraint system* $\langle S|U \rangle$ as follows:

- $S = \{a : C \mid C(a) \in \text{ABox}\} \cup \{a \xrightarrow{R} b \mid aRb \in \text{ABox}\}$;
- $U = \{C \sqsubseteq D^{\emptyset} \mid C \sqsubseteq D \in \text{TBox}\}$.

Definition 14 *(Model satisfying a constraint system).* Let $\mathcal{M} = \langle \Delta, I, < \rangle$ be a model as defined in Definition 3. We define a function α which assigns to each variable of $\mathcal{V}$ an element of Δ, and assigns every individual constant $a \in \mathcal{O}$ to $a^I \in \Delta$. $\mathcal{M}$ satisfies a constraint F under α, written $\mathcal{M} \models_{\alpha} F$, as follows:

- $\mathcal{M} \models_{\alpha} x : C$ if and only if $\alpha(x) \in C^I$;
- $\mathcal{M} \models_{\alpha} x \xrightarrow{R} y$ if and only if $(\alpha(x), \alpha(y)) \in R^I$;
- $\mathcal{M} \models_{\alpha} y < x$ if and only if $\alpha(y) < \alpha(x)$.

A constraint system $\langle S|U \rangle$ is satisfiable if there is a model $\mathcal{M}$ and a function α such that $\mathcal{M}$ satisfies every constraint in S under α and that, for all $C \sqsubseteq D^L \in U$ and for all $x \in \Delta$, we have that if $x \in C^I$ then $x \in D^I$.

Let us now show that:

Proposition 4. KB = (TBox, ABox) *is satisfiable in an* $\mathcal{ALC} + \mathbf{T}$ *model if and only if its corresponding constraint system* $\langle S|U \rangle$ *is satisfiable in the same model.*

Proof. We show that a model $\mathcal{M}$ as in Definition 3 satisfies KB if and only if there is a function α such that $\langle S|U \rangle$ is satisfiable in $\mathcal{M}$ under α. We simply define α as follows: α assigns each individual constant $a \in \mathcal{O}$ to $a^I \in \Delta$. Let us first consider the ABox and each formula belonging to it. By Definition 4, given $C(a) \in \text{ABox}$, we have that $\mathcal{M} \models C(a)$ iff $a^I \in C^I$. By Definition 13 of the corresponding constraint system, we have that $a : C \in S$; since a is an individual constant occurring in the ABox, we have that $\alpha(a) = a^I$, thus $a^I \in C^I$ iff $\alpha(a) \in C^I$ and, by Definition 14, iff $\mathcal{M} \models_{\alpha} a : C$. In case $\mathcal{M} \models aRb$, we have that $a \xrightarrow{R} b \in S$. $\mathcal{M} \models aRb$ iff $(a^I, b^I) \in R^I$ iff $(\alpha(a), \alpha(b)) \in R^I$ iff $\mathcal{M} \models_{\alpha} a \xrightarrow{R} b$. Concerning the TBox, $\mathcal{M} \models C \sqsubseteq D$ iff, for each $x \in \Delta$, if $x \in C^I$ then $x \in D^I$, i.e. $\mathcal{M} \models C \sqsubseteq D^{\emptyset}$. □

To verify the satisfiability of KB $\cup \{\neg F\}$, we use $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ to check the satisfiability of the constraint system $\langle S|U \rangle$ obtained by adding the constraint corresponding to $\neg F$ to S', where $\langle S'|U \rangle$ is the corresponding constraint system of KB. To this purpose, the rules of the calculus $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ are applied until either a contradiction is generated (*clash*) or a model satisfying $\langle S|U \rangle$ can be obtained from the resulting constraint system. As in the calculus proposed in [26], given a node $\langle S|U \rangle$, for each subsumption $C \sqsubseteq D^L \in U$ and for each label x that appears in the tableau, we add to S the constraint $x : \neg C \sqcup D$: we refer to this mechanism as *subsumption expansion*. As mentioned above, each subsumption $C \sqsubseteq D$ is equipped with a list L of labels in which the subsumption has been expanded in the current branch. This is needed to avoid multiple expansions of the same subsumption by using the same label, generating infinite branches.

Before introducing the rules of $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ we need some more definitions. First, as in [13], we define an ordering relation $\prec$ to keep track of the temporal ordering of insertion of labels in the tableau, that is to say if y is introduced in the tableau, then $x \prec y$ for all labels x that are already in the tableau. Moreover, we need to define the *equivalence* between two labels: intuitively, two labels x and y are equivalent if they label the same set of extended concepts. This notion is stated in the following definition, and it is used in order to apply the blocking machinery described in the following, based on the fact that equivalent labels represent the same element in the model built by $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$.

Definition 15. Given a tableau node $\langle S|U \rangle$ and a label x, we define

$$\sigma\left(\langle S|U \rangle, x\right) = \{C \mid x : C \in S\}$$

Furthermore, we say that two labels x and y are *S-equivalent*, written $x \equiv_S y$, if they label the same set of concepts, i.e.

$$\sigma\left(\langle S|U \rangle, x\right) = \sigma\left(\langle S|U \rangle, y\right)$$

Last, we define the set of formulas $S_{x \rightarrow y}^M$, that will be used in the rule ($\square^-$) when $y < x$, in order to introduce $y : \neg C$ and $y : \square \neg C$ for each $x : \square \neg C$ in the current branch:

Definition 16. Given a tableau node $\langle S|U \rangle$ and two labels x and y, we define

$$S_{x \rightarrow y}^M = \{y : \neg C, y : \square \neg C \mid x : \square \neg C \in S\}$$

The rules of $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ are presented in Fig. 1. Rules ($\exists^+$) and ($\Box^-$) are called *dynamic* since they introduce a new variable in their conclusions. The other rules are called *static*. A brief explanation of the rules follows:

- (Clash), (Clash)$_\top$ and (Clash)$_\bot$ are used to detect *clashes*, i.e. unsatisfiable constraint systems.
- Rules for $\sqcup$, $\sqcap$, $\neg$, and $\forall$ are similar to the corresponding ones in the tableau calculus for standard $\mathcal{ALC}$ [13]: as an example, the rule ($\sqcup^+$) is applied to a constraint system of the form $\langle S, x : C \sqcup D | U\rangle$ in order to deal with the constraint $x : C \sqcup D$ introducing two branches in the tableau construction, to check the two conclusions obtained by adding the constraints $x : C$ and $x : D$, respectively. The side condition of the rules are the usual conditions needed to avoid multiple applications on the same principal formula: concerning the example of ($\sqcup^+$), it can be applied only if $x : C \notin S$ and $x : D \notin S$.
- The rules ($\mathbf{T}^+$) and ($\mathbf{T}^-$) are used to "translate" formulas of the form $\mathbf{T}(C)$ in the corresponding modal interpretation: for ($\mathbf{T}^+$), this corresponds to introduce $x : C \sqcap \Box\neg C$ to a constraint system containing $x : \mathbf{T}(C)$, whereas for ($\mathbf{T}^-$) a branching is introduced to add either $x : \neg C$ or $x : \neg\Box\neg C$ in case $x : \neg\mathbf{T}(C)$ belongs to the constraint system.
- The rule ($\sqsubseteq$) is used in order to check whether, for all x belonging to a branch, the inclusion relations of the TBox are satisfied: given a label x and an inclusion $C \sqsubseteq D^L \in U$, the branching introduced by the rule ensures that either $x : \neg C$ holds or that $x : D$ holds.
- The rule ($\Box^-$), applied to a principal formula $x : \neg\Box\neg C$ (x is not a typical instance of the concept C, i.e. there exists an element z which is a typical instance of C and is more normal than x), introduces the constraints $z < x$, $z : C$ and $z : \Box\neg C$. A branching on the choice of the label z to use is introduced, since it can be either a "new" label y, not occurring in the branch, or one of the labels $v_1, v_2, \ldots, v_n$ already belonging to the branch.
  We do not need any extra rule for the positive occurrences of the $\Box$ operator, since these are taken into account by the computation of $S_{x\to y}^M$ of ($\Box^-$). ($\exists^+$) deals with constraints of the form $x : \exists R.C$ in a similar way. The additional side conditions on ($\exists^+$) and ($\Box^-$) are introduced in order to ensure a terminating proof search, by implementing the standard *blocking* technique described below. Intuitively, they are applied to constraints $x : \exists R.C$ and $x : \neg\Box\neg C$, respectively, only if x *is not blocked*, i.e. if there is no label (*witness*) z, labeling the same concepts of x, such that the rule has been already applied to $z : \exists R.C$ (resp. $z : \neg\Box\neg C$). This is formally stated in Definition 18 below.
- The (cut) rule ensures that, given any concept $C \in \mathcal{L}_\mathbf{T}$, an open branch built by $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ contains either $x : \Box\neg C$ or $x : \neg\Box\neg C$ for each label x: this is needed in order to allow $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$ to check the minimality of the model corresponding to the open branch, as we will discuss later.

All the rules of the calculus copy their principal formulas, i.e. the formulas to which the rules are applied, in all their conclusions. As we will discuss later, for the rules ($\exists^+$), ($\forall^-$) and ($\Box^-$) this is used in order to apply the blocking technique, whereas for the rules ($\exists^-$), ($\forall^+$), ($\sqsubseteq$), and (cut) this is needed in order to have a complete calculus. Rules for $\sqcap$, $\sqcup$, $\neg$, and $\mathbf{T}$ also copy their principal formulas in their conclusions for uniformity sake.

In order to ensure the completeness of the calculus, the rules of $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ are applied with the following *standard strategy*:

1. Apply a rule to a label x only if no rule is applicable to a label y such that $y \prec x$.
2. Apply dynamic rules only if no static rule is applicable.

The calculus so obtained is sound and complete with respect to the semantics in Definition 14. In order to prove this, we first define the notion of *regular* node:

Definition 17 (*Regular node*). A node $\langle S | U\rangle$ of $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ is regular if and only if the following conditions hold:

- if $x : \Box\neg C \in S$, then $C \in \mathcal{L}_\mathbf{T}$;
- if $x : \neg\Box\neg C \in S$, then $C \in \mathcal{L}_\mathbf{T}$.

We can show that:

Lemma 6. *Given an $\mathcal{ALC} + \mathbf{T}$ KB, its corresponding constraint system $\langle S|U\rangle$, and a set of concepts $\mathcal{L}_\mathbf{T}$, the nodes of every tableau of $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ having $\langle S|U\rangle$ as a root are regular nodes.*

Proof. Considering each rule of $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$, we can show that if the premise is a regular node, then the conclusions are also regular nodes. The rules introducing boxed formulas are ($\mathbf{T}^+$), ($\mathbf{T}^-$), (cut), and ($\Box^-$). ($\mathbf{T}^+$) and ($\mathbf{T}^-$) introduce $(\neg)\Box\neg C$ in their conclusions when applied to some formula $(\neg)\mathbf{T}(C)$: we conclude that the conclusions are regular nodes, since $C \in \mathcal{L}_\mathbf{T}$ by definition of $\mathcal{L}_\mathbf{T}$ (it contains at least all concepts in the scope of the $\mathbf{T}$ operator). By definition of the rule, (cut) introduces $(\neg)\Box\neg C$ in its conclusions by taking $C \in \mathcal{L}_\mathbf{T}$, and we are done. Concerning ($\Box^-$), suppose an application to a regular node $\langle S, x : \neg\Box\neg C | U\rangle$. Each conclusion has the form $\langle S, x : \neg\Box\neg C, y < x, y : C, y : \Box\neg C, S_{x\to y}^M | U\rangle$, and we conclude

as follows: $C \in \mathcal{L}_\mathbf{T}$, otherwise the premise would not be regular; if $y : \Box\neg D \in S^M_{x \to y}$, then $x : \Box\neg D \in S$ and $D \in \mathcal{L}_\mathbf{T}$, otherwise the premise would not be regular. $\Box$

From now on, by Lemma 6, we restrict our concern to regular nodes.

Furthermore, we introduce the notions of *witness* and of *blocked label*:

Definition 18 *(Witness and blocked label).* Given a constraint system $\langle S|U \rangle$ and two labels x and y occurring in S, we say that x is a witness of y if the following conditions hold:

1. $x \equiv_S y$;
2. $x \prec y$;
3. there is no label z s.t. $z \prec x$ and z satisfies conditions 1 and 2, i.e., x is the least label satisfying conditions 1 and 2 w.r.t. $\prec$.

We say that y is *blocked* by x in $\langle S|U \rangle$ if y has witness x.

By the strategy on the application of the rules described above and by Definition 18, we can prove the following lemma:

Lemma 7. *In any constraint system $\langle S|U \rangle$, if x is blocked, then it has exactly one witness.*

Proof. The property immediately follows from the definition of a witness (Definition 18). $\Box$

As mentioned above, we apply a standard *blocking* technique to control the application of the rules $(\exists^+)$ and $(\Box^-)$, in order to ensure the termination of the calculus. Intuitively, we can apply $(\exists^+)$ to a constraint system of the form $\langle S, x : \exists R.C|U \rangle$ only if x *is not blocked*, i.e. it does not have any witness: indeed, in case x has a witness z, by the strategy on the application of the rules described above the rule $(\exists^+)$ has already been applied to some $z : \exists R.C$, and we do not need a further application to $x : \exists R.C$. This is ensured by the side condition on the application of $(\exists^+)$, namely if $\nexists z \prec x$ such that $z \equiv_{S,x:\exists R.C} x$. The same blocking machinery is used to control the application of $(\Box^-)$, which can be applied only if $\nexists z \prec x$ such that $z \equiv_{S,x:\neg\Box\neg C} x$.

We also need the following definitions:

Definition 19 *(Satisfiability of a branch).* A branch $\mathbf{B}$ of a tableau of $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$ is satisfiable w.r.t. $\mathcal{ALC} + \mathbf{T}$ by a model $\mathcal{M}$ if there is a mapping α from the labels in $\mathbf{B}$ to the domain of $\mathcal{M}$ such that for all constraint systems $\langle S|U \rangle$ on $\mathbf{B}$, $\mathcal{M}$ satisfies under α (see Definition 14) every constraint in S and, for all $C \sqsubseteq D^L \in U$ and for all x occurring in S, we have that if $\alpha(x) \in C^I$ then $\alpha(x) \in D^I$.

Definition 20 *(Saturated branch).* A branch $\mathbf{B} = \langle S_0|U_0 \rangle, \langle S_1|U_1 \rangle, \ldots, \langle S_i|U_i \rangle, \ldots$ is *saturated* if the following conditions hold:

1. for all $C \sqsubseteq D^L$ and for all labels x occurring in $\mathbf{B}$, either $x : \neg C$ or $x : D$ belong to $\mathbf{B}$;
2. if $x : \mathbf{T}(C)$ occurs in $\mathbf{B}$, then $x : C$ and $x : \Box\neg C$ occur in $\mathbf{B}$;
3. if $x : \neg\mathbf{T}(C)$ occurs in $\mathbf{B}$, then either $x : \neg C$ or $x : \neg\Box\neg C$ occurs in $\mathbf{B}$;
4. if $x : \Box\neg C$ and $y < x$ occur in $\mathbf{B}$, also $y : \neg C$ and $y : \Box\neg C$ occur in $\mathbf{B}$;
5. if $x : \neg\Box\neg C$ occurs in $\mathbf{B}$, then either there is y such that $y < x$, $y : C$, $y : \Box\neg C$, and $S^M_{x \to y}$ occur in $\mathbf{B}$ or x is blocked by a witness w, and $y < w$, $y : C$, $y : \Box\neg C$, and $S^M_{w \to y}$ occur in $\mathbf{B}$;
6. if $x : \exists R.C$ occurs in $\mathbf{B}$, then either there is y such that $x \xrightarrow{R} y$ and $y : C$ occur in $\mathbf{B}$ or x is blocked by a witness w, and $w \xrightarrow{R} y$ and $y : C$ occur in $\mathbf{B}$;
7. if $x : \forall R.C$ and $x \xrightarrow{R} y$ occur in $\mathbf{B}$, also $y : C$ occurs in $\mathbf{B}$;
8. for $x : \neg\forall R.C$ and for $x : \neg\exists R.C$ the condition of saturation is defined symmetrically to points 6 and 7, respectively;
9. for the boolean rules the condition of saturation is defined in the usual way. For instance, if $x : C \sqcap D$ occurs in $\mathbf{B}$, so $x : C$ and $x : D$ occur in $\mathbf{B}$;
10. for all $C \in \mathcal{L}_\mathbf{T}$ and for all labels x occurring in $\mathbf{B}$, either $x : \neg\Box\neg C$ occurs in $\mathbf{B}$ or $x : \Box\neg C$ occurs in $\mathbf{B}$.

By following the strategy on the order of application of the rules outlined above and by Lemma 7, we can prove that any open branch can be expanded into an open *saturated* branch. However, it is worth noticing that, as a difference with the tableau calculus for $\mathcal{ALC} + \mathbf{T}$ presented in [26], as well as the one for the standard DL $\mathcal{ALC}$ introduced in [13], the strategy on the order of application of the rules of $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$ does not ensure that the labels are considered one at a time, following the order $\prec$. Indeed, the rules $(\exists^+)$ and $(\Box^-)$ reconsider labels already introduced in the branch in their conclusions. When $(\Box^-)$ is applied to a formula $x : \neg\Box\neg C$, a branching is introduced on the choice of the label used in the

conclusion. In the leftmost conclusion, a "new" label y is used to add $y : C$, $y : \Box\neg C$, $S^M_{x \to y}$. In all the other conclusions, a label v_i already present in the branch is chosen. Therefore, rules of $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$ are furthermore applied to formulas labeled with the "older" label v_i. One may conjecture that this could lead to an incomplete calculus, in particular that condition 4 of saturation above could be not fulfilled. We show in the proof of Proposition 5 below that this does not happen. Intuitively, it could be the case that $v_i : \Box\neg C$ is introduced by $(\Box^-)$, however a previous application of the same rule to $v_i : \neg\Box\neg D$, introducing a new label $u < v_i$, causes the loss of the propagation of the concepts $\neg C$ and $\Box\neg C$ ($u : \neg C$ and $u : \Box\neg C$ should belong to a saturated branch). However, this cannot happen due to the order on the application of the rules and, in particular, by virtue of the rule (cut): since (cut) is a static rule, it has been already applied by using label v_i *before* taking into account labels "younger" than v_i. By Lemma 6, $C \in \mathcal{L}_\mathbf{T}$, therefore either $v_i : \neg\Box\neg C$ or $v_i : \Box\neg C$ have been already introduced: in the former case, the branch is closed, otherwise $u : \neg C$ and $u : \Box\neg C$ have been also introduced by the application of $(\Box^-)$ introducing u (in the computation of $S^M_{v_i \to u}$), ensuring the saturation of the branch.

Proposition 5. *Any open branch* **B** *can be expanded by applying the rules of* $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$ *into an open* saturated *branch.*

Proof. As mentioned, let us analyze the case of condition 4. For the other conditions, the proof is standard and then left to the reader. Suppose that $x : \Box\neg C$ and $y < x$ belong to **B**. The relation $y < x$ has been added to **B** by an application of the rule $(\Box^-)$ to a node $\langle S, x : \neg\Box\neg D | U \rangle$. We show that $x : \Box\neg C$ was already in S before the application of $(\Box^-)$ to $x : \neg\Box\neg D$. Indeed, by the order on the application of the rules, if $(\Box^-)$ is going to be applied to introduce $y < x$, then all the static rules have already been applied to formulas labeled by x, including the (cut) rule. By Lemma 6, we have that $C \in \mathcal{L}_\mathbf{T}$, and (cut) has been also applied to C by using the label x. Therefore, S contains either $x : \neg\Box\neg C$ or $x : \Box\neg C$: the former case cannot be, otherwise **B** would become closed when $x : \Box\neg C$ is introduced, then we are done. By the fact that $x : \Box\neg C \in S$, we can conclude that $y : C$, $y : \Box\neg C$ belong to **B**, since they are introduced by the application of $(\Box^-)$ to $\langle S, x : \neg\Box\neg D | U \rangle$ (we have that $\{y : \neg C, y : \Box\neg C\} \subseteq S^M_{x \to y}$). □

In order to show the completeness of $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$, given an open, saturated branch **B**, we explicitly add to **B** the relation $y < x$, if x is blocked and w is the witness of x and $y < w$ occurs in **B**.

Before proving the completeness, we prove the following lemmas:

Lemma 8. *In any tableau built by* $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$, *there is no open saturated branch* **B** *containing an infinite descending chain of labels* $\cdots x_2 < x_1 < x_0$.

Proof. The only way to obtain an infinite descending chain $\cdots x_2 < x_1 < x_0$ would be to have either (i) a loop or (ii) an infinite set of distinct labels. We can show that neither (i) nor (ii) can occur.

As far as (i) is concerned, suppose for a contradiction that there is a loop, that is to say there is an infinite descending chain $x < u < \cdots < y_i < \cdots < y < x$. We distinguish three cases:

- The relation $x < u$ has been inserted in the branch by the rule $(\Box^-)$ in the leftmost conclusion of this rule: this cannot be the case, since in the leftmost conclusion of the rule x is a new label.
- The relation $x < u$ has been inserted in the branch by the rule $(\Box^-)$ not in the leftmost conclusion, i.e. by using x occurring in **B**, $x \neq u$: the relation $y < x$ has been introduced by an application of $(\Box^-)$, then there is $x : \neg\Box\neg C$ in **B** (the formula to which the $(\Box^-)$ rule is applied). Therefore, $x : \Box\neg C$ belongs to **B**, as well as $y : \Box\neg C$ belongs to **B**. Moreover, $y_i : \Box\neg C$ belongs to **B**, for all y_i, then also $u : \Box\neg C$ belongs to **B**. When $(\Box^-)$ is applied to introduce $x < u$, the constraint $x : \Box\neg C$ is also added to **B**, since $x : \Box\neg C \in S^M_{u \to x}$, which contradicts the hypothesis that **B** was open.
- The relation $x < u$ has been explicitly inserted in the branch because u is blocked by some witness w, and $x < w$ occurs in **B**. Notice, however, that in this case: 1. $x < w$ has been introduced by $(\Box^-)$ applied to some $w : \neg\Box\neg C$, hence, $x : \Box\neg C$ occurs in **B**; 2. similarly to the previous case, it can be shown that also for all y_i and for u, we have that $y_i : \Box\neg C$ and $u : \Box\neg C$ belong to **B**; 3. since w is a witness of u, also $u : \neg\Box\neg C$ occurs in the branch **B**, which contradicts the hypothesis that **B** was open.

Concerning (ii), suppose there were an infinite descending chain $\cdots < x_i < \cdots < x_0$. Each relation must be generated by a $\neg\Box\neg C$ that has not yet been used in the chain, either by an application of the rule $(\Box^-)$ to $\neg\Box\neg C$ in x_{i-1}, or by an application of the rule $(\Box^-)$ to $\neg\Box\neg C$ in the witness w of x_{i-1}. Indeed, if $\neg\Box\neg C$ had been previously used in the chain, say in introducing $x_i < x_{i-1}$, for each x_j such that $x_j < \cdots < x_i$, we have that $x_j : \Box\neg C$ is in **B**, hence $x_j : \neg\Box\neg C$ cannot be in **B**, otherwise **B** would be closed, against the hypothesis. Notice however that, by Lemma 6, the only formulas $\neg\Box\neg C$ that appear in the branch are such that $C \in \mathcal{L}_\mathbf{T}$. Since $\mathcal{L}_\mathbf{T}$ is finite, it follows that also the number of possible different $\neg\Box\neg C$ is finite, and the infinite descending chain cannot be generated. □

Let us now show that all the rules of $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$ are *invertible*. In order to do this, we first show that *weakening* is admissible, namely:

Lemma 9 *(Admissibility of weakening). Given a constraint F and a constraint system $\langle S|U\rangle$, if $\langle S|U\rangle$ has a closed tableau in $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$, then also $\langle S, F|U\rangle$ has a closed tableau in $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$.*

Proof. By induction on the height of the closed tableau for $\langle S|U\rangle$, in the sense of Definition 12. For the base case, it is easy to observe that, if $\langle S|U\rangle$ is a *clash*, then also $\langle S, F|U\rangle$ is a *clash*. As an example, consider the case of $\langle S', x : \bot|U\rangle$, which is an instance of (Clash)$_\bot$: obviously, also $\langle S', x : \bot, F|U\rangle$ is an instance of (Clash)$_\bot$. For the inductive step, we analyze the first step in the tableau construction for $\langle S|U\rangle$, by considering all the rules. We only show the most interesting cases of $(\square^-)$ and $(\sqsubseteq)$, the other cases are similar and left to the reader. Suppose that $(\square^-)$ has been applied to $\langle S', x : \neg\square\neg C|U\rangle$, by generating the conclusion $\langle S', x : \neg\square\neg C, y < x, y : C, y : \square\neg C, S_{x\to y}^M|U\rangle$, where y does not occur in S', as well as the conclusions $\langle S', x : \neg\square\neg C, v_i < x, v_i : C, v_i : \square\neg C, S_{x\to v_i}^M|U\rangle$ for each v_i occurring in S'. We can apply the inductive hypothesis on each conclusion, to obtain a closed tableau for $\langle S', x : \neg\square\neg C, y < x, y : C, y : \square\neg C, S_{x\to y}^M, F|U\rangle$ and for $\langle S', x : \neg\square\neg C, v_i < x, v_i : C, v_i : \square\neg C, S_{x\to v_i}^M, F|U\rangle$, from which we can conclude by an application of $(\square^-)$ to obtain a closed tableau also for $\langle S', x : \neg\square\neg C, F|U\rangle$. Notice that, in case F contains the label y, we can replace y in the tableau with a new label y' wherever it occurs. For $(\sqsubseteq)$, consider a tableau starting with an application of such a rule to $\langle S|U', C \sqsubseteq D^L\rangle$, whose conclusion is $\langle S, x : \neg C \sqcup D|U', C \sqsubseteq D^{L,x}\rangle$ (with $x \notin L$). By inductive hypothesis, we have a closed tableau for $\langle S, x : \neg C \sqcup D, F|U', C \sqsubseteq D^{L,x}\rangle$, from which we obtain a closed tableau for $\langle S, F|U', C \sqsubseteq D^L\rangle$ by an application of $(\sqsubseteq)$. $\square$

Now we can easily prove that the rules of $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ are invertible:

Lemma 10. *Let (R) be a rule of the calculus $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$, let $\langle S|U\rangle$ be its premise and let $\langle S_1|U_1\rangle, \langle S_2|U_2\rangle, \ldots, \langle S_n|U_n\rangle$ be its conclusions. If $\langle S|U\rangle$ has a closed tableau in $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$, then also $\langle S_1|U_1\rangle, \langle S_2|U_2\rangle, \ldots, \langle S_n|U_n\rangle$ have a closed tableau, i.e. the rules of $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ are invertible.*

Proof. It can be easily observed that all the rules of $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ copy their principal formulas in all their conclusions. Therefore, if we have a closed tableau for the premise of a given rule (R), by weakening (Lemma 9 above) we have also a closed tableau for each of its conclusions, and we are done. $\square$

By Lemma 10, we have that in $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ the order of application of the rules is not relevant. Hence, no backtracking is required in the tableau construction, and we can assume, without loss of generality, that a given constraint system $\langle S|U\rangle$ has a unique tableau.

With the above propositions at hand, we can show that:

Theorem 5 *(Soundness of $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$). If the tableau for the constraint system corresponding to $\mathrm{KB} \cup \{\neg F\}$ is closed then $\mathrm{KB} \models_{\mathcal{ALC}+\mathbf{T}} F$.*

Proof. We first show that if the tableau for the constraint system corresponding to $\mathrm{KB} \cup \{\neg F\}$ is closed, then $(*)$ $\mathrm{KB} \cup \{\neg F\}$ is unsatisfiable. By Proposition 4, $\mathrm{KB} \cup \{\neg F\}$ is satisfiable if and only if its corresponding constraint system $\langle S|U\rangle$ is satisfiable in the same model. We proceed by induction on the height of the closed tableau for $\langle S|U\rangle$. For the base case, it is easy to observe that if $\langle S|U\rangle$ is an instance of either (Clash) or (Clash)$_\bot$ or (Clash)$_\top$, then $\mathrm{KB} \cup \{\neg F\}$ is unsatisfiable. For the inductive step, we consider each rule applied to the root $\langle S|U\rangle$ of the closed tableau, and we show that $\mathrm{KB} \cup \{\neg F\}$ is unsatisfiable assuming, by inductive hypothesis, that also the conclusions are unsatisfiable. We proceed by contraposition, that is to say, by considering each rule of $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$, it can be shown that if the premise is satisfiable in an $\mathcal{ALC} + \mathbf{T}$ model, so is (at least) one of its conclusions. To save space, we only show the most interesting case: the $(\square^-)$ rule. The other cases are easy and then left to the reader. Suppose the premise $\langle S, x : \neg\square\neg C|U\rangle$ is satisfiable, i.e. there is a model $\mathcal{M} = \langle \Delta, I, < \rangle$ and a function α such that $\mathcal{M} \models_\alpha F$ for each $F \in S$. Moreover, we have that $C^I \subseteq D^I$ for each $C \sqsubseteq D^L \in U$. Finally, $\mathcal{M} \models_\alpha x : \neg\square\neg C$, i.e. there exists $a \in \Delta$ such that $a < I(x)$ and $a \in C^I$. By Lemma 1, either $a \in Min_<(C^I)$ or there is $b < I(x)$ such that $b \in Min_<(C^I)$. Let c be such individual (a or b) which is preferred to $I(x)$ and belongs to $Min_<(C^I)$. We have that $c \in C^I$ and $c \in (\square\neg C)^I$. Since $c < I(x)$, for all $x : \square\neg D \in S$, we have that $c \in (\neg D)^I$ by Definition 5 and, since $<$ is transitive, $c \in (\square\neg D)^I$. Let us define a function α' such that $\alpha'(k) = \alpha(k)$ for all $k \neq y$, whereas $\alpha'(y) = c$. We conclude that the leftmost conclusion of $(\square^-)$ is satisfiable in $\mathcal{M}$ via α', since y does not occur in S. Indeed, $\mathcal{M} \models_{\alpha'} F$ for all $F \in S$ and, by definition of α', we have that $\mathcal{M} \models_{\alpha'} y < x, y : C, y : \square\neg C, S_{x\to y}^M$.

We can conclude by observing that, if $\mathrm{KB} \not\models_{\mathcal{ALC}+\mathbf{T}} F$, then $\mathrm{KB} \cup \{\neg F\}$ is satisfiable in an $\mathcal{ALC} + \mathbf{T}$ model. Given $(*)$, we conclude that $\mathrm{KB} \models_{\mathcal{ALC}+\mathbf{T}} F$ by contraposition. $\square$

In the proof of the theorem and later in the paper we will use the notion of canonical model $\mathcal{M}^\mathbf{B}$ built from an open branch $\mathbf{B}$. The canonical model $\mathcal{M}^\mathbf{B} = \langle \Delta_B, <', I^B \rangle$ is defined as follows:

- $\Delta_B = \{x: x$ is a label appearing in $\mathbf{B}\}$;

- we first define $<^*$ as follows: $<^* = \{y <^* x$: either $y < x$ occurs in **B** or x is blocked and w is the witness of x (by Lemma 7 such w exists) and $y < w$ occurs in **B**$\}$. We define $<'$ as the transitive closure of relation $<^*$;
- I^B is an interpretation function such that for all atomic concepts A, $A^{I^B} = \{x$ such that $x : A$ occurs in **B**$\}$. I^B is then extended to all concepts C in the standard way, according to the semantics of the operators. For role names R, $R^{I^B} = \{(x, y):$ either $x \xrightarrow{R} y$ occurs in **B** or x is blocked and w is the witness of x (by Lemma 7 such w exists) and $w \xrightarrow{R} y$ occurs in **B**$\}$.

Theorem 6 (*Completeness of* $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$). *If* KB $\models_{\mathcal{ALC}+\mathbf{T}} F$, *then the tableau for the constraint system corresponding to* KB $\cup \{\neg F\}$ *is closed.*

Proof. We show the contrapositive, that if the tableau is open, then the starting constraint system $\langle S|U \rangle$ is satisfiable in an $\mathcal{ALC} + \mathbf{T}$ model, and by Proposition 4 KB $\cup \{\neg F\}$ is satisfiable in the same model, hence KB $\models_{\mathcal{ALC}+\mathbf{T}} F$. An open tableau contains an open branch that by Proposition 5 can be expanded into an open *saturated* branch. From such a branch, call it **B**, we define the canonical model $\mathcal{M}^\mathbf{B} = \langle \Delta_B, <', I^B \rangle$ as described above.

We can show that:

- $<'$ *is irreflexive, transitive, and satisfies the Smoothness Condition.* Irreflexivity follows from the fact that the relation $<$ is either introduced by rule $(\square^-)$ between a label x already present in **B** and either a new label or a label different from x, or it is explicitly added in case some $x : \neg\square\neg C$ is on the branch and x is blocked. In this case, suppose for a contradiction that $x < x$ is added, this means that x is blocked by a witness w and $x < w$, thus $w : \neg\square\neg C$ belongs to **B**, as well as $x < w, x : C, x : \square\neg C$ belong to **B**, but this contradicts the fact that **B** is open (both $x : \neg\square\neg C$ and $x : \square\neg C$ occur). Transitivity follows from definition of $<'$. The Smoothness Condition follows from transitivity of $<'$ together with the finiteness of chains of $<$ deriving from Lemma 8.
- *For all concepts C we have*: (a) *if* $x : C$ *occurs in* **B**, *then* $x \in C^{I^B}$; (b) *if* $x : \neg C$ *occurs in* **B**, *then* $x \in (\neg C)^{I^B}$. We reason by induction on the complexity of C. If C is a boolean combination of concepts, the proof is simple and left to the reader.
  - If C is $\exists R.D$, then by saturation, either $x \xrightarrow{R} y$, $y : C$ occur in **B** or $w \xrightarrow{R} y$, $y : C$ occur in **B**, for w witness of x. In both cases, $(x, y) \in R^{I^B}$ by construction, and by inductive hypothesis $y \in C^{I^B}$, hence (a) follows. (b) can be proven similarly to case (a) in the following item.
  - If C is $\forall R.D$, then by saturation, for all y s.t. $x \xrightarrow{R} y$ occurs in **B**, also $y : D$ occurs in **B**. By construction, $(x, y) \in R^{I^B}$ and, by inductive hypothesis, $y \in D^{I^B}$, hence (a) follows. (b) can be proven similarly to case (a) in the previous item.
  - If C is $\square\neg D$ and $x : \square\neg D$ occurs in **B**, then, by saturation, for all $y < x$, we have that also $y : \neg D$ occurs in **B**. By definition of $<'$, we have that $y <' x$. By inductive hypothesis, $y \notin D^{I^B}$ for all $y <' x$, and we are done. If $x : \neg\square\neg D$ occurs in **B**, then by saturation and by definition of $<'$ there is y s.t. $y <' x$, and $y : D$ and $y : \square\neg D$ occur in **B**. By inductive hypothesis, $y \in D^{I^B}$. It follows that $x \in (\neg\square\neg D)^{I^B}$.
  - If C is $\mathbf{T}(D)$ and $x : \mathbf{T}(D)$ occurs in **B**, by saturation, both $x : D$ and $x : \square\neg D$ occur in **B**, hence by inductive hypothesis $x \in D^{I^B}$ and $x \in (\square\neg D)^{I^B}$, and by Proposition 1, $x \in (\mathbf{T}(D))^{I^B}$. If $x : \neg\mathbf{T}(D)$ occurs in **B**, then by saturation also either $x : \neg D$ occurs in **B** or $x : \neg\square\neg D$ occurs in **B**. By inductive hypothesis either $x \notin D^{I^B}$ or $x \notin (\square\neg D)^{I^B}$. In both cases, we conclude that $x \in (\neg\mathbf{T}(D))^{I^B}$.
- For all $C \sqsubseteq D \in U$ and all labels x, we want to show that either $x \in (\neg C)^{I^B}$ or $x \in D^{I^B}$, i.e., $C^{I^B} \subseteq D^{I^B}$. By saturation, either $x : \neg C$ occurs in **B** or $x : D$ occurs in **B**. The property follows by inductive hypothesis.

The above points allow us to conclude that $\mathcal{M}^\mathbf{B}$ satisfies the starting constraint system. $\square$

The above theorem concerns satisfiability of KB $\cup \{\neg F\}$ in any $\mathcal{ALC} + \mathbf{T}$ model (as in Definition 4). However in order to deal with entailment in $\mathcal{ALC} + \mathbf{T}_{min}$ we need something stronger: we need to restrict our attention to minimal models of KB. Given $\mathcal{L}_\mathbf{T}$, the following theorem shows that KB $\cup \{\neg F\}$ is satisfiable in a minimal model of KB (i.e. KB $\not\models^{\mathcal{L}_\mathbf{T}}_{min} F$) if and only if the open tableau for the constraint system corresponding to KB $\cup \{\neg F\}$ contains an open branch **B** such that $\mathcal{M}^\mathbf{B}$ is a minimal model of KB. $\mathcal{M}^\mathbf{B}$ is the canonical model built from **B** as in the construction used in the proof of Theorem 6 above. With the following theorem, the problem of deciding whether KB $\not\models^{\mathcal{L}_\mathbf{T}}_{min} F$ amounts to deciding whether any possibly open branch **B** of $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$ gives rise to an $\mathcal{M}^\mathbf{B}$ which is a minimal model of KB. As we will see this is the purpose of the second phase of the calculus ($\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH2}$) introduced in the next subsection.

Theorem 7. *Given* $\mathcal{L}_\mathbf{T}$, KB $\models^{\mathcal{L}_\mathbf{T}}_{min} F$ *if and only if there is no open branch* **B** *in the tableau built by* $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$ *for the constraint system corresponding to* KB $\cup \{\neg F\}$ *such that* $\mathcal{M}^\mathbf{B}$ *is a minimal model of* KB.

Proof. *If direction.* We show the contrapositive by proving that if KB $\not\models^{\mathcal{L}_\mathbf{T}}_{min} F$, i.e. if KB $\cup \{\neg F\}$ is satisfiable in a minimal model of KB w.r.t. $\mathcal{L}_\mathbf{T}$, then there is an open branch **B** such that $\mathcal{M}^\mathbf{B}$ is a minimal model of KB. The proof comprises three

steps: (i) if KB $\not\models^{\mathcal{L}_\mathbf{T}}_{min} F$, then there is an open saturated branch **B** for the constraint system corresponding to KB $\cup \{\neg F\}$ which is satisfiable in a minimal model of KB, call it $\mathcal{M}$; (ii) the model $\mathcal{M}'$ obtained from $\mathcal{M}$ by restricting its domain to the elements denoted by the labels in **B**, and by renaming the elements of the domain with the names of the labels in **B** is also a minimal model of *KB* that satisfies **B**; (iii) the canonical model $\mathcal{M}^\mathbf{B}$ for **B** is a minimal model of KB.

For (i), we show that the starting constraint system is satisfiable by $\mathcal{M}$ by an injective mapping (by the unique name assumption in Definition 4), and that each rule preserves the property. For (ii), since **B** is saturated, by reasoning by induction on the complexity of the formulas, it can be proven that $\mathcal{M}'$ (whose extension function and $<$ coincide with those in $\mathcal{M}$ when restricted to the domain in $\mathcal{M}'$) satisfies KB. Furthermore, we can prove that $\mathcal{M}'$ is a minimal model of KB. For a contradiction, suppose it was not. Then, there would be $\mathcal{M}''$, model of KB, with $\mathcal{M}'' < \mathcal{M}'$. Consider then $\mathcal{M}'''$ built by adding to $\mathcal{M}''$ all the elements in $\mathcal{M}$ that are not in $\mathcal{M}''$. The $<$ relation and the extension function in $\mathcal{M}'''$ are defined as in $\mathcal{M}''$ for the elements already present in $\mathcal{M}''$. For the other elements, no $<$ is introduced, and the extension function is defined as for some fixed a in $\mathcal{M}''$ such that there is no $b < a$ in $\mathcal{M}''$ (by the Smoothness Condition in $\mathcal{M}''$ such a exists). It can be shown that $\mathcal{M}'''$ satisfies KB, and that $\mathcal{M}''' < \mathcal{M}$, which contradicts the minimality of $\mathcal{M}$. It follows that also $\mathcal{M}'$ must be minimal. We then obtain the $\mathcal{M}'$ of (ii) by simply renaming its elements with the names of the labels of which they are images. (iii) consider that $\mathcal{M}'$ has the same domain as $\mathcal{M}^\mathbf{B}$. Furthermore, by Definition 20, for all $C \in \mathcal{L}_\mathbf{T}$ and for all labels x occurring in **B**, either $x : \neg\Box\neg C$ occurs in **B** or $x : \Box\neg C$ occurs in **B**. Therefore, in $\mathcal{M}'$ and in $\mathcal{M}^\mathbf{B}$, we have that $x \in (\neg\Box\neg C)^I$ just in case $x : \neg\Box\neg C$ occurs in **B**. Hence, $\mathcal{M}'^{\Box^-} = \mathcal{M}^{\mathbf{B}^{\Box^-}}$, and from the minimality of $\mathcal{M}'$ we conclude that $\mathcal{M}^\mathbf{B}$ is minimal too.

Only if direction. The contrapositive easily follows: since $\mathcal{M}^B$ is a minimal model of KB in which F does not hold, by Definition 8 we conclude that KB $\not\models^{\mathcal{L}_\mathbf{T}}_{min} F$. □

Let us conclude this section by analyzing termination and complexity of $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$. In general, non-termination in labeled tableau calculi can be caused by two different reasons: 1. some rules copy their principal formula in the conclusion(s), and can thus be reapplied over the same formula without any control; 2. dynamic rules may generate infinitely many labels, creating infinite branches. Similarly to the calculus $T^{\mathcal{ALC}+\mathbf{T}}$ for $\mathcal{ALC} + \mathbf{T}$ [26], we adopt the standard loop-checking machinery known as *blocking* to ensure termination.

Concerning the first source of non-termination (point 1), as mentioned above, all the rules copy their principal formulas in their conclusions. However, the side conditions on the application of the rules avoid multiple applications on the same formula. Indeed, $(\sqsubseteq)$ can be applied to a constraint system $\langle S|U, C \sqsubseteq D^L\rangle$ by using the label x only if it has not yet been applied to x in the current branch (i.e., x does not belong to L). Concerning $(\forall^+)$, the rule can be applied to $\langle S, x : \forall R.C, x \xrightarrow{R} y|U\rangle$ only if $y : C$ does not belong to S. When $y : C$ is introduced in the branch, the rule will not further apply to $x : \forall R.C$. Similarly for $(\exists^+)$, $(\Box^-)$, and the rules for **T**, $\neg$, $\sqcap$ and $\sqcup$.

Concerning the second source of non-termination (point 2), we can prove that we only need to adopt the standard loop-checking machinery known as *blocking*, which ensures that the rules $(\exists^+)$ and $(\Box^-)$ do not introduce infinitely many labels on a branch. Thanks to the properties of $\Box$, no other additional machinery would be required to ensure termination. Indeed, it can be shown that the interplay between rules $(\mathbf{T}^-)$ and $(\Box^-)$ does not generate branches containing infinitely many labels.

It is also worth noticing that the (cut) rule does not affect termination, since it is applied only to the finitely many formulas belonging to $\mathcal{L}_\mathbf{T}$.

Let us discuss termination in more detail. Without the side conditions on the rules $(\exists^+)$ and $(\Box^-)$, the calculus $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$ does not ensure a terminating proof search. Indeed, given a constraint system $\langle S|U\rangle$, it could be the case that $(\exists^+)$ is applied to a constraint $x : \exists R.C \in S$, introducing a new label y and the constraints $x \xrightarrow{R} y$ and $y : C$ in the leftmost conclusion. If an inclusion $\mathbf{T}(\exists R.C) \sqsubseteq D$ belongs to U, then $(\sqsubseteq)$ can be applied by using y, thus generating a branch containing $y : \neg\mathbf{T}(\exists R.C)$, to which $(\mathbf{T}^-)$ can be applied introducing $y : \neg\Box\neg(\exists R.C)$. An application of $(\Box^-)$ introduces a new variable z and the constraint $z : \exists R.C$ in the leftmost conclusion, to which $(\exists^+)$ can be applied generating a new label u. $(\sqsubseteq)$ can then be re-applied on $\mathbf{T}(\exists R.C) \sqsubseteq D$ by using u, incurring a loop. In order to avoid this source of non-termination, we adopt the standard technique of *blocking*: the side condition of the $(\exists^+)$ rule says that this rule can be applied to a node $\langle S, x : \exists R.C|U\rangle$ only if x is not blocked. In other words, if there is a witness z of x, then $(\exists^+)$ is not applicable, since the condition and the strategy imply that the $(\exists^+)$ rule has already been applied to z. In this case, we say that x is *blocked* by z. The same for $(\Box^-)$.

As mentioned, another possible source of infinite branches could be determined by the interplay between rules $(\mathbf{T}^-)$ and $(\Box^-)$. However, even if we had no blocking on $(\Box^-)$ this could not occur, i.e., the interplay between these two rules does not generate branches containing infinitely many labels. Intuitively, the application of $(\Box^-)$ to $x : \neg\Box\neg C$ adds $y : \Box\neg C$ to the conclusion, so that $(\mathbf{T}^-)$ can no longer consistently introduce $y : \neg\Box\neg C$. This is due to the properties of $\Box$ (no infinite descending chains of $<$ are allowed). More in detail, if $(\sqsubseteq)$ is applied to $\mathbf{T}(C) \sqsubseteq D$ by using x, an application of $(\mathbf{T}^-)$ introduces a branch containing $x : \neg\Box\neg C$; when a new label y is generated by an application of $(\Box^-)$ on $x : \neg\Box\neg C$, we have that $y : \Box\neg C$ is added to the current constraint system. If $(\sqsubseteq)$ and $(\mathbf{T}^-)$ are also applied to $\mathbf{T}(C) \sqsubseteq D$ on the new label y, then the conclusion where $y : \neg\Box\neg C$ is introduced is closed, by the presence of $y : \Box\neg C$. By this fact, we would not need to introduce any loop-checking machinery on the application of $(\Box^-)$. A detailed proof of termination of the calculus without

blocking on ($\Box^-$) can be found in [27]. However, in this paper we have introduced blocking also on ($\Box^-$) for complexity reasons.

In order to prove that the calculus $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ ensures termination in a rigorous way, we need the following lemma:

Lemma 11. *Given a constraint system $\langle S|U \rangle$, let $n_{\langle S|U \rangle}$ be the number of extended concepts appearing in $\langle S|U \rangle$, including also all the concepts appearing as a substring of another concept. In any set of labels in S including more than $2^{n_{\langle S|U \rangle}}$ labels there are at least two labels x and y s.t. $x \equiv_S y$, i.e. there are at most $2^{n_{\langle S|U \rangle}}$ non-blocked labels.*

Proof. Since there are $n_{\langle S|U \rangle}$ extended concepts, given a label x there cannot be more than $2^{n_{\langle S|U \rangle}}$ different sets of constraints $x : C$ in S. As a consequence, in S there are at most $2^{n_{\langle S|U \rangle}}$ non-blocked labels. $\Box$

Theorem 8 (*Termination of $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$*). *Let $\langle S|U \rangle$ be a constraint system, then any tableau generated by $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ is finite.*

Proof. First, we prove that only a finite number of labels can be introduced in a tableau. The only rules introducing a new label are dynamic rules. However, these rules are applicable only to formulas whose label is not blocked. By Lemma 11, there are at most $2^{n_{\langle S|U \rangle}}$ non-blocked labels in $\langle S|U \rangle$. Dynamic rules can be further applied to those $2^{n_{\langle S|U \rangle}}$ non-blocked labels, therefore obtaining at most $m \times 2^{n_{\langle S|U \rangle}}$ labels, where m is the maximun number of labels directly generated by an application of a dynamic rule from a label in S. When $m \times 2^{n_{\langle S|U \rangle}}$ labels belong to the constraint system, dynamic rules cannot be further applied.

Second, we prove that, since only a finite number of labels are introduced in a tableau, static rules can be applied only a finite number of times. Let us consider all the rules:

- ($\forall^+$): the rule is applied to a constraint system of the form $\langle S, x : \forall R.C, x \xrightarrow{R} y|U \rangle$, to obtain a conclusion of the form $\langle S, x : \forall R.C, x \xrightarrow{R} y, y : C|U \rangle$. However, the side condition on the application of the rule imposes that the rule is applied if $y : C \notin S$, therefore it is applied only once in a branch, for a given $\forall R.C$ and for two labels x and y. Since only a finite number of labels as well as a finite number of formulas $\forall R.C$ are introduced in a tableau (for the formulas, only (sub-)formulas of the initial KB or (sub-)formulas of $\mathcal{L}_\mathbf{T}$ or (sub-)formulas of the query), we can conclude that the rule ($\forall^+$) is applied only a finite number of times.
- Rules ($\sqcap^+$), ($\sqcap^-$), ($\sqcup^+$), ($\sqcup^-$), ($\neg$), ($\mathbf{T}^+$), ($\mathbf{T}^-$), ($\exists^-$): the application of these rules is restricted exactly as ($\forall^+$), then we can conclude as we have done in the previous case.
- (*cut*): just observe that it is applied by introducing $x : (\neg)\Box\neg C$ for all concepts $C \in \mathcal{L}_\mathbf{T}$: since $\mathcal{L}_\mathbf{T}$ is finite, and we have to consider a finite number of labels x, this rule is applied only a finite number of times.
- ($\sqsubseteq$): we can reason analogously to what done for (*cut*), since ($\sqsubseteq$) is applied to a finite set of subsumption relations $C \sqsubseteq D \in U$ by using a finite number of labels. $\Box$

Since $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ is sound and complete (Theorem 5 and Theorem 6), and since a KB is satisfiable in an $\mathcal{ALC}+\mathbf{T}$ model iff its corresponding constraint system is satisfiable in the same model (Proposition 4), from Theorem 8 above it follows that checking whether a given KB (TBox, ABox) is satisfiable is a decidable problem.

Furthermore, we can prove that, with the calculus $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ above, the satisfiability of a KB can be decided in nondeterministic exponential time in the size of the KB.

Theorem 9 (*Complexity*). *Given a KB and a query F, checking whether $\mathrm{KB} \cup \{\neg F\}$ is satisfiable in an $\mathcal{ALC}+\mathbf{T}$ model can be solved in nondeterministic exponential time.*

Proof. In order to check whether $\mathrm{KB} \cup \{\neg F\}$ is satisfiable w.r.t $\mathcal{ALC}+\mathbf{T}$, we build its corresponding constraint system $\langle S|U \rangle$ and we try to build a tableau having $\langle S|U \rangle$ as a root by means of the rules of $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$. We first show that the number of labels generated on a branch is at most exponential in the size of $\mathrm{KB} \cup \{\neg F\}$. Let n be the size of $\mathrm{KB} \cup \{\neg F\}$. Given a constraint system $\langle S|U \rangle$, the number of extended concepts appearing in $\langle S|U \rangle$, including also all the ones appearing as a subformula of other concepts, is $O(n)$. We have already shown in Lemma 11 that, as there are at most $O(n)$ concepts, there are at most $O(2^n)$ variables labeling distinct sets of concepts. Hence, there are $O(2^n)$ non-blocked variables in S.

Let m be the maximum number of direct successors of each variable x occurring in S, obtained by applying dynamic rules. m is bound by the number of $\exists R.C$ concepts ($O(n)$) plus the number of $\neg \forall R.C$ concepts ($O(n)$) plus the number of $\neg \Box \neg C$ concepts ($O(n)$). Therefore, there are at most $O(2^n \times m)$ *variables* in S, where $m \leqslant 3n$. The number of *individual constants* in the ABox is bound by n too, and each individual constant has at most m direct successors. The number of *labels* in S is then bound by $(2^n + n) \times m \leqslant (2^n + n) \times 3n \leqslant (2^n + 3n) \times (2^n + 3n) = (2^n + 3n)^2$, and hence by $O(2^{2n})$.

For a given label x, the concepts labeled by x introduced in the branch (namely, all the possible subconcepts of the initial constraint system, as well as all boxed subconcepts) are $O(n)$. Hence, the labeled concepts introduced on the branch is $O(n)$ for each label, and the number of all labeled concepts on the branch is $O(n \times 2^{2n})$. Since no rule deletes the principal formula to which it is applied, a branch can contain at most an exponential number of applications of tableau rules.

$$\langle S, x : C, x : \neg C \mid U \mid K \rangle \quad \text{(Clash)} \qquad \langle S, x : \bot \mid U \mid K \rangle \quad \text{(Clash)}_\bot \qquad \langle S, x : \neg\top \mid U \mid K \rangle \quad \text{(Clash)}_\top \qquad \langle S \mid U \mid \emptyset \rangle \quad \text{(Clash)}_\emptyset \qquad \langle S, x : \neg\square\neg C \mid U \mid K \rangle \quad \text{(Clash)}_{\square^-} \quad \text{if } x : \neg\square\neg C \notin \mathbf{B}^{\square^-}$$

$$\frac{\langle S, x : \neg(C \sqcap D) \mid U \mid K \rangle}{\langle S, x : \neg C \mid U \mid K \rangle \qquad \langle S, x : \neg D \mid U \mid K \rangle} (\sqcap^-) \qquad\qquad \frac{\langle S, x : C \sqcap D \mid U \mid K \rangle}{\langle S, x : C, x : D \mid U \mid K \rangle} (\sqcap^+) \qquad\qquad \frac{\langle S, x : \neg\neg C \mid U \mid K \rangle}{\langle S, x : C \mid U \mid K \rangle} (\neg)$$

$$\frac{\langle S, x : \mathbf{T}(C) \mid U \mid K \rangle}{\langle S, x : C, x : \square\neg C \mid U \mid K \rangle} (\mathbf{T}^+) \qquad\qquad\qquad\qquad \frac{\langle S, x : \neg\mathbf{T}(C) \mid U \mid K \rangle}{\langle S, x : \neg C \mid U \mid K \rangle \qquad \langle S, x : \neg\square\neg C \mid U \mid K \rangle} (\mathbf{T}^-)$$

$$\frac{\langle S \mid U, C \sqsubseteq D^L \mid K \rangle}{\langle S, x : \neg C \sqcup D \mid U, C \sqsubseteq D^{L,x} \mid K \rangle} (\sqsubseteq) \qquad\qquad \frac{\langle S, x : \forall R.C, x \xrightarrow{R} y \mid U \mid K \rangle}{\langle S, x : \forall R.C, x \xrightarrow{R} y, y : C \mid U \mid K \rangle} (\forall^+)$$
$$x \in \mathcal{D}(\mathbf{B}) \text{ and } x \notin L \qquad\qquad\qquad\qquad \text{if } y : C \notin S$$

$$\frac{\langle S, x : \neg\square\neg C \mid U \mid K, x : \neg\square\neg C \rangle}{\langle S, v_1 : C, v_1 : \square\neg C, S^M_{x \to v_1}, x : \neg\square\neg C \mid U \mid K \rangle \quad \langle S, v_2 : C, v_2 : \square\neg C, S^M_{x \to v_2}, x : \neg\square\neg C \mid U \mid K \rangle \cdots \langle S, v_n : C, v_n : \square\neg C, S^M_{x \to v_n}, x : \neg\square\neg C \mid U \mid K \rangle} (\square^-)$$
$$\text{if } \nexists u \text{ s.t. } \{u : C, u : \square\neg C, S^M_{x \to u}\} \subseteq S$$
$$\forall v_i \in \mathcal{D}(\mathbf{B}), x \neq v_i$$

$$\frac{\langle S \mid U \mid K \rangle}{\langle S, x : \square\neg C \mid U \mid K \rangle \quad \langle S, x : \neg\square\neg C \mid U \mid K \rangle} (cut) \qquad \frac{\langle S, x : \exists R.C \mid U \mid K \rangle}{\langle S, x \xrightarrow{R} v_1, v_1 : C \mid U \mid K \rangle \quad \langle S, x \xrightarrow{R} v_2, v_2 : C \mid U \mid K \rangle \cdots \langle S, x \xrightarrow{R} v_n, v_n : C \mid U \mid K \rangle} (\exists^+)$$
$$\text{if } x : \neg\square\neg C \notin S \text{ and } x : \square\neg C \notin S \qquad\qquad\qquad \forall v_i \in \mathcal{D}(\mathbf{B})$$
$$C \in \mathcal{L}_\mathbf{T}$$
$$x \in \mathcal{D}(\mathbf{B})$$

Fig. 2. The calculus $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH2}$. To save space, we omit the rules $(\sqcup^+)$ and $(\sqcup^-)$.

The satisfiability of KB $\cup \{\neg F\}$ can thus be solved by defining a procedure which nondeterministically generates an open branch of $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$ of exponential size (in the size of KB $\cup \{\neg F\}$). The problem is in NExpTime. $\square$

5.2. The tableau calculus $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH2}$

Let us now introduce the calculus $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH2}$ which, for each open branch $\mathbf{B}$ built by $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$, verifies if $\mathcal{M}^\mathbf{B}$ is a minimal model of the KB.

Definition 21. Given an open branch $\mathbf{B}$ of a tableau built from $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$, we define:

- $\mathcal{D}(\mathbf{B})$ as the set of labels occurring on $\mathbf{B}$;
- $\mathbf{B}^{\square^-} = \{x : \neg\square\neg C \mid x : \neg\square\neg C \text{ occurs in } \mathbf{B}\}$.

A tableau of $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH2}$ is a tree whose nodes are triples of the form $\langle S|U|K \rangle$, where $\langle S|U \rangle$ is a constraint system, whereas K contains formulas of the form $x : \neg\square\neg C$, with $C \in \mathcal{L}_\mathbf{T}$.

The basic idea of $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH2}$ is as follows. Given an open branch $\mathbf{B}$ built by $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$ and corresponding to a model $\mathcal{M}^\mathbf{B}$ of KB $\cup \{\neg F\}$, $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH2}$ checks whether $\mathcal{M}^\mathbf{B}$ is a minimal model of KB by trying to build a model of KB which is preferred to $\mathcal{M}^\mathbf{B}$. Starting from $\langle S|U|\mathbf{B}^{\square^-} \rangle$ where $\langle S|U \rangle$ is the constraint system corresponding to the initial KB $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH2}$ tries to build an open branch containing all and only the labels appearing on $\mathbf{B}$, i.e. those in $\mathcal{D}(\mathbf{B})$, and containing less negated boxed formulas than $\mathbf{B}$ does. To this aim, first the dynamic rules use labels in $\mathcal{D}(\mathbf{B})$ instead of introducing new ones in their conclusions. Second the negated boxed formulas used in $\mathbf{B}$ are stored in the additional set K of a tableau node, initialized with $\mathbf{B}^{\square^-}$. A branch built by $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH2}$ closes if it does not represent a model preferred to the candidate model $\mathcal{M}^\mathbf{B}$, and this happens if the branch contains a contradiction (Clash) or it contains at least all the negated boxed formulas contained in $\mathbf{B}$ ((Clash)$_{\square^-}$ and (Clash)$_\emptyset$).

More in detail the rules of $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH2}$ are shown in Fig. 2. The rule $(\exists^+)$ is applied to a constraint system containing a formula $x : \exists R.C$; it introduces $x \xrightarrow{R} y$ and $y : C$ where $y \in \mathcal{D}(\mathbf{B})$, instead of y being a new label. The choice of the label y introduces a branching in the tableau construction. The rule $(\sqsubseteq)$ is applied in the same way as in $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$ to all the labels of $\mathcal{D}(\mathbf{B})$ (and not only to those appearing in the branch). The rule $(\square^-)$ is applied to a node $\langle S, x : \neg\square\neg C|U|K \rangle$, when $x : \neg\square\neg C \in K$, i.e. when the formula $x : \neg\square\neg C$ also belongs to the open branch $\mathbf{B}$. In this case, the rule introduces

a branch on the choice of the individual $v_i \in \mathcal{D}(\mathbf{B})$ which is preferred to x and is such that C and $\Box\neg C$ hold in v_i. In case a tableau node has the form $\langle S, x : \neg\Box\neg C|U|K\rangle$, and $x : \neg\Box\neg C \notin \mathbf{B}^{\Box^-}$, then $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$ detects a clash, called (Clash)$_{\Box^-}$: this corresponds to the situation in which $x : \neg\Box\neg C$ does not belong to $\mathbf{B}$, while $S, x : \neg\Box\neg C$ is satisfiable in a model $\mathcal{M}$ only if $\mathcal{M}$ contains $x : \neg\Box\neg C$, and hence only if $\mathcal{M}$ is *not* preferred to the model represented by $\mathbf{B}$.

The calculus $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$ also contains the clash condition (Clash)$_\emptyset$. Since each application of ($\Box^-$) removes the principal formula $x : \neg\Box\neg C$ from the set K, when K is empty all the negated boxed formulas occurring in $\mathbf{B}$ also belong to the current branch. In this case, the model built by $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$ satisfies the same set of negated boxed formulas (for all individuals) as $\mathbf{B}$ and, thus, it is not preferred to the one represented by $\mathbf{B}$.

We can now prove that:

Theorem 10 *(Soundness and completeness of $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$). Given a KB and a query F, let $\langle S|U\rangle$ be the corresponding constraint system of KB$\cup\{\neg F\}$ and $\langle S'|U\rangle$ be the corresponding constraint system of KB. Given an open saturated branch $\mathbf{B}$ built by $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ for $\langle S|U\rangle$, the canonical model $\mathcal{M}^{\mathbf{B}}$ built from $\mathbf{B}$ is a minimal model of KB iff the tableau in $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$ for $\langle S'|U|\mathbf{B}^{\Box^-}\rangle$ is closed.*

Proof. First, given an open branch $\mathbf{B}$ built from $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$, by Theorem 6 and Proposition 4, $\mathcal{M}^{\mathbf{B}} = \langle \Delta_B, <, I^B\rangle$ is a model of KB. In order to show the soundness (if direction), we show that if the tableau in $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$ for $\langle S'|U|\mathbf{B}^{\Box^-}\rangle$ is closed, then $\mathcal{M}^{\mathbf{B}}$ is a minimal model of KB. We show the contrapositive, that if $\mathcal{M}^{\mathbf{B}}$ was not minimal (i.e. if there was a model $\mathcal{M} = \langle \Delta, <, I\rangle$ of KB s.t. $\mathcal{M} <_{\mathcal{L}_\mathbf{T}} \mathcal{M}^{\mathbf{B}}$) then there would be an open branch in $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$ for $\langle S'|U|\mathbf{B}^{\Box^-}\rangle$ by showing that: (i) $\langle S'|U\rangle$ would be satisfiable in $\mathcal{M}$ under the identity function i, (ii) each rule of the calculus preserves the satisfiability in $\mathcal{M}$ under i, and (iii) no clash condition is satisfiable in such a model under i. (i) $\mathcal{M}$ is a model of KB, and for all individual constants a in the ABox $a^I = a$ (since $a^I = a^{I^B}$ and $a^{I^B} = a$); by Proposition 4 it can be easily shown that also (i) holds. (ii) can be easily proven for all the rules. To save space, we only consider rules ($\Box^-$) and ($\exists^+$). The other rules are easy and then left to the reader. ($\Box^-$): suppose the premise $\langle S, x : \neg\Box\neg C|U|K, x : \neg\Box\neg C\rangle$ is satisfiable in $\mathcal{M}$ under i, i.e. $x \in \neg\Box\neg C^I$. Then there must be $v_i < x$ in $\mathcal{M}$ with $v_i \in \Delta = \Delta_B = \mathcal{D}(\mathbf{B})$ such that $v_i \in C^I$, $v_i \in \Box\neg C^I$, $v_i \in \neg C'^I$, $v_i \in \Box\neg C'^I$ for all the formulas $v_i : \neg C'$, $v_i : \Box\neg C'$ in $S^M_{x\to v_i}$. It immediately follows that the conclusion of ($\Box^-$) containing $\langle S, v_i : C^I, v_i : \Box\neg C^I, S^M_{x\to v_i}, x : \neg\Box\neg C|U|K\rangle$ is satisfiable in $\mathcal{M}$ under i. ($\exists^+$): suppose the premise $\langle S, x : \exists R.|U|K\rangle$ is satisfiable in $\mathcal{M}$ under i, i.e. $x \in \exists R.C^I$. Then there is $v_i \in \Delta = \mathcal{D}(\mathbf{B})$ s.t. $(x, v_i) \in R^I$, and $v_i \in C^I$. The conclusion of the rule containing $\langle S, x \xrightarrow{R} v_i, v_i : C|U|K\rangle$ is therefore satisfiable in $\mathcal{M}$ under i. (iii) clearly holds for (Clash), (Clash)$_\perp$ and (Clash)$_\top$. For (Clash)$_\emptyset$: if $K = \emptyset$, this means that rule $\Box^-$ (the only that removes formulas $x : \neg\Box\neg C$ from K) has been applied to all $x : \neg\Box\neg C$ in $\mathbf{B}^{\Box^-}$, and all $x : \neg\Box\neg C$ in $\mathbf{B}^{\Box^-}$ are in S. However in this case the constraint system $\langle S|U|K\rangle$ in (Clash)$_\emptyset$ is not satisfiable in $\mathcal{M}$ since by hypothesis $\mathcal{M}^{\Box^-} \subset \mathcal{M}^{\mathbf{B}^{\Box^-}}$. Last (iii) holds for (Clash)$_{\Box^-}$, otherwise there would be a $\neg\Box\neg C$ s.t. $\mathcal{M} \models_i x : \neg\Box\neg C$, i.e. $x \in (\neg\Box\neg C)^I$ in $\mathcal{M}$ but $x : \neg\Box\neg C$ does not belong to $\mathbf{B}^{\Box^-}$, i.e. $x \notin (\neg\Box\neg C)^{I^B}$ in $\mathcal{M}^B$, which contradicts that $\mathcal{M}^{\Box^-} \subset \mathcal{M}^{\mathbf{B}^{\Box^-}}$.

We now consider the completeness (only if direction). By hypothesis, $\mathcal{M}^{\mathbf{B}}$ is a minimal model for KB. We want to show that the tableau in $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$ for $\langle S'|U|\mathbf{B}^{\Box^-}\rangle$ is closed. For a contradiction, suppose that the tableau was open, with an open branch $\mathbf{B}'$. It can be easily shown that the canonical model $\mathcal{M}^{B'}$ built from $\mathbf{B}'$ would be a model of KB which is preferred to $\mathcal{M}^B$. Indeed, the domain of $\mathcal{M}^B$ coincides with that of $\mathcal{M}^{B'}$ (which is $\mathcal{D}(\mathbf{B})$). Furthermore, $\mathcal{M}_{\mathcal{L}_\mathbf{T}}^{B'^{\Box^-}} \subset \mathcal{M}_{\mathcal{L}_\mathbf{T}}^{B^{\Box^-}}$, since the negated box formulas that hold in these canonical models are those that explicitly appear on the branch (by (cut)) for all $C \in \mathcal{L}_\mathbf{T}$, for all labels x, either $x : \Box\neg C \in \mathbf{B}'$ or $x : \neg\Box\neg C \in \mathbf{B}'$), and $\mathbf{B}'^{\Box^-} \subset \mathbf{B}^{\Box^-}$, otherwise by (Clash)$_\emptyset$ $\mathbf{B}'$ would be closed. However, this contradicts the minimality of $\mathcal{M}^{\mathbf{B}}$. This contradiction forces us to conclude that there cannot be an open $\mathbf{B}'$ in $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$, and that the tableau must be closed. $\quad\Box$

$\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$ always terminates. Intuitively, termination is ensured by the fact that dynamic rules make use of labels belonging to $\mathcal{D}(\mathbf{B})$, which is finite, rather than introducing "new" labels in the tableau.

Theorem 11 *(Termination of $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$). Let $\langle S'|U|\mathbf{B}^{\Box^-}\rangle$ be a constraint system starting from an open branch $\mathbf{B}$ built by $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$, then any tableau generated by $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$ is finite.*

Proof. Only a finite number of labels can occur on the tableau built by $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$, namely only those in $\mathcal{D}(\mathbf{B})$ which is finite. Moreover, the side conditions on the application of the rules ($\forall^+$), ($\sqsubseteq$), ($\Box^-$), and (cut), copying their principal formulas in their conclusion(s), avoid the uncontrolled application of the same rules. $\quad\Box$

Definition 22. Let KB be a knowledge base whose corresponding constraint system is $\langle S|U\rangle$. Let F be a query and let S' be the set of constraints obtained by adding to S the constraint corresponding to $\neg F$. The calculus $\mathcal{TAB}_{min}^{\mathcal{ALC}+\mathbf{T}}$ checks whether a query F can be minimally entailed from a KB by means of the following procedure:

- the calculus $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ is applied to $\langle S'|U\rangle$;
- if, for each branch **B** built by $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$, either:
   (i) **B** is closed or
   (ii) the tableau built by the calculus $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$ for $\langle S|U|\mathbf{B}^{\square^-}\rangle$ is open,
   then the procedure says YES
   else the procedure says NO.

The following theorem shows that the overall procedure is sound and complete.

Theorem 12 (*Soundness and completeness of* $\mathcal{TAB}_{min}^{\mathcal{ALC}+\mathbf{T}}$). $\mathcal{TAB}_{min}^{\mathcal{ALC}+\mathbf{T}}$ *is a sound and complete decision procedure for verifying if* $KB \models_{min}^{\mathcal{L}_{\mathbf{T}}} F$.

Proof. (Soundness) We show that if the procedure outputs YES, then $KB \models_{min}^{\mathcal{L}_{\mathbf{T}}} F$ holds. If the procedure outputs YES, then for all branches generated by $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ either they are closed or (ii) holds. If all branches generated by $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ are closed then, by Theorem 5 we have that $KB \models_{\mathcal{ALC}+\mathbf{T}} F$, then we conclude that $KB \models_{min}^{\mathcal{L}_{\mathbf{T}}} F$. Consider now all open branches **B** generated by $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$. Since the procedure outputs YES, then (ii) must hold for all **B**, i.e. the tableau built by $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$ for $\langle S|U|\mathbf{B}^{\square^-}\rangle$ is open. In this case, by Theorem 10, for all **B**, $\mathcal{M}^{\mathbf{B}}$ is not a minimal model of KB and, by Theorem 7, $KB \models_{min}^{\mathcal{L}_{\mathbf{T}}} F$ holds.

(Completeness) We show that if $KB \models_{min}^{\mathcal{L}_{\mathbf{T}}} F$ holds then the procedure outputs YES. First of all if all branches generated by $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ are closed, (i) holds for all branches and then the procedure outputs YES. Suppose now there are open branches **B** generated by $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$. Since $KB \models_{min}^{\mathcal{L}_{\mathbf{T}}} F$, by Theorem 7, $\mathcal{M}^{\mathbf{B}}$ is not a minimal model of KB and by Theorem 10 the tableau in $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$ for $\langle S'|U|\mathbf{B}^{\square^-}\rangle$ is open, hence (ii) holds and the procedure outputs YES. □

We provide an upper bound on the complexity of the procedure for computing the minimal entailment $KB \models_{min}^{\mathcal{L}_{\mathbf{T}}} F$:

Theorem 13 (*Complexity of* $\mathcal{TAB}_{min}^{\mathcal{ALC}+\mathbf{T}}$). *The problem of deciding whether* $KB \models_{min}^{\mathcal{L}_{\mathbf{T}}} F$ *is in* co-NExp^{NP}.

Proof. We first consider the complementary problem: $KB \not\models_{min}^{\mathcal{L}_{\mathbf{T}}} F$. This problem can be solved according to the procedure in Definition 22: by nondeterministically generating (NExp) an open branch of exponential length in the size of KB in $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$ (a model $\mathcal{M}^{\mathbf{B}}$ of $KB \cup \{\neg F\}$), and then by calling an NP oracle which verifies that $\mathcal{M}^{\mathbf{B}}$ is a minimal model of KB. In fact, the verification that $\mathcal{M}^{\mathbf{B}}$ is not a minimal model of the KB can be done by an NP algorithm which non-deterministically generates a branch in $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$ (of polynomial size in the size of $\mathcal{M}^{\mathbf{B}}$), representing a model $\mathcal{M}^{B'}$ of KB preferred to $\mathcal{M}^{\mathbf{B}}$. Hence, the problem of verifying that $KB \not\models_{min}^{\mathcal{L}_{\mathbf{T}}} F$ is in NExp^{NP}, and the problem of deciding whether $KB \models_{min}^{\mathcal{L}_{\mathbf{T}}} F$ is in co-NExp^{NP}. □

By the above results, observe that if a formula is satisfiable in a minimal model, then there is a tableau in $\mathcal{TAB}_{min}^{\mathcal{ALC}+\mathbf{T}}$ containing a finite branch which is open in phase 1 and whose corresponding tableau in phase 2 is closed. By the above construction this branch provides a finite minimal model of the formula. Therefore, we obtain the following theorem:

Theorem 14 (*Finite model property of* $\mathcal{ALC} + \mathbf{T}_{min}$). *The logic* $\mathcal{ALC} + \mathbf{T}_{min}$ *has the finite model property.*

The above result on complexity of minimal entailment implicitly provides an upper bound on the size of a minimal model of a KB satisfiable in $\mathcal{ALC} + \mathbf{T}$.

As an example, let a KB contain the following formulas:

TBox

$$\mathbf{T}(C) \sqsubseteq \neg P$$

ABox

$$C(a)$$

$$D(a)$$

We adopt the calculus $\mathcal{TAB}_{min}^{\mathcal{ALC}+\mathbf{T}}$ in order to show that
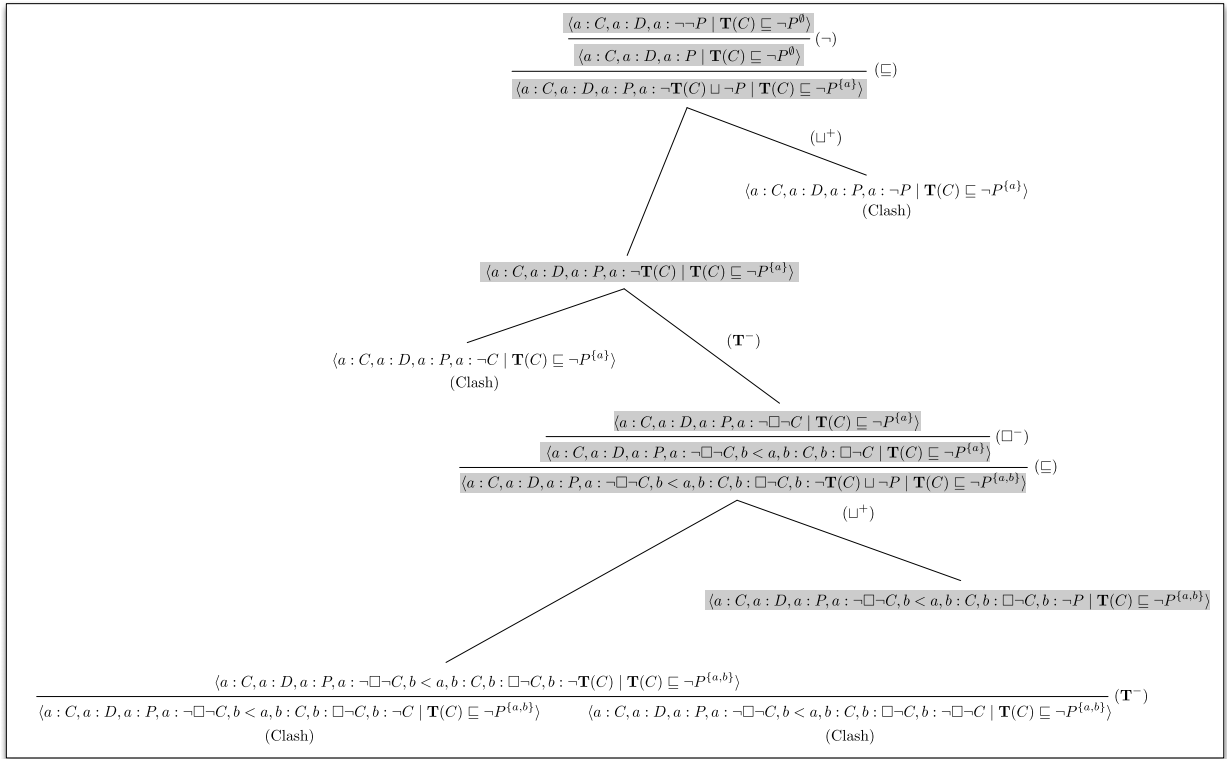
$$KB \models_{min}^{\mathcal{L}_{\mathbf{T}}} \neg P(a)$$

$$\frac{\langle a:C, a:D, a:\neg\neg P \mid \mathbf{T}(C) \sqsubseteq \neg P^{\emptyset}\rangle}{\langle a:C, a:D, a:P \mid \mathbf{T}(C) \sqsubseteq \neg P^{\emptyset}\rangle}\ (\neg)$$

$$\frac{}{\langle a:C, a:D, a:P, a:\neg\mathbf{T}(C) \sqcup \neg P \mid \mathbf{T}(C) \sqsubseteq \neg P^{\{a\}}\rangle}\ (\sqsubseteq)$$

$$(\sqcup^+)$$

$$\langle a:C, a:D, a:P, a:\neg P \mid \mathbf{T}(C) \sqsubseteq \neg P^{\{a\}}\rangle\ (\text{Clash})$$

$$\langle a:C, a:D, a:P, a:\neg\mathbf{T}(C) \mid \mathbf{T}(C) \sqsubseteq \neg P^{\{a\}}\rangle$$

$$(\mathbf{T}^-)$$

$$\langle a:C, a:D, a:P, a:\neg C \mid \mathbf{T}(C) \sqsubseteq \neg P^{\{a\}}\rangle\ (\text{Clash})$$

$$\frac{\langle a:C, a:D, a:P, a:\neg\Box\neg C \mid \mathbf{T}(C) \sqsubseteq \neg P^{\{a\}}\rangle}{\langle a:C, a:D, a:P, a:\neg\Box\neg C, b<a, b:C, b:\Box\neg C \mid \mathbf{T}(C) \sqsubseteq \neg P^{\{a\}}\rangle}\ (\Box^-)$$

$$\frac{}{\langle a:C, a:D, a:P, a:\neg\Box\neg C, b<a, b:C, b:\Box\neg C, b:\neg\mathbf{T}(C) \sqcup \neg P \mid \mathbf{T}(C) \sqsubseteq \neg P^{\{a,b\}}\rangle}\ (\sqsubseteq)$$

$$(\sqcup^+)$$

$$\langle a:C, a:D, a:P, a:\neg\Box\neg C, b<a, b:C, b:\Box\neg C, b:\neg P \mid \mathbf{T}(C) \sqsubseteq \neg P^{\{a,b\}}\rangle$$

$$\langle a:C, a:D, a:P, a:\neg\Box\neg C, b<a, b:C, b:\Box\neg C, b:\neg\mathbf{T}(C) \mid \mathbf{T}(C) \sqsubseteq \neg P^{\{a,b\}}\rangle$$

$$(\mathbf{T}^-)$$

$$\langle a:C, a:D, a:P, a:\neg\Box\neg C, b<a, b:C, b:\Box\neg C, b:\neg C \mid \mathbf{T}(C) \sqsubseteq \neg P^{\{a,b\}}\rangle\ (\text{Clash})$$

$$\langle a:C, a:D, a:P, a:\neg\Box\neg C, b<a, b:C, b:\Box\neg C, b:\neg\Box\neg C \mid \mathbf{T}(C) \sqsubseteq \neg P^{\{a,b\}}\rangle\ (\text{Clash})$$

Fig. 3. A tableau in $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$ used in order to check whether $KB \models^{\mathcal{L}_\mathbf{T}}_{min} \neg P(a)$, where $KB = \mathbf{T}(C) \sqsubseteq \neg P, C(a), D(a)$ (phase 1).

with $\mathcal{L}_\mathbf{T} = \{C\}$. First, the calculus $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$ is applied to the corresponding constraint system of $KB \cup \{\neg\neg P(a)\}$, namely to the root

$$\langle a:C, a:D, a:\neg\neg P \big| \mathbf{T}(C) \sqsubseteq \neg P^{\emptyset}\rangle$$

The tableau built by $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH1}$ is shown in Fig. 3. The tableau contains only one open branch, say **B**, evidenced in light grey. We now apply the procedure $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH2}$ to **B**: the root of the tableau is initialized with $\langle S|U|K\rangle$, where $\langle S|U\rangle$ is the corresponding constraint system of KB, and $K = \{a:\neg\Box\neg C\}$, i.e. we start the tableau with

$$\langle a:C, a:D \big| \mathbf{T}(C) \sqsubseteq \neg P^{\emptyset} \big| a:\neg\Box\neg C\rangle$$

The tableau built by $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH2}$ is shown in Fig. 4. It contains (at least) one open branch (again, the one evidenced in light grey). Therefore, the tableau $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{PH2}$ for **B** is open, whence **B** is closed. Thus the whole procedure $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{min}$ verifies that $KB \models^{\mathcal{L}_\mathbf{T}}_{min} \neg P(a)$.

6. Other reasoning problems

In previous sections we have focused on the problem of minimal entailment in $\mathcal{ALC} + \mathbf{T}_{min}$. In this section we show how the other well-known reasoning problems in DLs can be reduced to minimal entailment. We also provide complexity upper bounds for such problems.

Given the main reasoning problems that can be found in the DLs literature [2], we define the corresponding reasoning problems in $\mathcal{ALC} + \mathbf{T}_{min}$:

Definition 23 (*Reasoning problems*). Given a KB = (TBox, ABox) and a set $\mathcal{L}_\mathbf{T}$, we define the following reasoning problems:

- *Instance checking*: given an individual constant a occurring in ABox and an extended concept C, we say that a is an instance of C with respect to KB if in all minimal models of KB with respect to $\mathcal{L}_\mathbf{T}$, it holds that $a^I \in C^I$.
- *Subsumption*: given two extended concepts C and D, we say that C is subsumed by D ($C \sqsubseteq D$) with respect to KB if in all minimal models of KB with respect to $\mathcal{L}_\mathbf{T}$, it holds that $C^I \subseteq D^I$.
- *Concept satisfiability*: given an extended concept C, we say that C is satisfiable with respect to KB if there exists a minimal model of KB with respect to $\mathcal{L}_\mathbf{T}$ in which $C^I \neq \emptyset$.
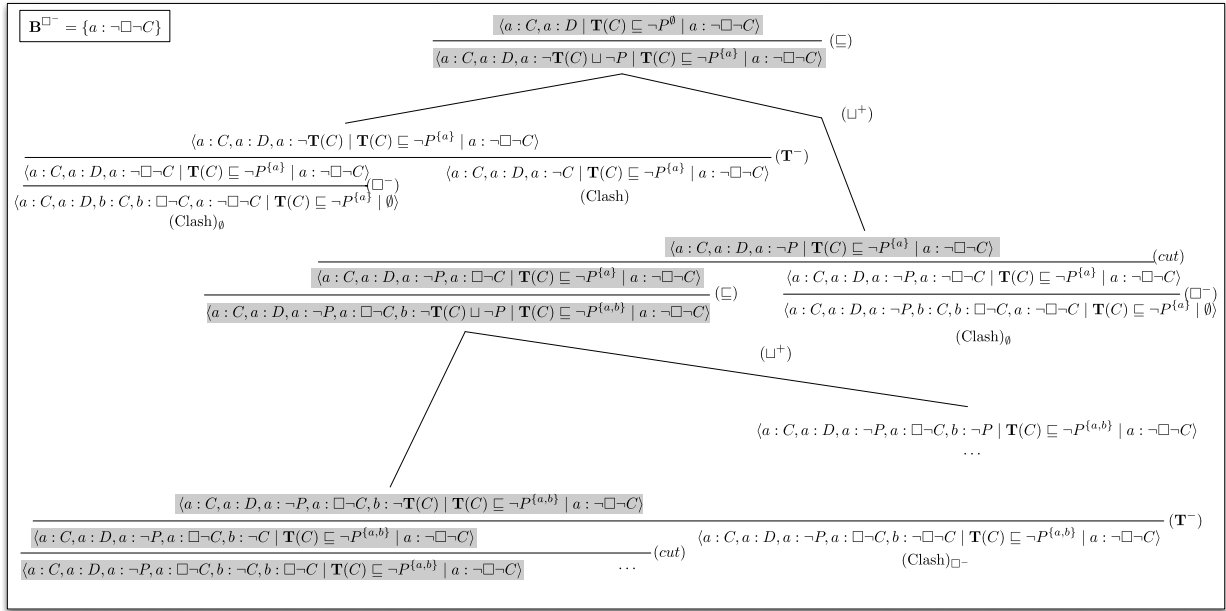
Fig. 4. A tableau in $\mathcal{TAB}_{PH2}^{\mathcal{ALC}+\mathbf{T}}$ used in order to check whether KB $\models_{min}^{\mathcal{L}_{\mathbf{T}}} \neg P(a)$, where KB $= \mathbf{T}(C) \sqsubseteq \neg P, C(a), D(a)$ (phase 2).

- *KB satisfiability in $\mathcal{ALC}+\mathbf{T}_{min}$*: KB is satisfiable in $\mathcal{ALC}+\mathbf{T}_{min}$ if there exists a minimal model of KB with respect to $\mathcal{L}_{\mathbf{T}}$.

The first three reasoning problems above can be reduced to minimal entailment as follows:

- *Instance checking*: it can be easily observed that it corresponds to minimal entailment. Indeed, given an individual constant a occurring in ABox and an extended concept C, checking whether a is an instance of C with respect to KB corresponds to checking whether KB $\models_{min}^{\mathcal{L}_{\mathbf{T}}} C(a)$.
- *Subsumption*: we show that $C \sqsubseteq D$ with respect to KB if and only if
  - either $\mathcal{O}$ is empty and KB $\models_{min}^{\mathcal{L}_{\mathbf{T}}} \forall R.(\neg C \sqcup D)(c)$, where c is a new individual constant
  - or $\mathcal{O}$ is not empty (let $a \in \mathcal{O}$) and KB $\models_{min}^{\mathcal{L}_{\mathbf{T}}} \forall R.(\neg C \sqcup D)(a)$,

  where R is a role not occurring in the KB.
  First, it can be easily observed that if $C^I \subseteq D^I$ in all minimal models of KB with respect to $\mathcal{L}_{\mathbf{T}}$, then for all possible values of c^I, and R^I, $c^I \in \forall R.(\neg C \sqcup D)^I$ (respectively $a^I \in \forall R.(\neg C \sqcup D)^I$). Hence KB $\models_{min}^{\mathcal{L}_{\mathbf{T}}} \forall R.(\neg C \sqcup D)(c)$ and KB $\models_{min}^{\mathcal{L}_{\mathbf{T}}} \forall R.(\neg C \sqcup D)(a)$. On the other hand, we proceed by contraposition. If $C \not\sqsubseteq D$ with respect to KB, then there is a minimal model $\mathcal{M}$ of KB and a domain element y such that $y \in (C \sqcap \neg D)^I$. If $\mathcal{O} = \emptyset$, from $\mathcal{M}$ we can easily build a model $\mathcal{M}'$ by extending I such that $c^I = y$ and $(c^I, y) \in R^I$. Since neither R nor c occur in the KB, this extension of I is well behaved, and since $\mathcal{O} = \emptyset$, I satisfies the unique name assumption. Furthermore, since $\mathcal{M}$ is a minimal model of KB with respect to $\mathcal{L}_{\mathbf{T}}$, also $\mathcal{M}'$ is. Indeed, the boxed formulas holding in the two models are the same. Furthermore, $\mathcal{M}'$ does not satisfy $\forall R.(\neg C \sqcup D)(c)$, hence we conclude that KB $\not\models_{min}^{\mathcal{L}_{\mathbf{T}}} \forall R.(\neg C \sqcup D)(c)$. If $\mathcal{O} \neq \emptyset$ we build $\mathcal{M}'$ by letting $(a^I, y) \in R^I$, and we reason in the same way.
- *Concept satisfiability*: we can easily observe that checking whether a concept C is satisfiable with respect to KB corresponds to verifying that C is *not* subsumed by $\bot$ with respect to KB. Given the above reduction of subsumption to minimal entailment, we can reduce concept satisfiability as follows[9]:
  - either $\mathcal{O}$ is empty and KB $\not\models_{min}^{\mathcal{L}_{\mathbf{T}}} \forall R.(\neg C)(c)$, where R and c do not occur in KB
  - or $\mathcal{O}$ is not empty (let $a \in \mathcal{O}$) and KB $\not\models_{min}^{\mathcal{L}_{\mathbf{T}}} \forall R.(\neg C)(a)$.

The above reductions can be adopted to use the calculus $\mathcal{TAB}_{min}^{\mathcal{ALC}+\mathbf{T}}$ to deal with different reasoning problems. For instance, if we want to check whether a concept C is satisfiable with respect to a given KB, we can start the calculus on the constraint system corresponding to KB $\cup \{\neg\forall R.(\neg C)(c)\}$ (if $\mathcal{O}$ is empty) or to KB $\cup \{\neg\forall R.(\neg C)(a)\}$ (if $\mathcal{O}$ is not empty, and a is any element in $\mathcal{O}$) and see whether the calculus outputs NO. Moreover, by the above reductions we can obtain the following complexity upper bounds for the respective reasoning problems, namely:

---

[9] Corresponding to the reduction of subsumption $C \sqsubseteq \bot$ to minimal entailment.

Theorem 15 (*Complexity of reasoning problems in $\mathcal{ALC} + \mathbf{T}_{min}$*). *The problem of instance checking and the problem of subsumption in $\mathcal{ALC} + \mathbf{T}_{min}$ are in* CO-NEXP^{NP}. *The problem of concept satisfiability in $\mathcal{ALC} + \mathbf{T}_{min}$ is in* NEXP^{NP}.

Let us now conclude this section by taking into account the problem of KB satisfiability. First of all, we can prove the following theorem:

Theorem 16. *Given a KB, if there exists a model $\mathcal{M}$ satisfying KB in the sense of Definition 4, then there exists also a model $\mathcal{M}'$ which is a* minimal *model of KB in the sense of Definition 7.*

Proof. The theorem immediately follows the fact that $\mathcal{ALC} + \mathbf{T}_{min}$ has the finite model property (Theorem 14). Since there are finitely many finite models, and since the preference relation $<_{\mathcal{L}_{\mathbf{T}}}$ among models of Definition 7 obviously does not allow loops, then there exists a minimal such model. $\quad\square$

By Theorem 16, we immediately observe that the problem of checking whether a KB is satisfiable can be reduced to checking whether KB has a model, that is to say the problem of checking the satisfiability of a KB in $\mathcal{ALC} + \mathbf{T}$, which is known to be EXPTIME complete [26]. Therefore, we can conclude that:

Theorem 17 (*Complexity of KB satisfiability in $\mathcal{ALC} + \mathbf{T}_{min}$*). *The problem of* KB *satisfiability in $\mathcal{ALC} + \mathbf{T}_{min}$ is* EXPTIME *complete.*

7. Related works

Several non-monotonic extensions of DLs have been proposed in the literature. In Section 4 we have already discussed about the extensions of DLs with circumscription. In the following Section 7.1, we try to summarize the other main approaches proposed in the literature, leaving to Section 5.1 in [26] for a more detailed discussion. We start with approaches based on default logic [49]. In Section 7.2 we discuss on the alternative of adopting the rational logic **R** (rather than the Preferential **P**) in order to define the semantics of the typicality operator **T**.

7.1. Non-monotonic extensions of DLs in the literature

7.1.1. DLs and default logic

The work [3] proposes an extension of DL with Reiter's default logic. Intuitively, in this setting, a KB comprises, in addition to TBox and ABox, a finite set of default rules whose prerequisites, justifications, and consequents are concepts. Default rules are used in order to formalize prototypical properties. It is shown that, when default rules are only applied to individuals explicitly mentioned in the ABox, methods by Junker and Konolige [39] and by Schwind and Risch [51] can be applied in order to compute all the extensions. However, the same authors have pointed out that this integration may lead to both semantical and computational difficulties, both caused by an unsatisfactory treatment of open defaults via Skolemization. Skolemization of the ABox and of the consequents of default rules is needed in order to capture some intuitive inferences. The treatment of open defaults via Skolemization may also lead to an undecidable default consequence relation, even if the underlying logic is decidable. For this reason, [3] proposes a restricted semantics for open default theories, in which default rules are only applied to individuals explicitly mentioned in the ABox.

The extension of DLs with Reiter's defaults, even if restricted to explicitly mentioned individuals, inherits from general default logic the difficulty of modeling inheritance with exceptions giving precedence to more specific defaults in a direct way. This behavior appears to be more problematic in a DL framework where the emphasis lies on the hierarchical organization of the concepts. To attack this problem, one has to impose priorities on default application or to find a smarter (but ad hoc) encoding of defaults giving priority to more specific information. This has motivated the study of extensions of DLs with prioritized defaults [52,4]. To give a brief account, in [52] the author introduces an extension of DLs to perform default inheritance reasoning, a kind of default reasoning specifically tailored to reason in presence of a taxonomy of concepts. Specificity is handled by defining, for a given KB = (TBox, ABox) and an individual constant a occurring in the ABox, a preference relation over atomic concepts. The problem of specificity ia also addressed in [4]. As a difference with [52], priorities between defaults are induced by the position of their prerequisites in the concept hierarchy of the TBox, then the specificity is not determined by the defaults. A method for computing extensions is also proposed. However, as for the proposal in [3], in order to avoid semantical and computational difficulties due to the treatment of open defaults via Skolemization, all these approaches adopt a semantics in which defaults are only applied to individuals explicitly mentioned in the ABox, thus introducing an asymmetric treatment of domain elements.

7.1.2. DLs with epistemic operators

An alternative approach is undertaken in [21], where Description Logics of minimal knowledge and negation as failure are proposed by augmenting DLs with two epistemic operators, K and A, interpreted according to Lifschitz's non-monotonic logic MKNF [44,45]. In particular, [21] studies the extension of $\mathcal{ALC}$, called $\mathcal{ALCK}_{\mathcal{NF}}$, which allows to capture Reiter's default logic, integrity constraints, procedural rules as well as role and concept closure. The paper provides a sound, complete and

terminating tableau calculus for checking satisfiability of *simple* $\mathcal{ALCK}_{\mathcal{NF}}$ KBs, where in a simple KB the occurrences of the operator K within the scope of quantifiers are limited. The calculus uses triple exponential time in the size of the KB. For MKNF-DLs without quantifying-in (i.e., with no occurrences of epistemic operators in the scope of quantifiers), a general deductive method can be defined (see [20]), which is parametric with respect to the underlying DL. The authors prove that the problem of instance checking in a MKNF-DL without quantifying-in is decidable if and only if the problem of instance checking in the underlying DL is decidable. In particular, for the logic $\mathcal{ALCK}_{\mathcal{NF}}$ without quantifying-in the problem of instance checking is EXPTIME-complete as in the non-modal case. [40] extends the work in [21] by providing a translation of an $\mathcal{ALCK}_{\mathcal{NF}}$ KB to an equivalent *flat* KB and by defining a simplified tableau algorithm for flat KBs, which includes an optimized minimality check.

In both [21] and [20], the domain of epistemic interpretations is assumed to be countably-infinite and to be the same for all interpretations. Although this assumption restricts the semantics of first-order MKNF, nevertheless it allows an encoding of prerequisite-free defaults with an open semantics. [21] also provides an encoding of closed defaults by translating them into simple $\mathcal{ALCK}_{\mathcal{NF}}$ inclusions. [47] introduces the formalism of *MKNF$^+$* knowledge bases, which allows for a flexible integration of DLs and Answer Set Programming. The semantics of the formalisms, based on the logic of MKNF, overcomes the discrepancy between the open world assumption of DLs and the closed world assumption of rules.

A recent line of research on integrating DLs and logic programming rules introduces further non-monotonic extensions of DLs via negation-as-failure. Some approaches [22] introduce a loosely coupled integration of logic programs and DLs where the interpretations of terminological knowledge are not restricted, while logic program variables range over the set of constant symbols. Therefore this approach is similar to the classical extensions of DLs based on defaults: the non-monotonic inferences induced by program rules are limited to named individuals only. A common limitation of the non-monotonic extensions of DLs based on minimal knowledge and negation as failure (including the integrations of DLs and rules) is that they provide no support for specificity nor priorities.

7.1.3. DLs with circumscription

Circumscribed knowledge bases as presented in [10] are described above in Section 4. In [10] the authors provide decidability and complexity results based on theoretical analysis. They show that reasoning is decidable under the restriction that only concepts can be circumscribed, whereas roles have to vary during circumscription. This also holds for expressive DLs such as $\mathcal{ALCIO}$ and $\mathcal{ALCQO}$. Allowing roles to be fixed during minimization leads to an undecidability results even in the extension of basic $\mathcal{ALC}$.

In [5], the authors analyze the complexity of reasoning with circumscribed KBs by focusing their attention on low-complexity DLs. In detail, it is shown that reasoning in circumscribed DL-lite$_R$ [15,16], as well as in *left local* fragment of $\mathcal{EL}^{\perp}$ [1], is in the second level of the polynomial hierarchy, whereas reasoning in general circumscribed $\mathcal{EL}$ KBs remains ExpTime-hard. Further results are provided in [8], in particular matching lower complexity bounds are given. Moreover, the proposed framework is extended to more general queries, as well as to defeasible inclusions $C \sqsubseteq_n D$, where C is a compound concept. A generalization of specificity-based priorities introduced in [5] is also allowed by means of explicit priorities over defeasible inclusions. Finally, the left local fragment under consideration is obtained by means of a weaker restriction, namely a more liberal use of existential restrictions and terminologies is allowed. Furthermore, in [6] a fragment of $\mathcal{EL}^{\perp}$ has been identified for which the complexity of circumscribed KBs is polynomial. In [7] such extension is generalized to the logic $\mathcal{EL}^{++}$ without role composition.

As in our approach, the extension of DLs with circumscription avoids the restriction of reasoning about elements explicitly mentioned in the ABox. [42] defines a variant of circumscribed knowledge bases [10] for extending DLs with *local closed world* capabilities, to provide a knowledge representation language combining open and closed world reasoning. In this approach, the extensions of minimized predicates can only contain the domain elements representing individual constants explicitly mentioned in the ABox. Decidability of the language can be proved also for expressive DLs, even when minimization of roles is allowed.

Other early approaches to non-monotonic extensions of DLs are based on circumscription. A non-monotonic semantics based on circumscription is applied to a frame system in [11], however decidability and complexity of reasoning tasks are not provided. Circumscription has been also applied to a fragment of the DL $\mathcal{ALE}$ in [14]. This approach is similar to the one proposed in [10], however only non-prioritized circumscription is considered. Complexity results are also provided, namely it is shown that reasoning in the proposed non-monotonic $\mathcal{ALE}$ is in Π_2^p.

7.1.4. Relation with rational closure and KLM

In [17] a non-monotonic extension of $\mathcal{ALC}$ is proposed. This approach is based on the application of Lehmann and Magidor's *rational closure* [43] to $\mathcal{ALC}$, intuitively by the introduction of a consequence relation $\mathrel{|\!\sim}$ among concepts and of a consequence relation $\Vdash$ among an unfoldable KB and assertions. The authors show that such consequence relations are *rational*. It is also shown that such relations inherit the same computational complexity of the underlying DL. In a subsequent work [18], the authors introduce an approach based on the combination of rational closure and *Defeasible Inheritance Networks* (INs), in order to tackle the main weaknesses of both of them: on the one hand, rational closure has limited inference capabilities, for instance it does not allow an exceptional class not to inherit any of the typical properties of its superclasses; on the other hand, INs present some controversial logical properties. More precisely, the authors introduce a reasoning mechanism for INs relying on a procedure for rational closure; such a mechanism is then adopted in order

to define a boolean extension of INs, called *Boolean defeasible Inheritance Networks* (BINs). BINs are then used as the base to develop a defeasible propositional logic, which is also applied to the case of defeasible inheritance-based Description Logics. In this respect, as we have done in this work, the authors focus on the basic $\mathcal{ALC}$, and show that the resulting non-monotonic DL is characterized by all the desired logical properties of rational closure. The proposed mechanism only requires the existence of a decision procedure of classical entailment: therefore, it can be implemented on the top of existing propositional SAT solvers as well as DL reasoners.

An approach similar to the one of $\mathcal{ALC} + \mathbf{T}$ is proposed in [12]. Such an approach is based on the fact that some individuals in the domain are more typical than others, namely x is more typical than y if $x \geqslant y$, where the relation $\geqslant$ is *modular* as in KLM rational logic **R**. A defeasible inclusion $C \sqsubset_D D$ holds if the most preferred (typical) Cs with respect to $\geqslant$ are also Ds.

We have already mentioned that the semantics of the typicality operator $\mathbf{T}$ of Definition 1 is strongly related with the semantics of non-monotonic entailment in KLM preferential logic **P**. In Section 5.2 of [26] a precise relation between KLM logic **P** and the Description Logic $\mathcal{ALC} + \mathbf{T}$ is provided. In [30], a non-monotonic extension of logic **P** called $\mathbf{P}_{min}$ is proposed. $\mathbf{P}_{min}$ is based on the same idea of the non-monotonic logic $\mathcal{ALC} + \mathbf{T}_{min}$ presented here, that is to say a minimal model approach based on the restriction to models that contain as little as possible of atypical (or non minimal) worlds. More in detail, given a modal interpretation of a minimal A-world as $A \wedge \Box \neg A$, the intuition is that preferred, or minimal models are those that minimize the number of worlds where $\neg \Box \neg A$ holds, that is of A-worlds which are not minimal. Furthermore, in [30] a decision procedure for checking satisfiability and validity in $\mathbf{P}_{min}$ is provided. This decision procedure has the form of a tableau calculus, with a two-step construction, similar to the procedure $\mathcal{TAB}^{\mathcal{ALC}+\mathbf{T}}_{min}$ presented in Section 5. This procedure is used to determine an upper bound of the complexity of $\mathbf{P}_{min}$, in particular it is shown that checking entailment for $\mathbf{P}_{min}$ is in Π_2, thus it has the same complexity as standard non-monotonic (skeptical) mechanisms.

7.2. Rational vs. preferential DLs

The family of KLM logics contains other interesting members, notably the stronger logic **R**, known as Rational Preferential Logic. This system is obtained by adding to **P** the axiom/rule of rational monotonicity:

$$A \mathrel{|\!\sim} C \wedge \neg(A \mathrel{|\!\sim} \neg B) \to \big((A \wedge B) \mathrel{|\!\sim} C\big)$$

That is to say, from $A \mathrel{|\!\sim} C$ we can conclude $(A \wedge B) \mathrel{|\!\sim} C$ unless we can derive $A \mathrel{|\!\sim} \neg B$. For a discussion and a justification of this property we refer to the literature [43]. The semantics of rational logic **R** is well-understood: the rational monotonicity principle corresponds to the additional property of *modularity* of the preference relation. In [31,32], we have investigated the properties characterizing the semantics of the **T** operator in $\mathcal{ALC} + \mathbf{T}$ as compared with the properties that would result for **T** if we adopted the stronger logic of non-monotonic entailment **R**. More precisely, we have added to the conditions for $f_{\mathbf{T}}$ in Definition 1 the following condition of Rational Monotonicity:

$$(f_{\mathbf{T}} - 6) \quad \text{if } f_{\mathbf{T}}(S) \cap R \neq \emptyset, \quad \text{then } f_{\mathbf{T}}(S \cap R) \subseteq f_{\mathbf{T}}(S)$$

obtaining a stronger DL based on Rational Entailment. $(f_{\mathbf{T}} - 6)$ forces again a form of monotonicity: if there is a typical S having the property R, then all typical S and Rs inherit the properties of typical Ss. We call $\mathcal{ALC} + \mathbf{T_R}$ the logic resulting from the addition of $(f_{\mathbf{T}} - 6)$ to the properties $(f_{\mathbf{T}} - 1) - (f_{\mathbf{T}} - 5)$. As for the logic $\mathcal{ALC} + \mathbf{T}$, the semantics of $\mathcal{ALC} + \mathbf{T_R}$ can be formulated in terms of possible world structures $\langle \Delta, I, < \rangle$ in which $<$ is *modular*, i.e. for each x, y, z, if $x < y$, then either $z < y$ or $x < z$.

In [32] it is shown that the following facts hold in $\mathcal{ALC} + \mathbf{T_R}$:

(i) $\neg(\mathbf{T}(A) \sqcap B \sqsubseteq \bot)$ implies $\mathbf{T}(A \sqcap B) \sqsubseteq \mathbf{T}(A)$
(ii) $\neg(\mathbf{T}(A) \sqcap B \sqsubseteq \bot)$ implies $\mathbf{T}(B) \sqcap A \sqsubseteq \mathbf{T}(A)$

Both properties allow us to draw conclusions from the simple fact that there is *one* individual that (1) is a typical instance of the concept A and that (2) is an instance of concept B. From (i), we derive that *all* typical A and Bs are typical As. From (ii) we derive something about typical Bs, even if A and B are unrelated properties. In particular, we derive that typical Bs that are also instances of concept A are typical As.

More in detail, from (ii) we derive the following counterintuitive example, where from an empty TBox and an ABox containing the following facts:

(a) $\mathbf{T}(Brilliant)(john)$
(b) $Writer(john)$
(c) $\neg\mathbf{T}(Writer)(john)$

we can then conclude that

(d) $\mathbf{T}(Writer) \sqsubseteq \neg Brilliant$

Indeed, from the ABox we can first obtain that $\mathbf{T}(Brilliant) \sqcap Writer \not\sqsubseteq \mathbf{T}(Writer)$, then, by making the contrapositive of (ii), we get $\mathbf{T}(Writer) \sqcap Brilliant \sqsubseteq \bot$, from which we can immediately conclude (d) $\mathbf{T}(Writer) \sqsubseteq \neg Brilliant$.

As a further example, given the following ABox:

> $\mathbf{T}(Graduated)(andras)$
>
> $SoccerPlayer(andras)$
>
> $\mathbf{T}(SoccerPlayer)(lilian)$
>
> $Graduated(lilian)$

and an empty TBox, we can get that:

> $\mathbf{T}(SoccerPlayer)(andras)$

which does not make sense given that *lilian* is a different person not related to *andras*, hence we do not want to use *lilian*'s properties to make inferences about *andras*.

In our opinion, some of the inferences in $\mathcal{ALC} + \mathbf{T_R}$ are rather arbitrary (or, at least, controversial) and counterintuitive, therefore we believe that the logic $\mathbf{R}$ is too strong and unsuitable to reason about typicality.

In [32] we have also shown that the logic $\mathcal{ALC} + \mathbf{T_R}$ is equivalent to the logic for defeasible subsumptions in DLs proposed by [12], when considered with $\mathcal{ALC}$ as the underlying DL. The properties of $\geqslant$ in [12] correspond to those of $<$ in $\mathcal{ALC} + \mathbf{T_R}$. At a syntactic level the two logics differ, so that in [12] one finds the defeasible inclusions $C \sqsubset_D D$ instead of $\mathbf{T}(C) \sqsubseteq D$ of $\mathcal{ALC} + \mathbf{T_R}$. However, the intuition in the two cases is similar: the inclusion holds if the most preferred (typical) Cs are also Ds. Indeed, in [32] it is shown that the logic of preferential subsumption can be translated into $\mathcal{ALC} + \mathbf{T_R}$ by replacing $C \sqsubset_D D$ with $\mathbf{T}(C) \sqsubseteq D$. The approach in [12], therefore, inherits the above criticisms for extensions of DLs that use $\mathbf{R}$.

8. Conclusions and further research

In this work, we have proposed $\mathcal{ALC} + \mathbf{T}_{min}$, a non-monotonic extension of $\mathcal{ALC}$ for reasoning about prototypical properties in Description Logic framework. The extension is obtained by adding first a typicality operator, originally defined in [26], to $\mathcal{ALC}$. The typicality operator is characterized by a set of postulates which are essentially the same of KLM preferential logic $\mathbf{P}$. This extension, called $\mathcal{ALC} + \mathbf{T}$, provides a monotonic extension of $\mathcal{ALC}$ that enjoys a simple modal semantics. One advantage of the use of a typicality operator is that we can express prototypical properties easily and directly in the form "the most typical instances of concept C are instances of concept P" (corresponding to $\mathbf{T}(C) \sqsubseteq P$). However, $\mathcal{ALC} + \mathbf{T}$ is not sufficient to perform defeasible reasoning. For this reason, in the present work we have developed a preferential semantics, called $\mathcal{ALC} + \mathbf{T}_{min}$. This non-monotonic extension of $\mathcal{ALC} + \mathbf{T}$ allows to perform defeasible reasoning in particular in the context of inheritance with exceptions. We have then developed a procedure for deciding query-entailment in $\mathcal{ALC} + \mathbf{T}_{min}$. The procedure has the form of a two-phase tableau calculus for generating $\mathcal{ALC} + \mathbf{T}_{min}$ minimal models. The procedure is sound, complete, and terminating, whereby giving a decision procedure for deciding $\mathcal{ALC} + \mathbf{T}_{min}$ entailment in co-NExp^{NP}. We have also considered other reasoning problems in DLs, namely instance checking, subsumption, concept satisfiability and KB satisfiability. For the first three problems, we show that they can be reduced to minimal entailment. This allows to obtain complexity upper bounds for such problems, namely that instance checking and subsumption for $\mathcal{ALC} + \mathbf{T}_{min}$ are in co-NExp^{NP} and that concept satisfiability for $\mathcal{ALC} + \mathbf{T}_{min}$ is in NExp^{NP}. Concerning KB satisfiability, we show that it is EXPTIME complete.

We plan to extend the work presented in this paper in several directions. First of all, the tableau procedure we have described in Section 5 can be optimized in many ways. For instance, we guess that the calculus $\mathcal{TAB}_{PH1}^{\mathcal{ALC}+\mathbf{T}}$, dealing with the monotonic logic $\mathcal{ALC} + \mathbf{T}$, can be made more efficient by applying standard techniques such as caching, in order to obtain an ExpTime decision procedure for $\mathcal{ALC} + \mathbf{T}$.

From the point of view of knowledge representation, a limit of our logic is the inability to handle inheritance of multiple properties in case of exceptions as in the example:

> $\mathbf{T}(Student) \sqsubseteq \neg HasIncome$
>
> $\mathbf{T}(Student) \sqsubseteq \exists Owns.LibraryCard$
>
> $PhDStudent \sqsubseteq Student$
>
> $\mathbf{T}(PhDStudent) \sqsubseteq HasIncome$

Our semantics does not support the inference

> $\mathbf{T}(PhDStudent) \sqsubseteq \exists Owns.LibraryCard$

that is, PhD students typically own a library card, as we might want to conclude (since having an income has nothing to do with owning a library card). The reason why our semantics fails to support this inference is that the first two

inclusions are obviously equivalent to the single one $\mathbf{T}(Student) \sqsubseteq \neg HasIncome \sqcap \exists Owns.LibraryCard$ which is contradicted by $\mathbf{T}(PhDStudent) \sqsubseteq HasIncome$. As already mentioned in the Introduction, to handle this type of inferences we would need a tighter semantics where the truth of $\mathbf{T}(C) \sqsubseteq P$ is no longer a function of $\mathbf{T}(C)$ and P or a smarter (and less direct) encoding of the knowledge. This problem is perhaps better addressed by probabilistic extensions of Description Logics such as [37].

KLM logics, which are at the basis of our semantics, are related to probabilistic reasoning. In [37], the notion of conditional constraint allows typicality assertions to be expressed (with a specified interval of probability values). In order to perform defeasible reasoning, a notion of minimal entailment is introduced based on a *lexicographic preference* relation on probabilistic interpretations. We plan to compare in detail this probabilistic approach to ours in further research.

We aim to extend our minimal model semantics to other Description Logics, taking into account both more expressive and less expressive DLs. Concerning low-complexity DLs, preliminary results are given in [35], where we have considered the minimal model semantics applied to the low complexity logics DL-lite$_R$ and $\mathcal{EL}^\perp$. We have studied the complexity of the resulting logics $\mathcal{EL}^\perp\mathbf{T}_{min}$ and DL-lite$_R\mathbf{T}_{min}$. For $\mathcal{EL}^\perp$, we have shown that its extension $\mathcal{EL}^\perp\mathbf{T}_{min}$ is unfortunately ExpTime-hard. However, we have shown that the complexity decreases to Π_2^p for the fragment of *Left Local $\mathcal{EL}^\perp$* KBs. We have also obtained the same complexity upper bound Π_2^p for the logic DL-lite$_R\mathbf{T}_{min}$. These results match the complexity upper bounds of the same fragments in circumscribed KBs [5]. Concerning more expressive DLs, we intend to study in a systematic way how the $\mathbf{T}$ operator and the minimal model semantics can be applied to extensions of $\mathcal{ALC}$, including number restrictions, inverse roles and role hierarchies. Some preliminary results concerning the logics $\mathcal{ALCN}$ and $\mathcal{ALCQ}$, extending $\mathcal{ALC}$ with (qualified) number restrictions, have been introduced in [50].

Last, we intend to explore alternative notions of preference among models, which possibly generalize the notion considered in this paper. Preliminary results in this direction are contained in [36], where we define a general framework for non-monotonic reasoning based on a different notion of minimal model. We show that, under certain conditions, this semantics can capture the well-known construction of *rational closure*. As mentioned above, a non-monotonic extension of DL based on rational closure has been studied in [18]. We thus think that the minimal model semantics proposed in [36] may be of interest to define useful non-monotonic DLs, corresponding to rational closure and possible variants.

Acknowledgements

References

[1] F. Baader, S. Brandt, C. Lutz, Pushing the $\mathcal{EL}$ envelope, in: L. Kaelbling, A. Saffiotti (Eds.), Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI 2005), Edinburgh, Scotland, UK, Professional Book Center, 2005, pp. 364–369.

[2] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, P. Patel-Schneider, The Description Logic Handbook – Theory, Implementation, and Applications, 2nd edition, Cambridge, 2007.

[3] F. Baader, B. Hollunder, Embedding defaults into terminological knowledge representation formalisms, Journal of Automated Reasoning (JAR) 14 (1) (1995) 149–180.

[4] F. Baader, B. Hollunder, Priorities on defaults with prerequisites, and their application in treating specificity in terminological default logic, Journal of Automated Reasoning (JAR) 15 (1) (1995) 41–68.

[5] P. Bonatti, M. Faella, L. Sauro, Defeasible inclusions in low-complexity DLs: Preliminary notes, in: C. Boutilier (Ed.), Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI 2009), Pasadena, California, USA, Morgan Kaufmann, 2009, pp. 696–701.

[6] P. Bonatti, M. Faella, L. Sauro, $\mathcal{EL}$ with default attributes and overriding, in: P. Patel-Schneider, Y. Pan, P. Hitzler, P. Mika, L. Zhang, J.Z. Pan, I. Horrocks, B. Glimm (Eds.), 9th International Semantic Web Conference (ISWC 2010), Shanghai, China, in: Lecture Notes in Computer Science (LNCS), vol. 6496, Springer, 2010, pp. 64–79.

[7] P.A. Bonatti, M. Faella, L. Sauro, Adding default attributes to $\mathcal{EL}$++, in: W. Burgard, D. Roth (Eds.), Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence (AAAI 2011), San Francisco, California, USA, AAAI Press, 2011.

[8] P.A. Bonatti, M. Faella, L. Sauro, On the complexity of $\mathcal{EL}$ with defeasible inclusions, in: T. Walsh (Ed.), Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI 2011), Barcelona, Spain, Morgan Kaufmann, 2011, pp. 762–767.

[9] P.A. Bonatti, C. Lutz, F. Wolter, Description logics with circumscription, in: P. Doherty, J. Mylopoulos, C.A. Welty (Eds.), Proceedings of the Tenth International Conference on Principles of Knowledge Representation and Reasoning (KR 2006), Lake District of the United Kingdom, AAAI Press, 2006, pp. 400–410.

[10] P.A. Bonatti, C. Lutz, F. Wolter, The complexity of circumscription in DLs, Journal of Artificial Intelligence Research (JAIR) 35 (2009) 717–773.

[11] G. Brewka, The logic of inheritance in frame systems, in: J. McDermott (Ed.), Proceedings of the 10th International Joint Conference on Artificial Intelligence (IJCAI 1987), Milan, Italy, Morgan Kaufmann, 1987, pp. 483–488.

[12] K. Britz, J. Heidema, T. Meyer, Semantic preferential subsumption, in: G. Brewka, J. Lang (Eds.), Principles of Knowledge Representation and Reasoning: Proceedings of the Eleventh International Conference (KR 2008), Sidney, Australia, AAAI Press, 2008, pp. 476–484.

[13] M. Buchheit, F.M. Donini, A. Schaerf, Decidable reasoning in terminological knowledge representation systems, Journal of Artificial Intelligence Research (JAIR) 1 (1993) 109–138.

[14] M. Cadoli, F. Donini, M. Schaerf, Closed world reasoning in hybrid systems, in: Proceedings of the 6th International Symposium on Methodologies for Intelligent Systems (ISMIS-90), North-Holland, 1990, pp. 474–481.

[15] D. Calvanese, G.D. Giacomo, D. Lembo, M. Lenzerini, R. Rosati, DL-Lite: Tractable description logics for ontologies, in: M. Veloso, S. Kambhampati (Eds.), Proceedings of the 20th National Conference on Artificial Intelligence and the 17th Innovative Applications of Artificial Intelligence Conference, Pittsburgh, Pennsylvania, USA, AAAI Press/The MIT Press, 2005, pp. 602–607.

[16] D. Calvanese, G.D. Giacomo, D. Lembo, M. Lenzerini, R. Rosati, Tractable reasoning and efficient query answering in description logics: The *DL-Lite* family, Journal of Automated Reasoning (JAR) 39 (3) (2007) 385–429.

[17] G. Casini, U. Straccia, Rational closure for defeasible description logics, in: T. Janhunen, I. Niemelä (Eds.), Proceedings of the 12th European Conference on Logics in Artificial Intelligence (JELIA 2010), Helsinki, Finland, in: Lecture Notes in Artificial Intelligence (LNAI), vol. 6341, Springer, 2010, pp. 77–90.

[18] G. Casini, U. Straccia, Defeasible inheritance-based description logics, in: T. Walsh (Ed.), Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI 2011), Barcelona, Spain, Morgan Kaufmann, 2011, pp. 813–818.

[19] F.M. Donini, M. Lenzerini, D. Nardi, W. Nutt, A. Schaerf, An epistemic operator for description logics, Artificial Intelligence 100 (1–2) (1998) 225–274.

[20] F.M. Donini, D. Nardi, R. Rosati, Autoepistemic description logics, in: M. Georgeff, M. Pollack (Eds.), Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI 1997), vol. 1, Nagoya, Japan, Morgan Kaufmann, 1997, pp. 136–141.

[21] F.M. Donini, D. Nardi, R. Rosati, Description logics of minimal knowledge and negation as failure, ACM Transactions on Computational Logic (ToCL) 3 (2) (2002) 177–225.

[22] T. Eiter, T. Lukasiewicz, R. Schindlauer, H. Tompits, Combining answer set programming with description logics for the semantic web, in: D. Dubois, C. Welty, M. Williams (Eds.), Principles of Knowledge Representation and Reasoning: Proceedings of the 9th International Conference (KR 2004), Whistler, Canada, AAAI Press, 2004, pp. 141–151.

[23] D.W. Etherington, R. Reiter, On inheritance hierarchies with exceptions, in: AAAI, 1983, pp. 104–108.

[24] S.E. Fahlman, D.S. Touretzky, W. van Roggen, Cancellation in a parallel semantic network, in: P.J. Hayes (Ed.), Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI 1981), Vancouver, BC, Canada, William Kaufmann, 1981, pp. 257–263.

[25] L. Giordano, V. Gliozzi, N. Olivetti, G.L. Pozzato, Reasoning about typicality in preferential description logics, in: S. Hólldobler, C. Lutz, H. Wansing (Eds.), Proceedings of the 11th European Conference on Logics in Artificial Intelligence (JELIA 2008), in: Lecture Notes in Artificial Intelligence (LNAI), vol. 5293, Dresden, Germany, Springer-Verlag, 2008, pp. 192–205.

[26] L. Giordano, V. Gliozzi, N. Olivetti, G.L. Pozzato, $\mathcal{ALC} + T$: A preferential extension of Description Logics, Fundamenta Informaticae 96 (2009) 1–32.

[27] L. Giordano, V. Gliozzi, N. Olivetti, G.L. Pozzato, On extending description logics for reasoning about typicality: A first step, TR Dip. di Informatica, 2009, http://www.di.unito.it/~pozzato/tr09.pdf.

[28] L. Giordano, V. Gliozzi, N. Olivetti, G.L. Pozzato, Prototypical reasoning with low complexity description logics: Preliminary results, in: F. Lin, T. Schaub, E. Erdem (Eds.), Proceedings of 10th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR 2009), Potsdam, Germany, in: Lecture Notes in Artificial Intelligence (LNAI), vol. 5753, Springer-Verlag, 2009, pp. 430–436.

[29] L. Giordano, V. Gliozzi, N. Olivetti, G.L. Pozzato, Reasoning about typicality with low complexity description logics: The logic $\mathcal{EL}^{+\perp}\mathbf{T}$, in: R. Serra, R. Cucchiara (Eds.), AI*IA 2009: Emergent Perspectives in Artificial Intelligence: XI Conference of the Italian Association for Artificial Intelligence, Reggio Emilia, Italy, in: Lecture Notes in Artificial Intelligence (LNAI), vol. 5883, Springer-Verlag, 2009, pp. 62–71.

[30] L. Giordano, V. Gliozzi, N. Olivetti, G.L. Pozzato, A nonmonotonic extension of KLM preferential logic P, in: C.G. Fermüller, A. Voronkov (Eds.), Proceedings of the 17th Conference on Logic for Programming, Artificial Intelligence, and Reasoning (LPAR 2010), Yogyakarta, Indonesia, in: ARCoSS Lecture Notes in Computer Science (LNCS), vol. 6397, Springer-Verlag, 2010, pp. 317–332.

[31] L. Giordano, V. Gliozzi, N. Olivetti, G.L. Pozzato, Preferential vs rational description logics: Which one for reasoning about typicality? in: H. Coelho, R. Studer, M. Wooldridge (Eds.), Proceedings of the 19th European Conference on Artificial Intelligence (ECAI 2010), Lisbon, Portugal, in: Frontiers in Artificial Intelligence and Applications, vol. 215, IOS Press, 2010, pp. 1069–1070 (short paper).

[32] L. Giordano, V. Gliozzi, N. Olivetti, G.L. Pozzato, Reasoning about typicality in preferential description logics: Preferential vs rational entailment, in: W. Faber, N. Leone (Eds.), Proceedings of the 25th Convegno Italiano di Logica Computazionale (CILC 2010), in: CEUR Workshop Proceedings, CEUR-WS.org, vol. 598, Rende (CS), Italy, July 2010, pp. 1–15.

[33] L. Giordano, V. Gliozzi, N. Olivetti, G.L. Pozzato, A tableau calculus for a nonmonotonic extension of $\mathcal{EL}^{\perp}$, in: K. Brünnler, G. Metcalfe (Eds.), Proceedings of TABLEAUX 2011 (20th International Conference on Automated Reasoning with Analytic Tableaux and Related Methods), Bern, Switzerland, in: LNAI, vol. 6793, Springer-Verlag, 2011, pp. 180–195.

[34] L. Giordano, V. Gliozzi, N. Olivetti, G.L. Pozzato, A tableau calculus for a nonmonotonic extension of the Description Logic $DL - Lite_{core}$, in: R. Pirrone, F. Sorbello (Eds.), AI*IA 2011: Artificial Intelligence Around Man and Beyond, XIIth International Conference of the Italian Association for Artificial Intelligence, Palermo, Italy, in: LNAI, vol. 6934, Springer-Verlag, 2011, pp. 164–176.

[35] L. Giordano, V. Gliozzi, N. Olivetti, G.L. Pozzato, Reasoning about typicality in low complexity DLs: The logics $\mathcal{EL}^{\perp}\mathbf{T}_{min}$ and DL-lite$_R\mathbf{T}_{min}$, in: T. Walsh (Ed.), Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI 2011), Barcelona, Spain, Morgan Kaufmann, 2011, pp. 894–899.

[36] L. Giordano, V. Gliozzi, N. Olivetti, G.L. Pozzato, A minimal model semantics for nonmonotonic reasoning, in: L. Fariñas del Cerro, J.M. Andreas Herzig (Eds.), Logics in Artificial Intelligence, 13th European Conference (JELIA 2012), Toulouse, France, in: LNAI, vol. 7519, Springer-Verlag, 2012, pp. 228–241.

[37] R. Giugno, T. Lukasiewicz, P-$\mathcal{SHOQ}$(D): A probabilistic extension of $\mathcal{SHOQ}$(D) for probabilistic ontologies in the semantic web, in: S. Flesca, S. Greco, N. Leone, G. Ianni (Eds.), Proceedings of 8th European Conference on Logics in Artificial Intelligence (JELIA 2002), in: Lecture Notes in Artificial Intelligence (LNAI), vol. 2424, Springer-Verlag, 2002, pp. 86–97.

[38] S. Grimm, P. Hitzler, A preferential tableaux calculus for circumscriptive $\mathcal{ALCO}$, in: A. Polleres, T. Swift (Eds.), Proceedings of the 3rd International Conference on Web Reasoning and Rule Systems (RR 2009), Chantilly, VA, USA, in: Lecture Notes in Computer Science (LNCS), vol. 5837, Springer, 2009, pp. 40–54.

[39] U. Junker, K. Konolige, Computing the extensions of autoepistemic and default logics with a truth maintenance system, in: AAAI, 1990, pp. 278–283.

[40] P. Ke, U. Sattler, Next steps for description logics of minimal knowledge and negation as failure, in: F. Baader, C. Lutz, B. Motik (Eds.), Proceedings of the 21st International Workshop on Description Logics (DL2008), in: CEUR Workshop Proceedings, CEUR-WS.org, vol. 353, Dresden, Germany, May 2008.

[41] S. Kraus, D. Lehmann, M. Magidor, Nonmonotonic reasoning, preferential models and cumulative logics, Artificial Intelligence 44 (1–2) (1990) 167–207.

[42] A.A. Krisnadhi, K. Sengupta, P. Hitzler, Local closed world semantics: Keep it simple, stupid! in: Proceedings of the 24th International Workshop on Description Logics (DL 2011), in: CEUR Workshop Proceedings, vol. 745, Barcelona, Spain, July 2011.

[43] D. Lehmann, M. Magidor, What does a conditional knowledge base entail? Artificial Intelligence 55 (1) (1992) 1–60.

[44] V. Lifschitz, Nonmonotonic databases and epistemic queries, in: J. Mylopoulos, R. Reiter (Eds.), Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI 1991), Sidney, Australia, Morgan Kaufmann, 1991, pp. 381–386.

[45] V. Lifschitz, Minimal belief and negation as failure, Artificial Intelligence 70 (1–2) (1994) 53–72.

[46] J. McCarthy, Applications of circumscription to formalizing common-sense knowledge, Artificial Intelligence 28 (1) (1986) 89–116.

[47] B. Motik, R. Rosati, Reconciling description logics and rules, Journal of the ACM 57 (5) (2010).

[48] D. Nute, Topics in Conditional Logic, Reidel, Dordrecht, 1980.

[49] R. Reiter, A logic for default reasoning, Artificial Intelligence 13 (1–2) (1980) 81–132.

[50] D. Rognone, Non monotonic extensions of DLs: Reasoning about typicality with number restrictions, Master's thesis, Università degli Studi di Torino, 2009.

[51] C. Schwind, V. Risch, A tableau-based characterisation for default logic, in: ECSQARU, 1991, pp. 310–317.
[52] U. Straccia, Default inheritance reasoning in hybrid kl-one-style logics, in: R. Bajcsy (Ed.), Proceedings of the 13th International Joint Conference on Artificial Intelligence (IJCAI 1993), Chambéry, France, Morgan Kaufmann, 1993, pp. 676–681.