Inference in Probabilistic Ontologies with Attributive Concept Descriptions and Nominals

Rodrigo Bellizia Polastro and Fabio Gagliardi Cozman

Escola Politécnica, Universidade de São Paulo - São Paulo, SP - Brazil
rodrigopolastro@usp.br, fgcozman@usp.br

Abstract. This paper proposes a probabilistic description logic that combines (i) constructs of the well-known $\mathcal{ALC}$ logic, (ii) probabilistic assertions, and (iii) limited use of nominals. We start with our recently proposed logic CR$\mathcal{ALC}$, where any ontology can be translated into a relational Bayesian network with partially specified probabilities. We then add nominals to restrictions, while keeping CR$\mathcal{ALC}$'s interpretation-based semantics. We discuss the clash between a domain-based semantics for nominals and an interpretation-based semantics for queries, keeping the latter semantics throughout. We show how inference can be conducted in CR$\mathcal{ALC}$ and present examples with real ontologies that display the level of scalability of our proposals.

Key words: $\mathcal{ALC}$ logic, nominals, Bayesian/credal networks.

1 Introduction

Semantic web technologies typically rely on the theory of description logics, as these logics offer reasonable flexibility and decidability at an computational cost that seems to be acceptable [1, 2]. Recent literature has examined ways to enlarge description logics with uncertainty representation and management. In this paper we focus on two challenges in uncertainty representation: we seek to define coherent semantics for a probabilistic description logic and to derive algorithms for inference in this logic. More precisely, we wish to attach sensible semantics to sentences such as

$$P(\mathsf{Merlot}(\mathsf{a})|\exists\mathsf{Color}.\{\mathsf{red}\}) = \alpha, \tag{1}$$

that should refer to the probability that a particular wine is Merlot, given that its color is red. Note the presence of the nominal red in this expression, a feature that complicates matters considerably. Also, we wish to *compute* the smallest α that makes Expression (1) true with respect to a given ontology. This latter calculation is an "inference"; that is, an inference is the calculation of a tight bound on the probability of some assertion.

We have recently proposed a probabilistic description language, referred to as "credal $\mathcal{ALC}$" or simply CR$\mathcal{ALC}$ [3], that combines the well-known Attributive Concept Description ($\mathcal{ALC}$) logic and probabilistic inclusions such as

$P(\mathsf{hasWineSugarDry}|\mathsf{WineSugar}) = \beta$. One of the main features of CR$\mathcal{ALC}$ is that it adopts an interpretation-based semantics that allows it to handle probabilistic queries involving Aboxes i.e., sets of assertions. In CR$\mathcal{ALC}$ there is a one-to-one correspondence between a consistent set of sentences and a relational *credal* network; that is, a relational Bayesian network with partially specified probabilities. An inference in CR$\mathcal{ALC}$ is equivalent to an inference in such a network.

In this paper we wish to extend our previous effort [3] by handling realistic examples, such as the Wine and the Kangaroo ontologies, and by adding to CR$\mathcal{ALC}$ a limited version of nominals (that is, reference to individuals in concept descriptions). Nominals often appear in ontologies; besides, the study of nominals touches on central issues in probabilistic logic, as discussed later.

Section 2 summarizes the main features of CR$\mathcal{ALC}$ and reviews existing probabilistic description logics, emphasizing the differences between them and CR$\mathcal{ALC}$. Section 3 presents the challenges created by nominals, and introduces our proposal for dealing with (some of) them. Section 4 described our experiments with real ontologies in the literature. While our previous effort [3] was mainly directed at theoretical analysis of CR$\mathcal{ALC}$, in this paper we move to evaluation of inference methods in real ontologies. Finally, Section 5 evaluates the results and draws some thoughts on the next steps in the creation of a complete probabilistic semantic web.

2 Probabilistic Description Logics and CR$\mathcal{ALC}$

In this section we review a few important concepts, the literature on probabilistic description logics, and the logic CR$\mathcal{ALC}$. This section is based on our previous work [3].

2.1 A few definitions

Assume a vocabulary containing *individuals*, *concepts*, and *roles* [1]. Concepts and roles are combined to form new concepts using a set of *constructors*. In $\mathcal{ALC}$ [4], constructors are *conjunction* ($C \sqcap D$), *disjunction* ($C \sqcup D$), *negation* ($\neg C$), *existential* restriction ($\exists r.C$) and *value* restriction ($\forall r.C$). A *concept inclusion* is denoted by $C \sqsubseteq D$ and *concept definition* is denoted by $C \equiv D$, where C and D are concepts. Usually one is interested in *concept subsumption*: whether $C \sqsubseteq D$ for concepts C and D. A set of concept inclusions and definitions is called a *terminology*. If an inclusion/definition contains a concept C in its left hand side and a concept D in its right hand side, the concept C *directly uses* D. The transitive closure of "*directly uses*" is indicated by "*uses*". A terminology is *acyclic* if it is a set of concept inclusions and definitions such that no concept in the terminology uses itself [1]. Typically terminologies only allow the left hand side of a concept inclusion/definition to contain a concept name (and no constructors). Concept $C' \sqcup \neg C'$ is denoted by $\top$ and concept $C' \sqcap \neg C'$ is denoted by $\bot$, where C' is a dummy concept that does not appear anywhere else; also, $r.\top$ is abbreviated by r (for instance, $\exists r$).

A set of *assertions* about individuals may be associated to a terminology. An assertion $C(a)$ directly uses assertions of concepts (resp. roles) directly used by C instantiated by a (resp. by (a, b) for $b \in \mathcal{D}$), and likewise for the "uses" relation. As an example, we may have the assertions such as $\mathsf{Fruit}(\mathsf{appleFromJohn})$ and $\mathsf{buyFrom}(\mathsf{houseBob}, \mathsf{John})$.

The semantics of a description logic is almost always given by a *domain* $\mathcal{D}$ and an *interpretation* $\mathcal{I}$. The domain $\mathcal{D}$ is a nonempty set; we often assume its cardinality to be given as input. Note that in description logics the cardinality of the domain is usually left unspecified, while in probabilistic description logics this cardinality is usually specified (Section 2.2). The interpretation function $\mathcal{I}$ maps each individual to an element of the domain, each concept name to a subset of the domain, each role name to a binary relation on $\mathcal{D} \times \mathcal{D}$. The interpretation function is extended to other concepts as follows: $\mathcal{I}(C \sqcap D) = \mathcal{I}(C) \cap \mathcal{I}(D)$, $\mathcal{I}(C \sqcup D) = \mathcal{I}(C) \cup \mathcal{I}(D)$, $\mathcal{I}(\neg C) = \mathcal{D} \backslash \mathcal{I}(C)$, $\mathcal{I}(\exists r.C) = \{x \in \mathcal{D} | \exists y : (x, y) \in \mathcal{I}(r) \wedge y \in \mathcal{I}(C)\}$, $\mathcal{I}(\forall r.C) = \{x \in \mathcal{D} | \forall y : (x, y) \in \mathcal{I}(r) \rightarrow y \in \mathcal{I}(C)\}$. An inclusion $C \sqsubseteq D$ is entailed iff $\mathcal{I}(C) \subseteq \mathcal{I}(D)$, and $C \equiv D$ iff $\mathcal{I}(D) = \mathcal{I}(D)$.

Some logics in the literature offer significantly larger sets of features, such as numerical restrictions, role hierarchies, inverse and transitive roles (the OWL language contains several such features [2]). And most description logics have direct translations into multi-modal logics [5] or fragments of first-order logic [6]. The translation of $\mathcal{ALC}$ to first-order logic is: each concept C is interpreted as a unary predicate $C(x)$; each role r is interpreted as a binary predicate $r(x, y)$; the other constructs have direct translations into first-order logic, (e.g. $\exists r.C$ is translated to $\exists y : \; r(x, y) \wedge C(y)$ and $\forall r.C$ to $\forall y : \; r(x, y) \rightarrow C(y)$).

2.2 Probabilistic description logics

There are several probabilistic description logics in the literature. Heinsohn [7], Jaeger [8] and Sebastiani [9] consider probabilistic inclusion axioms such as $P_{\mathcal{D}}(\mathsf{Plant}) = \alpha$, meaning that a randomly selected individual is a Plant with probability α. This interpretation characterizes a *domain-based* semantics. Sebastiani also allows assessments as $P(\mathsf{Plant}(\mathsf{Tweety})) = \alpha$, specifying probabilities over the interpretations themselves, characterizing an *interpretation-based* semantics. Most proposals for probabilistic description logics adopt a domain-based semantics [7–14, 16], while relatively few adopt an interpretation-based semantics [9, 17].

Direct inference refers to the transfer of statistical information about domains to specific individuals [18, 19]. Direct inference is a problem for domain-based semantics; for instance, from $P(\mathsf{FlyingBird}) = 0.3$ there is nothing to be concluded over $P(\mathsf{FlyingBird}(\mathsf{Tweety}))$. We discuss direct inference further in Section 3. Due to the difficulties in solving direct inference, most proposals for probabilistic description logics with a domain-based semantics simply do not handle assertions. Dürig and Studer avoid direct inference by only allowing probabilities over assertions [11]. Also note that Lukasiewicz has proposed another strategy, where expressive logics are combined with probabilities through an entailment relation with non-monotonic properties, *lexicographic entailment* [12, 14, 15].

The probabilistic description logics mentioned so far do not encode independence relations, neither syntactically nor semantically. A considerable number of proposals for probabilistic description logics that represent independence through graphs has appeared in the last decade or so, in parallel with work on statistical relational models [20, 21]. Logics such as P-CLASSIC [13], Yelland's Tiny Description Logic [16], Ding and Peng's BayesOWL language [10], and Staker's logic [22] all employ Bayesian networks and various constructs of description logics to define probabilities over domains — that is, they have domain-based semantics. Costa and Laskey's PR-OWL language [17] uses an interpretation-based semantics inherited from Multi-entity Bayesian networks (MEBNs) [23]. Related and notable efforts by Nottelmann and Fuhr [24] and Hung et al [25] should be mentioned (note also the existence of several non-probabilistic variants of description logics [26]).

The logic CR$\mathcal{ALC}$, proposed previously by the authors [3], adopts an interpretation-based semantics, so as to avoid direct inference and to handle individuals smoothly (this is discussed in more detail later). The closest existing proposal is Costa and Laskey's PR-OWL; indeed one can understand CR$\mathcal{ALC}$ as a trimmed down version of PR-OWL where the focus is on the development of scalable inference methods. The next section summarizes the main features of CR$\mathcal{ALC}$.

2.3 CR$\mathcal{ALC}$

The logic CR$\mathcal{ALC}$ starts with all constructs of $\mathcal{ALC}$: concepts and roles combined through *conjunction* $C \sqcap D$, *disjunction* $C \sqcup D$, *negation* $\neg C$, *existential* restriction $\exists r.C$, and *value* restriction $\forall r.C$; concept *inclusions* $C \sqsubseteq D$ and concept *definitions* $C \equiv D$; individuals and assertions. An inclusion/definition can only have a concept name in its left hand side; also, restrictions $\exists r.C$ and $\forall r.C$ can only use a concept name C (an auxiliary definition may specify a concept C of arbitrary complexity). A set of assertions is called an *Abox*. The semantics is given by a domain $\mathcal{D}$ and an interpretation $\mathcal{I}$, just as in $\mathcal{ALC}$.

Probabilistic inclusions are then added to the language. A probability inclusion reads $P(C|D) = \alpha$, where D is a concept and C is a concept name. If D is $\top$, then we simply write $P(C) = \alpha$. Probabilistic inclusions are required to only have concept names in their conditioned concept (that is, an inclusions such as $P(\forall r.C|D)$ is not allowed). Given a probabilistic inclusion $P(C|D) = \alpha$, say that C "directly uses" B if B appears in the expression of D; again, "uses" is the transitive closure of "directly uses", and a terminology is acyclic if no concept uses itself. The semantics of a probabilistic inclusion is:

$$\forall x : P(C(x)|D(x)) = \alpha, \tag{2}$$

where it is understood that probabilities are over the set of all interpretation mappings $\mathcal{I}$ for a domain $\mathcal{D}$. We also allow assessments such as $P(r) = \beta$ to be made for roles, with semantics

$$\forall x, y : P(r(x, y)) = \beta, \tag{3}$$

where again the probabilities are over the set of all interpretation mappings.

These probabilistic assessments and their semantics allow us to smoothly interpret a query $P(A(a)|B(b))$ for concepts A and B and individuals a and b. Note that asserted facts must be conditioned upon; there is no contradiction between $\forall x : P(C(x)) = \alpha$ and observation $C(a)$ holds, as we can have $P(C(a)|C(a)) = 1$ while $P(C(a)) = \alpha$. As argued by Bacchus [18], for such a semantics to be useful, an assumption of rigidity for individuals must be made (that is, an element of the domain is associated with the same individual in all interpretations).

An *inference* is the calculation of a *query* $P(A(a)|\mathcal{A})$, where A is a concept, a is an individual, and $\mathcal{A}$ is an Abox.

Concept inclusions (including probabilistic ones) and definitions are assumed acyclic: a concept never uses itself. The acyclicity assumption allows one to draw any terminology $\mathcal{T}$ as a directed acyclic graph $\mathcal{G}(\mathcal{T})$ defined as follows. Each concept (even a restriction) is a node, and if a concept C directly uses concept D, then D is a *parent* of C in $\mathcal{G}(\mathcal{T})$. Also, each restriction $\exists r.C$ or $\forall r.C$ also appears as a node in the graph $\mathcal{G}(\mathcal{T})$, and the graph must contain a node for each role r, and an edge from r to each restriction directly using it.

The next step in the definition of CR$\mathcal{ALC}$ is a *Markov condition.* This Markov condition indicates which independence relations should be read off of a set of sentences. The Markov condition is similar to Markov conditions adopted in probabilistic description logics such as P-CLASSIC, BayesOWL and PR-OWL, but in those logics, a set of sentences is specified with the help of a directed acyclic graph, while in CR$\mathcal{ALC}$ a set of sentences $\mathcal{T}$ specifies a directed acyclic graph $\mathcal{G}(\mathcal{T})$. The Markov condition for CR$\mathcal{ALC}$ refers to this directed acyclic graph $\mathcal{G}(\mathcal{T})$. More details on the various possible Markov conditions can be found elsewhere [3].

The idea in CR$\mathcal{ALC}$ is that the structure of the "directly uses" relation encodes stochastic independence through a Markov condition: (i) for every concept $C \in \mathcal{T}$ and for every $x \in \mathcal{D}$, $C(x)$ is independent of every assertion that does not use $C(x)$, given assertions that directly use C; (ii) for every $(x, y) \in \mathcal{D} \times \mathcal{D}$, $r(x, y)$ is independent of all other assertions, except ones that use $r(x, y)$.

A terminology in CR$\mathcal{ALC}$ does not necessarily specify a single probability measure over interpretations. The following *homogeneity condition* is assumed. Consider a concept C with parents $D_1, \ldots, D_m$. For any conjunction of the m concepts $\pm D_i$, where $\pm$ indicates that D_i may be negated or not, we have that $P(C| \pm D_1 \sqcap \pm D_2 \sqcap \cdots \sqcap \pm D_m)$ is a constant. Consequently, any terminology can be translated into a non-recursive relational Bayesian network [28] where some probabilities are not fully specified. Indeed, for a fixed finite domain $\mathcal{D}$, the propositionalization of a terminology $\mathcal{T}$ produces a *credal network* [29].

In this paper we also adopt the *unique names assumption* (distinct elements of the domain refer to distinct individuals), and the assumption that the cardinality of the domain is fixed and known (*domain closure*). While the rigidity, acyclicity and Markov conditions are essential to the meaning of CR$\mathcal{ALC}$, the homogeneity, unique names, and domain closure assumptions seem less motivated, but are necessary for computational reasons at this point.

3 CR$\mathcal{ALC}$ and nominals

The logic $\mathcal{ALC}$ does not allow *nominals*; that is, it does not allow individuals to appear in concept definitions. Nominals are difficult to handle even in standard description logics. Several optimization techniques employed in description logics fail with nominals, and indeed few algorithms and packages do support nominals correctly at this point. For one thing, nominals introduce connections between a terminology and an Abox, thus complicating inferences. To some extent, nominals cause reasoning to require at least partial grounding of a terminology, a process that may incur significant cost. Still, nominals appear in many real ontologies; an important example is the Wine Ontology that has been alluded to in the Introduction [30].

In the context of uncertainty handling, nominals are particularly interesting as they highlight differences between domain-based and interpretation-based semantics. Consider for instance a domain-based semantics, and suppose that a nominal Tweety is used to define a class $\{\mathsf{Tweety}\}$ such that $P(\{\mathsf{Tweety}\}) = 0.3$. Presumably the assessment indicates that Tweety is "selected" with probability 0.3; this is a natural way to interpret nominals. However, now we face the challenge of *direct inference*; for instance, what is $P(\mathsf{Fly}(\mathsf{Tweety}))$? The difficulty is that for every interpretation mapping $\mathcal{I}$, $\mathsf{Fly}(\mathsf{Tweety})$ either holds or not; that is, Tweety either flies or not. Once we fix an interpretation mapping, as required by a domain-based semantics, the probability $P(\mathsf{Fly}(\mathsf{Tweety}))$ gets fixed at 0 or 1. We might then try to consider the set of all interpretation mappings; this takes us back to an interpretation-based semantics. Worse, with the set of interpretations mappings we have mappings fixing the behavior of Tweety either way (flying or otherwise). Thus we cannot conclude anything about the probability that Tweety flies, unless we make additional assumptions about the connection between domains and interpretations. Several proposals exist for connecting domains and interpretations, but the matter is still quite controversial at this point [19].

Our approach is to stay within the interpretation-based semantics of CR$\mathcal{ALC}$, allowing some situations to have nominals and interpreting those situations through an interpretation-based semantics as well. We do not allow general constructs such as

$$\mathsf{WineFlavor} \equiv \{\mathsf{delicate}, \mathsf{moderate}, \mathsf{strong}\}.$$

Rather, we allow nominals only as domains of roles in restrictions. That is, the semantics for $r.\{a\}$ is not based on quantification over the domain, as the semantics given by Expression (2). Instead, we wish to interpret this construct directly either as (in existential restrictions):

$$\exists x : r\,(x, y) \wedge (y = a), \tag{4}$$

or as (in universal restrictions):

$$\forall x : r\,(x, y) \rightarrow (y = a). \tag{5}$$

In restrictions containing more than one nominal as in $r.\{a, b, c\}$, the resulting restriction considers the disjunction of the various assignments to a, b, c and so on.

Inference in crALC, as presented previously [3], grounds a terminology into a credal network. The various conditions previously adopted (acyclicity, domain closure, homogeneity) guarantee that this is always possible. Inference is then the calculation of tight lower and upper bounds on some probability $P(A(a)|\mathcal{A})$ of interest, where A is a concept, a is an individual, and $\mathcal{A}$ is an Abox. Inference can be conducted in the grounded credal network using either exact [31–33] or approximate [34] algorithms.

In the presence of nominals, this grounding of a terminology in crALC may generate huge networks. To avoid this problem, the grounded network must be instantiated only at its relevant nominals; that is, the nominals present in the roles must have specific domains. So, if the role hasProperty(x, y) indicates that the element x has one specific property with value y, then x must be one object being described and y must be a nominal that describes the property indicated by the role. For instance, $\exists\mathsf{hasColor}.\{\mathsf{red}\}$ is interpreted as:

$$\exists x \in D : \mathsf{hasColor}\,(x, y) \wedge (y = \mathsf{red}), \tag{6}$$

where $\mathcal{D}$ is the domain with the elements being described and y ranges over all the nominals that "are" colors. This approach is very close to *Datatypes*, but its most significant characteristic is the definition of the semantic given by Expressions 4 and 5.

Nominals are often used to define mutually exclusive individuals. Although crALC does not have any construct to express this situation, it can be easily done through the inclusion of a probabilistic node that has the mutually exclusive nodes as its parents and a conditional probability table that mimics the behavior of a XOR logic gate. This node must be set as an observed node with value `true` so that all of its parents become inter-dependent.

4 Experiments

We now report on two experiments with well-known networks. The first one was done with the large Wine Ontology, and the second one was done with the not so famous Kangaroo ontology.

The Wine Ontology was extracted from a OWL file available at the ontology repository of the Temporal Knowledge Base Group from Universitat Jaume I (at http://krono.act.uji.es/Links/ontologies/wine.owl/view). It is a ontology that relies extensively in nominals for describing the different kind of wines and their properties. These nominals were represented as indicated in Section 3. Probability inclusions were added to the terminology; assertions were made on properties of an unspecified wine and the wine type was then inferred. Figure 1 shows the network generated for a domain of size 1. We have:

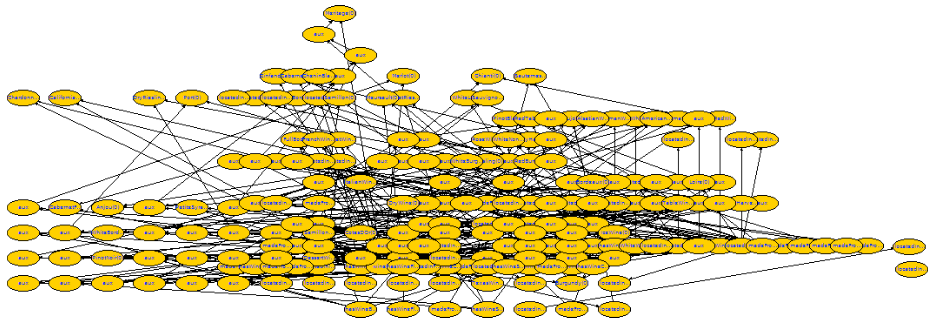


Fig. 1. Network generated from the Wine ontology with domain size 1.

Example 1. The probability of a wine to be Merlot given its body is medium, its color is red, its flavor is moderate, its sugar is dry and it is made from merlot grape:

$$P(\mathsf{Merlot}(a) \mid \mathit{medium}(a), \mathit{red}(a), \mathit{moderate}(a), \mathit{dry}(a), \mathit{merlotGrape}(a)) = 1.0.$$

Example 2. The probability of a wine to be Merlot given its body is medium, its color is red, its flavor is moderate and its sugar is dry:

$$P(\mathsf{Merlot}(b) \mid \mathit{medium}(b), \mathit{red}(b), \mathit{moderate}(b), \mathit{dry}(b)) = 0.5.$$

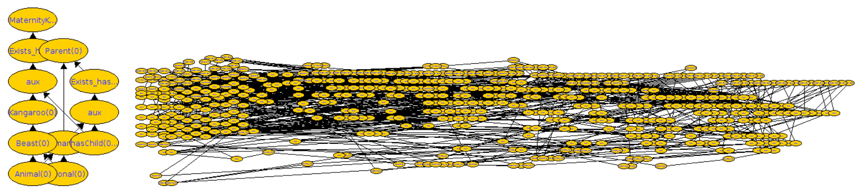
Example 3. The probability of a wine to be Merlot given it is made from merlot grape and its sugar is sweet:

$$P(\mathsf{Merlot}(c) \mid \mathit{merlotGrape}(c), \mathit{sweet}(c)) = 0.0.$$

The Wine ontology only presents restrictions over roles and properties, not having any restriction over individuals. That is, there is no connection between individuals other than the constraints imposed on restrictions by nominals. Consequently, the whole ontology can be translated into a single credal network of fixed size regardless of the actual size of the domain, as far as inference is concerned. Hence there are no qualms about scalability and computational cost when the domain grows. In fact, it was possible to run exact inference in this experiment, even with big domains, since we can separate only the necessary nodes using the Markov condition (we have run exact inferences using the SamIam package, available at http://reasoning.cs.ucla.edu/samiam/).

The second experiment was done with the Kangaroo ontology, adapted from a KRSS file available among the toy ontologies for the CEL System[1] at http://lat.inf.tu-dresden.de/meng/ontologies/kangaroo.cl. Although this ontology does not contain nominals, it uses restrictions amongst individuals in the

[1] A polynomial-time Classifier for the description logic $\mathcal{EL}+$, http://lat.inf.tu-dresden.de/systems/cel/.



(a) $|\mathcal{D}| = 1$ (b) $|\mathcal{D}| = 10$

Fig. 2. Network generated from kangaroo ontology for various domain sizes.

domain, leading to possible concerns on scalability issues as the domain grows. For instance, consider some of the definitions in this ontology:

$$\mathsf{Parent} \equiv \mathsf{Human} \sqcap \forall \mathsf{hasChild}.\mathsf{Human}.$$

$$\mathsf{MaternityKangaroo} \equiv \mathsf{Kangaroo} \sqcap \forall \mathsf{hasChild}.\mathsf{Kangaroo}.$$

In this case, the size of the grounded credal network is proportional to $|\mathcal{D}|^2$; that is, it is quadratic on domain size.

It was not possible to run exact inference in this ontology with big domains, but the *L2U* algorithm [34] produced approximate inferences with reasonable computational cost. Table 1 shows some results for a growing domain. In Figure 2 we can see the size of the network generated for different domain sizes: in Fig.2(a) the domain size is 1 while in Fig.2(b) the domain size is 10.

Table 1. Results from the L2U algorithm for the inference $P\left(Parent\left(0\right) \mid Human\left(1\right)\right)$ for various domain sizes

N	2	5	10	20	30	40	50
L2U	0.2232	0.3536	0.4630	0.5268	0.5377	0.5396	0.5399

5 Conclusion

In this paper we have continued our efforts to develop a probabilistic description logic that can handle both probabilistic inclusions and queries containing Aboxes. This may seem a modest goal, but it touches on the central question concerning semantics in probabilistic logics; that is, whether the semantics is a domain-based or interpretation-based one. We have kept our preference for an interpretation-based semantics in this paper, as it seems to be the only way to avoid the challenges of direct inference. Without an interpretation-based semantics, it is hard to imagine how an inference involving Aboxes could be defined.

Most existing probabilistic description logics do adopt domain-based semantics, but it seems that the cost in avoiding inferences with Aboxes is high.

In this paper we have shown that the algorithms outlined in a previous publication [3] do scale up to realistic ontologies in the literature. Obviously, there is a trade-off between expressivity and complexity in any description logic, and it is difficult to know which features can be added to a description logic before making it intractable in practice. In this paper we have examined the challenges in adding nominals to the CR$\mathcal{ALC}$ logic. Nominals are both useful in practice, and interesting on theoretical grounds. The discussion of nominals can shed light on issues of semantics and direct inference, and one of the goals of this paper was to start a debate in this direction. We have presented relatively simple techniques that handle nominals in a limited setting; that is, as domains of restrictions. Much more work must be done before the behavior of nominals in probabilistic description logics becomes well understood. The inclusion of nominals intro CRALC, however limited, moves us towards the SHOIN logic, and therefore closer to OWL, the recommended standard for the Semantic Web. We hope to gradually close the remaining gap and a complete probabilistic version of OWL in future work

Acknowledgements

This work was partially funded by FAPESP (04/09568-0); the first author is supported by HP Brazil R&D.; the second author is partially supported by CNPq. We thank all these organizations.

References

1. F. Baader, D. Calvanese, D.L. McGuinness, D. Nardi, and P.F. Patel-Schneider. *Description Logic Handbook*. Cambridge University Press, 2002.
2. I. Horrocks, P. F. Patel-Schneider, and F. van Harmelen. From SHIQ and RDF to OWL: The making of a web ontology language. *Journal of Web Semantics*, 1(1):7–26, 2003.
3. F. G. Cozman, and R. B. Polastro. Loopy Propagation in a Probabilistic Description Logic. In *Int. Conf. on Scalable Uncertainty Management*, to appear, 2008.
4. M. Schmidt-Schauss and G. Smolka. Attributive concept descriptions with complements. *Artificial Intelligence*, 48:1–26, 1991.
5. K. Schild. A correspondence theory for terminological logics: Preliminary report. In *Int. Joint Conf. on Artificial Intelligence*, pages 466–471, 1991.
6. A. Borgida. On the relative expressiveness of description logics and predicate logics. *Artificial Intelligence*, 82(1-2):353–367, 1996.
7. J. Heinsohn. Probabilistic description logics. In *Conf. on Uncertainty in Artificial Intelligence*, page 311318, 1994.
8. M. Jaeger. Probabilistic reasoning in terminological logics. In *Principles of Knowledge Representation (KR)*, pages 461–472, 1994.

9. F. Sebastiani. A probabilistic terminological logic for modelling information retrieval. In W.B. Croft and C.J. van Rijsbergen, editors, *17th Annual Int. ACM Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 122–130, Dublin, Ireland, 1994. Springer-Verlag.
10. Z. Ding, Y Peng, and R. Pan. BayesOWL: Uncertainty modeling in semantic web ontologies. In *Soft Computing in Ontologies and Semantic Web*, volume 204 of *Studies in Fuzziness and Soft Computing*, pages 3–29. Springer, Berlin/Heidelberg, 2006.
11. M. Dürig and T. Studer. Probabilistic ABox reasoning: preliminary results. In *Description Logics*, pages 104–111, 2005.
12. R. Giugno and T. Lukasiewicz. P-SHOQ(D): A probabilistic extension of SHOQ(D) for probabilistic ontologies in the semantic web. In *Proceedings of the 8th European Conf. on Logics in Artificial Intelligence (JELIA)*, volume 2424, pages 86–97, Cosenza, Italy, September 2002. Lecture Notes in Artificial Intelligence, Springer.
13. D. Koller and A. Pfeffer. Object-oriented Bayesian networks. In *Conf. on Uncertainty in Artificial Intelligence*, pages 302–313, 1997.
14. T. Lukasiewicz. Expressive probabilistic description logics. *Artificial Intelligence*, to appear, 2008.
15. C. d'Amato, N. Fanizzi, T. Lukasiewicz Tractable Reasoning with Bayesian Description Logics. In Proceedings SUM-2008.
16. P. M. Yelland. Market analysis using a combination of Bayesian networks and description logics. Technical Report SMLI TR-99-78, Sun Microsystems Laboratories, 1999.
17. P. C. G. Costa and K. B. Laskey. PR-OWL: A framework for probabilistic ontologies. In *Conf. on Formal Ontology in Information Systems*, 2006.
18. F. Bacchus. *Representing and Reasoning with Probabilistic Knowledge: A Logical Approach*. MIT Press, Cambridge, 1990.
19. H. E. Kyburg Jr. and C. M. Teng. *Uncertain Inference*. Cambridge University Press, 2001.
20. L. Getoor and B. Taskar. *Introduction to Statistical Relational Learning*. MIT Press, 2007.
21. B. Milch and S. Russell. First-order probabilistic languages: into the unknown. In *Int. Conf. on Inductive Logic Programming*, 2007.
22. R. Staker. Reasoning in expressive description logics using belief networks. In *Int. Conf. on Information and Knowledge Engineering*, pages 489–495, Las Vegas, USA, 2002.
23. P. C. G. da Costa and K. B. Laskey. Of Klingons and starships: Bayesian logic for the 23rd century. In *Conf. on Uncertainty in Artificial Intelligence*, 2005.
24. H. Nottelmann and N. Fuhr. Adding probabilities and rules to OWL lite subsets based on probabilistic datalog. *Int. Journal of Uncertainty, Fuzziness and Knowledge-based Systems*, 14(1):17–42, 2006.
25. E. Hung, L. Getoor, and V. S. Subrahmanian. Probabilistic interval XML. *ACM Transactions on Computational Logic*, 8(4), 2007.
26. T. Lukasiewicz and U. Straccia. Managing uncertainty and vagueness in description logics for the semantic web. *submitted*, 2008.
27. T. Lukasiewicz. Probabilistic logic programming. In *European Conf. on Artificial Intelligence*, pages 388–392, 1998.
28. M. Jaeger. Relational Bayesian networks. In Dan Geiger and Prakash Pundalik Shenoy, editors, *Conf. on Uncertainty in Artificial Intelligence*, pp. 266–273, San Francisco, California, 1997. Morgan Kaufmann.

29. F. G. Cozman. Credal networks. *Artificial Intelligence*, 120:199–233, 2000.
30. M. Smith, C. Welty, and D. McGuiness. OWL Web Ontology Language Guide (W3C Recommendation), 2004.
31. K. A. Andersen and J. N. Hooker. Bayesian logic. *Decision Support Systems*, 11:191–210, 1994.
32. F. G. Cozman, C. P. de Campos, and J. C. Ferreira da Rocha. Probabilistic logic with independence. *Int. Journal of Approximate Reasoning*, in press, (available online 7 September 2007, doi: 10.1016/j.ijar.2007.08.002).
33. C. P. de Campos, F. G. Cozman, and J. E. Ochoa Luna. Assessing a consistent set of sentences in relational probabilistic logic with stochastic independence. *Journal of Applied Logic*, to appear, 2008.
34. J. S. Ide and F. G. Cozman Approximate algorithms for credal networks with binary variables. Int. Journal of Approximate Reasoning, 48(1):275-296, 2008.