The Bayesian Description Logic $\mathcal{BALC}$

Leonard Botha[1], Thomas Meyer[2], and Rafael Peñaloza[3]

[1] University of Cape Town and CAIR, South Africa, leonardzbotha@gmail.com
[2] University of Cape Town and CAIR, South Africa, tmeyer@cs.uct.ac.za
[3] Free University of Bozen-Bolzano, Bolzano, Italy, rafael.penaloza@unibz.it

Abstract. Description Logics (DLs) that support uncertainty are not as well studied as their crisp alternatives, thereby limiting their use in real world domains. The Bayesian DL $\mathcal{BEL}$ and its extensions have been introduced to deal with uncertain knowledge without assuming (probabilistic) independence between axioms. In this paper we combine the classical DL $\mathcal{ALC}$ with Bayesian Networks. Our new DL includes a solution to the consistency checking problem and changes to the tableaux algorithm that are not a part of $\mathcal{BEL}$. Furthermore, $\mathcal{BALC}$ also supports probabilistic assertional information which was not studied for $\mathcal{BEL}$. We present algorithms for four categories of reasoning problems for our logic; two versions of concept satisfiability (referred to as *total concept satisfiability* and *partial concept satisfiability* respectively), knowledge base consistency, subsumption, and instance checking. We show that all reasoning problems in $\mathcal{BALC}$ are in the same complexity class as their classical variants, provided that the size of the Bayesian Network is included in the size of the knowledge base.

1 Introduction

Description Logics (DLs) [1] that support uncertainty are currently not as mature as their crisp alternatives. Furthermore, DLs capable of representing uncertain contextual knowledge are even more scarce. This lack of mature probabilistic reasoning services limits the application of DLs in many real world domains, which often require reasoning about uncertain or contradictory information. For example when planning free time activities, the weather is often a very real consideration. The kind of activities that are pleasant in poor weather are often very different from summer activities. As such, building an ontology that models activities will be quite difficult. However, modeling the weather as uncertain contexts simplifies this task. Consider for example the axiom

$$(Swimming \sqsubseteq Fun)^{Sunny} \quad (1)$$

which intuitively states that *Swimming* is *Fun* when the weather is *Sunny*. Importantly this axiom states nothing about the relation between swimming and fun when the context *sunny* does not hold. In other words, the axiom (1) expresses that if it is *Sunny* then *Swimming* is *Fun*. However, that is not to say that *Swimming* is *always* fun. By attaching probabilities to the contexts we are able to reason about how *probable* it is that an activity will be fun. Furthermore, if we make these probabilities conditional then we can use our knowledge of the world in these queries.

To move towards being able to perform this kind of reasoning we study the Bayesian Description Logic $\mathcal{BALC}$. $\mathcal{BALC}$ is a contextual Bayesian Description Logic based on the existing DL $\mathcal{BEL}$ and Bayesian ontology languages [5, 6]. Unlike many other probabilistic DLs (for a survey see [8]), Bayesian DLs do not directly encode the probabilities that concepts or roles are related. Instead axioms and assertions are annotated with an optional context in which they are required to hold. The probability of these contexts holding are then represented using a Bayesian Network (BN). This gives Bayesian DLs the ability to perform conditional probabilistic reasoning. In terms of the underlying logic, our approach is similar to [10]. However, contrary to [10], we do not assume independence between the different axioms, but rather describe their joint probability distribution with the help of the contextual knowledge.

2 $\mathcal{BALC}$

Bayesian networks (BNs) are graphical models capable of representing the joint probability distribution of several discrete random variables in a compact manner. Given a random variable X, we denote as $val(X)$ the set of values that X can take. Given an $x \in val(X)$, we denote as $X = x$ the *valuation* of X taking the value x. This notation is extended to sets of variables in the obvious way. Given a set of random variables V, a *world* ω is a set of valuations containing exactly one valuation for every random variable $X \in V$. A V*-literal* is an ordered pair of the form (X_i, x), where $X_i \in V$ and $x \in val(X_i)$. The name literal refers to them generalizing Boolean literals which are often denoted as x or $\neg x$ for the random variable X. For simplicity, in this paper we will often use the notation X for (X, T) and $\neg X$ for (X, F). A V*-context* is any set of V-literals. It is *consistent* if it contains at most one pair for each random variable. We will often call V-contexts *primitive contexts.*

Definition 1 (Bayesian Network). *A* Bayesian network *is a pair $\mathcal{B} = (G, \Theta)$ where $G = (V, E)$ is a directed acyclic graph and Θ is a set of conditional probability distributions for every variable $X \in V$ given its parents $\pi(X)$ on G:*

$$\Theta = \{P(X = x|\pi(X) = x') \mid X \in V\}.$$

We now describe $\mathcal{BALC}$ as an extension of the classical DL $\mathcal{ALC}$. The concept language for $\mathcal{BALC}$ is the same as for $\mathcal{ALC}$, but axioms are considered to hold only in a given context. This is expressed by annotations given to these axioms, as formalized next.

Definition 2 (KB). *Let V be a finite set of discrete random variables. A V-restricted general concept inclusion (V-GCI) is an expression of the form $(C \sqsubseteq D)^\kappa$ where C and D are $\mathcal{ALC}$ concepts and κ is a V-context. A V-TBox is* *a finite set of V-GCIs. A V-restricted assertion (V-assertion) is an expression of the form* $C(x)^\kappa$ *or* $r(x,y)^\kappa$ *where* C *is an* $\mathcal{ALC}$ *concept,* r *is an* $\mathcal{ALC}$ *role name,* x,y *are individual names, and* κ *is a V-context. A V-ABox is a finite set of V-assertions. A* $\mathcal{BALC}$ *knowledge base (KB) over* V *is a triple* $\mathcal{K} = (\mathcal{T}, \mathcal{A}, \mathcal{B})$ *where* $\mathcal{B}$ *is a BN over* V*,* $\mathcal{T}$ *is a V-TBox, and* $\mathcal{A}$ *is a V-ABox.*

Note that this definition does not prevent the encoding of classical statements. Axioms or assertions annotated with the empty set will hold in all contexts. We abbreviate $(C \sqsubseteq D)^\emptyset$ as $(C \sqsubseteq D)$ and $C(x)^\emptyset$ as $C(x)$. When it is clear from the context, we will omit the V prefix and refer only to literals, contexts, GCIs, assertions, ABoxes, and TBoxes.

As $\mathcal{BALC}$ is based on a model-theoretic semantics we next need to define what constitutes a model of a $\mathcal{BALC}$ knowledge base. In order to do this we first define two different types of interpretations; V-interpretations and probabilistic interpretations. V-interpretations should be thought of as an interpretation linked to a specific Bayesian world, while probabilistic interpretations are interpretations over all all worlds.

Definition 3 (V-interpretation). *A* V-interpretation *is a tuple* $\mathcal{V} = (\Delta^\mathcal{V}, \cdot^\mathcal{V}, v^\mathcal{V})$ *where* $\Delta^\mathcal{V}$ *is a non-empty set called the* domain*,* $v^\mathcal{V}$ *is a valuation function defined as* $v^\mathcal{V} : V \to \cup_{X \in V} val(X)$ *such that* $v^\mathcal{V}(X) \in val(X)$*, and* $\cdot^\mathcal{V}$ *is an interpretation function that maps every concept name* C *to a set* $C^\mathcal{V} \subseteq \Delta^\mathcal{V}$ *and every role name* r *to a binary relation* $r^\mathcal{V} \subseteq \Delta^\mathcal{V} \times \Delta^\mathcal{V}$*.The interpretation function* $v^\mathcal{V}$ *is extended to complex* $\mathcal{ALC}$ *concepts as usual.*

The V-interpretation $\mathcal{V}$ *is a* model *of the GCI* $(C \sqsubseteq D)^\kappa$*,* $(\mathcal{V} \models (C \sqsubseteq D)^\kappa)$*, iff (i)* $v^\mathcal{V} \not\models \kappa$*, or (ii)* $C^\mathcal{V} \subseteq D^\mathcal{V}$*.* $\mathcal{V}$ *is a* model *of the assertion* $C(x)^\kappa$ *(respectively* $r(x,y)^\kappa$*), denoted as* $\mathcal{V} \models C(x)^\kappa$ *(respectively* $\mathcal{V} \models r(x,y)^\kappa$*), iff (i)* $v^\mathcal{V} \not\models \kappa$*, or (ii)* $x^\mathcal{V} \in C^\mathcal{V}$ *(respectively* $(x^\mathcal{V}, y^\mathcal{V}) \in r^\mathcal{V}$*). It is a model of the TBox* $\mathcal{T}$ *(ABox* $\mathcal{A}$*) iff it is a* model *of all the GCIs in* $\mathcal{T}$ *(assertions in* $\mathcal{A}$*). It is a model of the knowledge base* $\mathcal{K}$ *iff it is a* model *of both* $\mathcal{T}$ *and* $\mathcal{A}$*.*

Given a valuation function $v^\mathcal{V}$, a Bayesian world ω, and a context κ we will denote $v^\mathcal{V} = \omega$ when a valuation function assigns each random variable the same value as it has in ω; $v^\mathcal{V} \models \kappa$ when for all $(X, x) \in \kappa$ we have that $v^\mathcal{V}(X) = x$; and $\omega \models \kappa$ when we have that $\omega = v^\mathcal{V}$ such that $v^\mathcal{V} \models \kappa$.

V-interpretations focus on only a single world, but a KB has information about the uncertainty of being in one world or another. Probabilistic interpretations combine multiple V-interpretations and the probability distribution from the BN.

Definition 4 (Probabilistic interpretation). *A probabilistic interpretation is a pair of the form* $\mathcal{P} = (\mathcal{J}, \mathcal{P}_\mathcal{J})$*, where* $\mathcal{J}$ *is a finite set of V-interpretations and* $\mathcal{P}_\mathcal{J}$ *is a probability distribution over* $\mathcal{J}$ *such that* $\mathcal{P}_\mathcal{J}(\mathcal{V}) > 0$ *for all* $\mathcal{V} \in \mathcal{J}$*.*

The probabilistic interpretation $\mathcal{P}$ *is a* model *of the GCI* $(C \sqsubseteq D)^\kappa$*, denoted as* $\mathcal{P} \models (C \sqsubseteq D)^\kappa$*, iff every* $\mathcal{V} \in \mathcal{J}$ *is a model of* $(C \sqsubseteq D)^\kappa$*. We say that* $\mathcal{P}$ *is a model of the TBox* $\mathcal{T}$ *iff every* $\mathcal{V} \in \mathcal{J}$ *is a model of* $\mathcal{T}$*.* $\mathcal{P}$ *is a model of the assertion* $C(x)^\kappa$ *(respectively* $r(x,y)^\kappa$*), denoted as* $\mathcal{P} \models C(x)^\kappa$ *(respectively*

$\mathcal{P} \models r(x,y)^\kappa$), iff every $\mathcal{V} \in \mathcal{J}$ is a model of $C(x)^\kappa$ (respectively $r(x,y)^\kappa$). We say that $\mathcal{P}$ is a model of the ABox $\mathcal{A}$ iff every $\mathcal{V} \in \mathcal{J}$ is a model of $\mathcal{A}$.

The distribution $\mathcal{P}_\mathcal{J}$ is consistent with the BN $\mathcal{B}$ if for every possible world ω of the variables in V it holds that

$$\sum_{\mathcal{V}\in\mathcal{J}, v^\mathcal{V}=\omega} P_J(\mathcal{V}) = P(\omega).$$

The probabilistic interpretation $\mathcal{P}$ is a model *of the KB $\mathcal{K} = (\mathcal{T}, \mathcal{A}, \mathcal{B})$ iff it is a (probabilistic) model of both $\mathcal{T}$ and $\mathcal{A}$, and is consistent with $\mathcal{B}$.*

$\mathcal{BALC}$ allows for the notion of a complex context and a context language. Due to space requirements we provide only the basic definitions required for the presentation of the reasoning problems in $\mathcal{BALC}$. For a thorough explanation, the interested reader can consult [4].

A *complex context* ϕ is a finite set containing one or more primitive contexts. Note that this allows us to easily convert from primitive to complex contexts by simply enclosing primitive contexts in an additional set; e.g., the primitive context κ would be converted into the complex context $\{\kappa\}$. Given a valuation function $v^\mathcal{V}$ and a complex context $\phi = \{\alpha_1, \ldots, \alpha_n\}$ we say that $v^\mathcal{V} \models \phi$ iff $v^\mathcal{V}$ satisfies at least one $\alpha_i \in \phi$. This immediately gives the result that if $v^\mathcal{V} \models \kappa$ then $v^\mathcal{V} \models \{\kappa\}$ as complex contexts are consistent with primitive contexts. Thus, in the following we assume that all contexts are in complex form unless explicitly stated otherwise. Finally we say that $\phi \models \psi$ iff for all $v^\mathcal{V} \models \phi$ then $v^\mathcal{V} \models \psi$, or alternatively $\phi \models \psi$ iff for all Bayesian worlds ω such that $\omega \models \phi$ then $\omega \models \psi$.

Given complex contexts $\phi = \{\alpha_1, \ldots, \alpha_n\}$ and $\psi = \{\beta_1, \ldots, \beta_m\}$ we define the operations

$$\phi \vee \psi := \phi \cup \psi, \qquad \text{and}$$
$$\phi \wedge \psi := \bigcup_{\alpha\in\phi,\beta\in\psi} \{\alpha \cup \beta\} = \{\alpha \cup \beta | \alpha \in \phi, \beta \in \psi\}.$$

That is we define operations that fulfill the roles of propositional disjunction ($\vee$) and propositional conjunction ($\wedge$), where disjunction has the property that either one of the two contexts holds and conjunction requires that both hold.

Lemma 5. *Given complex contexts ϕ and ψ we have*

1. *$\omega \models \phi \vee \psi$ iff $\omega \models \phi$ or $\omega \models \psi$, and*
2. *$\omega \models \phi \wedge \psi$ iff $\omega \models \phi$ and $\omega \models \psi$.*

Two important special complex contexts are top ($\top$) and bottom ($\bot$), which are defined such that for all valuation functions $v^\mathcal{V}$, $v^\mathcal{V} \models \top$ and $v^\mathcal{V} \not\models \bot$. If there are n primitive contexts these can be defined as $\top := \{\alpha_1, \ldots, \alpha_n\}$ and $\bot := v^\mathcal{V} \models \emptyset$.

After all these definitions, we are now ready to introduce and study the relevant decision and computation problems for our logic.

3 Total Concept Satisfiability and Consistency

As a first decision problem, we consider concept satisfiability. Generalizing from the classical case, we say that a concept C is totally satisfiable if it is satisfiable in all the contexts of a knowledge base that have positive probability.

Definition 6 (Total concept satisfiability). *A concept C is totally satisfiable with respect to a $\mathcal{BALC}$ KB $\mathcal{K}$ iff there exists a probabilistic model $\mathcal{P} = (\mathcal{J}, \mathcal{P}_{\mathcal{J}})$ of $\mathcal{K}$ s.t. $C^{\mathcal{V}} \neq \emptyset$ for all $\mathcal{V} \in \mathcal{J}$.*

When reasoning about a $\mathcal{BALC}$ KB it will be useful to refer to the specific TBox (or ABox) associated with a specific Bayesian world ω; i.e., the TBox (or ABox) containing only the axioms that hold in ω. We call this reduced TBox (or ABox) a restriction to the world ω, denoted as $\mathcal{T}_\omega$ (or respectively $\mathcal{A}_\omega$). Formally, if $\mathcal{K} = (\mathcal{T}, \mathcal{A}, \mathcal{B})$ is a $\mathcal{BALC}$ KB, and ω a world, the *restriction* of $\mathcal{T}$ and $\mathcal{A}$ to a world ω are defined as

$$\mathcal{T}_\omega := \{(C \sqsubseteq D) \mid (C \sqsubseteq D)^\kappa \in \mathcal{T}, \omega \models \kappa\}$$
$$\mathcal{A}_\omega := \{\alpha \mid \alpha^\kappa \in \mathcal{A}, \alpha \in \{C(x), r(x,y)\}, \omega \models \kappa\}.$$

We can think of total concept satisfiability as requiring that a concept be classically satisfiable in each restricted knowledge base $(\mathcal{T}_\omega, \mathcal{A}_\omega)$ where ω corresponds to some probabilistic world with positive probability.

Theorem 7. *Given a $\mathcal{BALC}$ KB $\mathcal{K}$, the concept C is* not *totally satisfiable in $\mathcal{K}$ iff there exists a world ω such that $P(\omega) > 0$ and C is unsatisfiable in the $\mathcal{ALC}$ KB $(\mathcal{T}_\omega, \mathcal{A}_\omega)$.*

The theorem suggests a process for verifying total satisfiability. In the following, we provide an algorithm based on this idea, but before, we must introduce some additional terminology.

We use $\phi_{\mathcal{K}}^C$ to denote the context that describes all worlds that lead to restricted $\mathcal{BALC}$ KB where C is not satisfiable. That is $\omega \models \phi_{\mathcal{K}}^C$ iff C is unsatisfiable in $(\mathcal{T}_\omega, \mathcal{A}_\omega)$. Moreover, $\phi_{\mathcal{B}}$ is context that describes all worlds with probability greater than 0 in the BN; i.e., if $\omega \models \phi_{\mathcal{B}}$ then $P(\omega) > 0$. Theorem 7 suggests that C is not totally satisfiable if there is a world that models both $\phi_{\mathcal{K}}^C$ and $\phi_{\mathcal{B}}$. This is formalized in the following theorem.

Theorem 8. *The concept C is not totally satisfiable w.r.t. the KB $\mathcal{K}$ iff $\phi_{\mathcal{K}}^C \wedge \phi_{\mathcal{B}}$ is satisfiable.*

We need to provide a method for computing the formulas $\phi_{\mathcal{K}}^C$ and $\phi_{\mathcal{B}}$. For the former, we present a variant of the glass-box approach for axiom pinpointing [3, 7, 9], originally based on the ideas from [2]. The idea for this approach is to modify the standard tableaux algorithm for $\mathcal{ALC}$, to keep track of the contexts in which the derived elements in the tableau hold. The modified tableaux rules are presented in Figure 1. Understanding these rules requires some additional notions that we present next.

⊓-rule if 1. $(C_1 \sqcap C_2)(x)^\phi \in \mathcal{A}$, and 2. either $C_1(x)^\phi$ or $C_2(x)^\phi$ is $\mathcal{A}$-insertable
then $\mathcal{A}' := (\mathcal{A} \oplus C_1(x)^\phi) \oplus C_2(x)^\phi$.
⊔-rule if 1. $(C_1 \sqcup C_2)(x)^\phi \in \mathcal{A}$, and 2. both $C_1(x)^\phi$ and $C_2(x)^\phi$ are $\mathcal{A}$-insertable
then $\mathcal{A}' := \mathcal{A} \oplus C_1(x)^\phi$, $\mathcal{A}'' := \mathcal{A} \oplus C_2(x)^\phi$.
$\exists_1$-rule if $(\exists R.C)(x)^\phi \in \mathcal{A}$, and there exists $\alpha \in \phi$ such that $(\exists R.C)(x)^\alpha$ is $\mathcal{A}$-insertable
then $\mathcal{A}' := \mathcal{A} \oplus (\exists R.C)(x)^\alpha$
$\exists_2$-rule if $(\exists R.C)(x)^\alpha \in \mathcal{A}$, there is no z such that both $R(x,z)^\alpha$ and $C(z)^\alpha$ are not $\mathcal{A}$-insertable, and x is not blocked
then $\mathcal{A}' := (\mathcal{A} \oplus R(x,y)^\alpha) \oplus C(y)^\alpha$, where y is a new individual name and $y > y'$ for all individual names $y' \in \mathcal{A}$.
∀-rule if 1. $\{(\forall R.C)(x)^\phi, R(x,y)^\psi\} \subseteq \mathcal{A}$, and 2. $C(y)^{\phi\wedge\psi}$ is $\mathcal{A}$-insertable
then $\mathcal{A}' := \mathcal{A} \oplus C(y)^{\phi\wedge\psi}$
⊑-rule if 1. $(C \sqsubseteq D)^\phi \in \mathcal{T}, E(x)^\psi \in \mathcal{A}$, and 2. $(\neg C \sqcup D)(x)^{\phi\wedge\psi}$ is $\mathcal{A}$-insertable
then $\mathcal{A}' := \mathcal{A} \oplus (\neg C \sqcup D)(x)^{\phi\wedge\psi}$

Fig. 1. Expansion rules for constructing $\phi_\mathcal{K}^C$

An assertion $C(x)^\phi$ is $\mathcal{A}$-*insertable* in an ABox $\mathcal{A}$ iff whenever there is a ψ such that $C(x)^\psi \in \mathcal{A}$, then $\phi \not\models \psi$. In the expansion rules $\oplus$ is used as shorthand for $\mathcal{A} \oplus C(x)^\phi := (\mathcal{A} \setminus \{C(x)^\psi\}) \cup \{C(x)^{\phi\vee\psi}\}$ if $C(x)^\psi \in \mathcal{A}$ and $\mathcal{A} \cup \{C(x)^\phi\}$ otherwise; and $\mathcal{A} \oplus r(x,y)^\phi := (\mathcal{A} \setminus \{r(x,y)^\psi\}) \cup \{r(x,y)^{\phi\vee\psi}\}$ if $r(x,y)^\psi \in \mathcal{A}$ and $\mathcal{A} \cup \{r(x,y)^\phi\}$ otherwise. The individual x is an *ancestor* of y if there is a chain of role assertions connecting x to y. The individual x *blocks* y iff x is an ancestor of y and for every $C(y)^\psi \in \mathcal{A}$, it is the case that $C(x)^\phi \in \mathcal{A}$ for some ϕ such that $\psi \models \phi$. An ABox contains a *clash* if it contains contradictory assertions. A *rule application* refers to applying one of the expansions rules to an ABox in order to generate a new ABox, and an ABox is *fully expanded* if none of the expansions rules can be applied to it.

Algorithm for finding $\phi_\mathcal{K}^C$: Given a $\mathcal{BALC}$ knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{A}, \mathcal{B})$ and a concept C we start by asserting that there exists an instance of C by adding $C(x)^\top$, where x is a fresh individual name to $\mathcal{A}$. We then apply the expansion rules in Figure 1 until all ABoxes are fully expanded. If at least one clash-free ABox is found, then return $\phi_\mathcal{T} = \perp$.

If at least one clash is found in each completely expanded ABox then we know that there exists some valuation ω s.t. $(\mathcal{T}_\omega, \mathcal{A}_\omega)$ is inconsistent. We now construct and return a context encoding these valuations. We do this by selecting a context representing a clash from each final ABox and then combining these contexts. Suppose $\mathcal{A}_1 \ldots \mathcal{A}_n$ are the completely expanded ABoxes then

$$\phi_{\mathcal{A}_i}^C = \vee_{C(x)^\phi, \neg C(x)^\psi \in \mathcal{A}_i} (\phi \wedge \psi)$$

is the context encoding all clashes for the i-th final ABox. After constructing such a context for each final ABox we combine them into the context

$$\phi_\mathcal{K}^C = \wedge_{i=1}^n \phi_{\mathcal{A}_i}^C.$$

We have shown [4] that this construction algorithm has the following characteristics.

Theorem 9. *The algorithm for finding $\phi_{\mathcal{K}}^{C}$ terminates and is sound and complete.*

Finally, we have the following corollary that puts together all previously presented work to determine total concept satisfiability. Once $\phi_{\mathcal{K}}^{C}$ has been constructed we can determine whether $\mathcal{K}$ is totally concept unsatisfiable for C by iterating over all worlds ω and calculating $P(\omega)$.

Corollary 10. *C is not totally satisfiable in $\mathcal{K}$ iff there is a world $\omega \models \phi_{\mathcal{K}}^{C}$ such that $P(\omega) > 0$.*

As is usual for DLs we say that a $\mathcal{BALC}$ knowledge base is consistent if, and only if, it has a (probabilistic) model. We will often write $\mathcal{K} \models \mathcal{P}$ when a probabilistic interpretation $\mathcal{P}$ is a model of $\mathcal{K}$.

Recall that in our definition of a V-interpretation we require that the domain $\Delta^{\mathcal{V}}$ be non-empty. This leads us to the obvious consequence that a $\mathcal{BALC}$ knowledge base is only consistent if $\top$ is totally satisfiable.

Theorem 11. *A $\mathcal{BALC}$ knowledge base $\mathcal{K}$ is* consistent *if, and only if, $\top$ is totally satisfiable in $\mathcal{K}$.*

This theorem shows that the $\mathcal{BALC}$ consistency problem can be reduced to an instance of the total concept satisfiability problem. Leading to the following lemma.

Lemma 12 (Complexity of consistency). *Checking the consistency of a $\mathcal{BALC}$ knowledge base is in $O(2^{||\mathcal{K}||})$.*

4 Subsumption

We adapt the classical definition of subsumption for $\mathcal{BALC}$ in order to take contexts into account. We do this by saying that a concept C subsumes a concept D in context κ if, and only if, in all worlds where κ is satisfied C is necessarily subsumed by D.

Definition 13 (Contextual subsumption). *Given $\mathcal{K} = (\mathcal{T}, \mathcal{A}, \mathcal{B})$ a $\mathcal{BALC}$ KB, C, D concepts, and κ a context. C is* contextually subsumed *by D in κ w.r.t. $\mathcal{K}$, denoted as $\mathcal{K} \models (C \sqsubseteq D)^{\kappa}$, if every probabilistic model of $\mathcal{K}$ is a probabilistic model of $(C \sqsubseteq D)^{\kappa}$.*

In our setting, however, contexts are used as aids for expressing the uncertainty of different consequences (e.g., subsumptions) to hold. Hence, we introduce the notion of the probability of a subsumption.

Definition 14 (Probability of a Subsumption). *Given the probabilistic model* $\mathcal{P} = (\mathcal{J}, P_\mathcal{J})$ *of the KB* $\mathcal{K}$*, and the concepts* C, D*, the* probability of $C \sqsubseteq D$ *is*

$$P_\mathcal{P}((C \sqsubseteq D)^\kappa) = \sum_{\mathcal{V} \in \mathcal{J}, \mathcal{V} \models (C \sqsubseteq D)^\kappa} P_\mathcal{J}(\mathcal{V}).$$

The probability of $(C \sqsubseteq D)^\kappa$ *w.r.t.* $\mathcal{K}$ *is*

$$P_\mathcal{K}((C \sqsubseteq D)^\kappa) = inf_{\mathcal{P} \models \mathcal{K}} P_\mathcal{P}((C \sqsubseteq D)^\kappa).$$

That is the probability of a subsumption in a specific model is the sum of the probabilities of the worlds in which C is subsumed by D in context κ; notice that this trivially includes all worlds where κ does not hold. In the case where $\mathcal{K}$ is inconsistent we define the probability of all subsumptions as 1 to ensure our definition is consistent with general probability theory ($\inf(\emptyset) = \infty$ in general).

Note that the relationship between the contextual subsumption problem and the probability of a subsumption is as one would expect. Namely we have that a KB entails a contextual subsumption iff the probability of the subsumption in the KB is 1.

Theorem 15. *Given a KB* $\mathcal{K}$*, concepts* C *and* D*, and a context* κ*, it holds that:*

$$\mathcal{K} \models (C \sqsubseteq D)^\phi \textit{ iff } P_\mathcal{K}((C \sqsubseteq D)^\phi) = 1.$$

This is convenient as it provides a method of reducing the contextual subsumption problem to calculating the probability of a subsumption. The following theorem provides a means of calculating this probability.

Theorem 16. *For a consistent KB* $\mathcal{K} = (\mathcal{T}, \mathcal{A}, \mathcal{B})$*, a contextual subsumption* $(C \sqsubseteq D)^\phi$*, and the extended KB* $\mathcal{K}' = (\mathcal{T}, \mathcal{A} \cup \{C(x)^\phi, \neg D(x)^\phi\}, \mathcal{B})$ *we have*

$$P_\mathcal{K}((C \sqsubseteq D)^\phi) = \sum_{\omega \models \phi_{\mathcal{K}'}} P(\omega) + 1 - P(\phi).$$

Furthermore, this approach runs in exponential time in the size of the input KB (given that the size of the BN is included).

Theorem 17. *Given a knowledge base* $\mathcal{K}$ *we can calculate the probability of a contextual subsumption in time* $O(exp(||\mathcal{K}|| + |V|))$*.*

5 Partial Concept Satisfiability

Partial concept satisfiability is a weaker form of satisfiability in $\mathcal{BALC}$. We formally define this notion next.

Definition 18 (Partial Concept Satisfiability). *The concept* C *is partially satisfiable with respect to the* $\mathcal{BALC}$ *KB* $\mathcal{K}$ *iff there exists a probabilistic model* $\mathcal{P} = (\mathcal{J}, P_\mathcal{J})$ *of* $\mathcal{K}$ *and a* V*-interpretation* $\mathcal{V} \in \mathcal{J}$*, with* $P_\mathcal{J}(\mathcal{V}) > 0$*, and* $C^\mathcal{V} \neq \emptyset$*.*

Clearly a concept cannot be even partially satisfiable if it is necessarily empty in all worlds. This leads us to the following theorem.

Theorem 19. *A concept C is partially satisfiable with respect to a $\mathcal{BALC}$ KB $\mathcal{K}$ iff $\mathcal{K} \not\models C \sqsubseteq \bot$.*

We complete this analysis by using the fact that if a KB is s.t. $P_{\mathcal{K}}((C \sqsubseteq \bot)^{\top}) = 1$ then there exist no model which has a V-interpretation where C is not empty.

Theorem 20. *C is not partially satisfiable in $\mathcal{K}$ iff $P((C \sqsubseteq \bot)^{\top}) = 1$.*

Next, we define the probability of partial satisfiability in a similar way to the probability of a subsumption. That is we first define the probability of partial satisfiability for a concept C in a probabilistic interpretation and then use this to define it in the context of a knowledge base.

Definition 21 (Probability of partial satisfiability). *Given a concept C and a KB $\mathcal{K}$, the* probability of C being partially satisfiable *in a probabilistic model $\mathcal{P}$ of $\mathcal{K}$ is*

$$P_{\mathcal{P}}(C) := \sum_{\mathcal{V} \in \mathcal{J}, \mathcal{V} \not\models (C \sqsubseteq \bot)^{\top}} P_{\mathcal{J}}(\mathcal{V}).$$

The probability of C being partially satisfiable *in $\mathcal{K}$ is*

$$P_{\mathcal{K}}(C) := \sup_{\mathcal{P} \models \mathcal{K}} (P_{\mathcal{P}}(C)).$$

We can reduce this problem to subsumption, in particular the probability of a subsumption.

Theorem 22. *C is partially satisfiable in $\mathcal{K}$ with probability $1 - P_{\mathcal{K}}((C \sqsubseteq \bot)^{\top})$.*

Overall, this means that we can deal with this problem, within the same complexity bounds, as long as we are able to handle total concept satisfiability, and the probability that it holds. Thus, we have already developed a method for solving it. We now turn our attention to instance checking, and the influence of the ABox.

6 Instance Checking

We consider a probabilistic extension to the classical instance checking problem. In $\mathcal{BALC}$ we call this problem probabilistic instance checking and we define both a decision problem problem and probability calculation for it.

Definition 23 (Instance). *Given an individual name a, a concept C, a primitive context ϕ, and a KB $\mathcal{K}$, we say that a is an* instance *of C in ϕ for $\mathcal{K}$, written as $\mathcal{K} \models C(x)^{\phi}$, iff for all probabilistic models $\mathcal{P} = (\mathcal{J}, P_{\mathcal{J}})$ of $\mathcal{K}$ we have that $a^{\mathcal{V}} \in C^{\mathcal{V}}$ for all $\mathcal{V} \in \mathcal{J}$ with $v^{\mathcal{V}} \models \phi$.*

Note that if the context associated with the instance check is $\top$ this definition is very similar to the classical case. In this case it would be required that the named individual be a member of the concept in all cases (in all worlds with positive probability) which is similar to the classical case. We next show how we go about providing a procedure that solves this problem.

Theorem 24. *Given an individual name a, a concept C, a context ϕ, and a KB $\mathcal{K} = (\mathcal{T}, \mathcal{A}, \mathcal{B})$, a is an instance of C in ϕ iff $P_{\mathcal{K}'}((D \sqsubseteq C)^{\phi}) = 1$ where D is a new concept name not in $\mathcal{T}$ and $\mathcal{K}' = (\mathcal{T}, \mathcal{A} \cup \{D(a)^{\phi}\}, \mathcal{B})$.*

Since we have reduced instance checking to probabilistic subsumption we have the result that instance checking is in the same complexity class as probabilistic subsumption. This gives us the following lemma.

Lemma 25. *Probabilistic instance checking in a knowledge base $\mathcal{K}$ is in the complexity class $O(2^{||\mathcal{K}||+|V|})$.*

We next formalize the probability calculation for the instance checking problem. This is done in a very similar way to the probability of a subsumption.

Definition 26 (probability of an instance). *The* probability of an instance *in a probabilistic model $\mathcal{P} = (\mathcal{J}, P_{\mathcal{J}})$ of a KB $\mathcal{K}$ is*

$$P_{\mathcal{P}}(C(x)^{\phi}) = \sum_{\mathcal{V} \in \mathcal{J}, \mathcal{V} \models C(x)^{\phi}} P_{\mathcal{J}}(\mathcal{V}).$$

The probability instance *w.r.t. a KB $\mathcal{K}$ is*

$$P_{\mathcal{K}}(C(x)^{\phi}) = \inf_{\mathcal{K} \models \mathcal{P}} P_{\mathcal{P}}(C(x)^{\phi}).$$

The probability of all instance checks for an inconsistent KB is always 1 to keep our definitions consistent with probability theory.

Similar to the algorithm for solving the decision problem we now show that the probability calculation can be reduced to subsumption.

Theorem 27. *Given the individual name a, concept C, primitive context ϕ, and KB $\mathcal{K} = (\mathcal{T}, \mathcal{A}, \mathcal{B})$, a is an instance of C in ϕ with probability $P_{\mathcal{K}'}((D \sqsubseteq C)^{\phi})$ where D is a new concept name not in $\mathcal{T}$ and $\mathcal{K}' = (\mathcal{T}, \mathcal{A} \cup \{D(a)^{\phi}\}, \mathcal{B})$.*

From this result we again see that calculating the probability of an instance can be reduced in constant time to probabilistic subsumption. Since we only add a single statement to the ABox of the knowledge base the input size does not change meaningfully post reduction.

Lemma 28. *Calculating the probability of an instance in a knowledge base $\mathcal{K}$ can be done in time $O(2^{||\mathcal{K}||+|V|})$.*

7 Conclusions

We have presented a new probabilistic extension of $\mathcal{ALC}$ based on the ideas of Bayesian ontology languages. In contrast to previous work that focused mainly on light-weight DLs, in our logic it is possible to express inconsistent knowledge, which requires the study of new reasoning problems. We developed a glass-box tableaux-based algorithm for finding out the context in which a given consequence holds. Using this information, we could also compute the probability of the consequence itself.

We showed that all the reasoning problems studied remain ExpTime-complete, just as the complexity of reasoning in classical $\mathcal{ALC}$. Our results also hint at possible optimizations that can be exploited in an attempt to develop an efficient reasoner for our logic. As future work, we plan to study these optimizations in detail, and analyse their applicability in practice.

Another interesting line of work would be to implement our ideas, and compare the resulting tool against other probabilistic DL reasoners. In particular, it would be interesting to compare against the tools from [11], which are also based on an extension of the tableaux algorithm, and use probabilistic semantics that are very similar to ours.

References

1. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P. (eds.): The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press, second edn. (2007)
2. Baader, F., Hollunder, B.: Embedding defaults into terminological knowledge representation formalisms. J. Autom. Reasoning **14**(1), 149–180 (1995). https://doi.org/10.1007/BF00883932, https://doi.org/10.1007/BF00883932
3. Baader, F., Peñaloza, R.: Axiom pinpointing in general tableaux. Journal of Logic and Computation **20**(1), 5–34 (February 2010). https://doi.org/10.1093/logcom/exn058, special Issue: Tableaux and Analytic Proof Methods
4. Botha, L.: The Bayesian Description Logic ALC. Master's thesis, University of Cape Town, South Africa (2018)
5. Ceylan, İ.İ., Peñaloza, R.: The Bayesian description logic BEL. In: Demri, S., Kapur, D., Weidenbach, C. (eds.) Proceedings of the 7th International Joint Conference on Automated Reasoning (IJCAR 2014). Lecture Notes in Computer Science, vol. 8562, pp. 480–494. Springer (2014)
6. Ceylan, İ.İ., Peñaloza, R.: The Bayesian ontology language BEL. Journal of Automated Reasoning **58**(1), 67–95 (2017). https://doi.org/10.1007/s10817-016-9386-0
7. Lee, K., Meyer, T.A., Pan, J.Z., Booth, R.: Computing maximally satisfiable terminologies for the description logic ALC with cyclic definitions. In: Parsia, B., Sattler, U., Toman, D. (eds.) Proceedings of the 2006 International Workshop on Description Logics (DL2006). CEUR Workshop Proceedings, vol. 189. CEUR-WS.org (2006), http://ceur-ws.org/Vol-189/submission_29.pdf
8. Lukasiewicz, T., Straccia, U.: Managing uncertainty and vagueness in description logics for the semantic web **6**(4), 291–308 (2008). https://doi.org/10.1016/j.websem.2008.04.001

9. Meyer, T.A., Lee, K., Booth, R., Pan, J.Z.: Finding maximally satisfiable terminologies for the description logic ALC. In: Proceedings of the Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference. pp. 269–274. AAAI Press (2006), http://www.aaai.org/Library/AAAI/2006/aaai06-043.php
10. Riguzzi, F., Bellodi, E., Lamma, E., Zese, R.: Probabilistic description logics under the distribution semantics. Semantic Web **6**(5), 477–501 (2015). https://doi.org/10.3233/SW-140154
11. Zese, R., Bellodi, E., Riguzzi, F., Cota, G., Lamma, E.: Tableau reasoning for description logics and its extension to probabilities. Ann. Math. Artif. Intell. **82**(1-3), 101–130 (2018). https://doi.org/10.1007/s10472-016-9529-3