Optimization and Evaluation of Reasoning in Probabilistic Description Logic: Towards a Systematic Approach

Pavel Klinov and Bijan Parsia

The University of Manchester
Manchester M13 9PL, UK

Abstract. This paper describes the first steps towards developing a methodology for testing and evaluating the performance of reasoners for the probabilistic description logic P-$\mathcal{SHIQ}$(D). Since it is a new formalism for handling uncertainty in DL ontologies, no such methodology has been proposed. There are no sufficiently large probabilistic ontologies to be used as test suites. In addition, since the reasoning services in P-$\mathcal{SHIQ}$(D) are mostly query oriented, there is no single problem (like classification or realization in classical DL) that could be an obvious candidate for benchmarking. All these issues make it hard to evaluate the performance of reasoners, reveal the complexity bottlenecks and assess the value of optimization strategies. This paper addresses these important problems by making the following contributions: First, it describes a probabilistic ontology that has been developed for the real-life domain of breast cancer which poses significant challenges for the state-of-art P-$\mathcal{SHIQ}$(D) reasoners. Second, it explains a systematic approach to generating a series of probabilistic reasoning problems that enable evaluation of the reasoning performance and shed light on what makes reasoning in P-$\mathcal{SHIQ}$(D) hard in practice. Finally, the paper presents an optimized algorithm for the non-monotonic entailment. Its positive impact on performance is demonstrated using our evaluation methodology.

1 Introduction

Probabilistic description logic P-$\mathcal{SHIQ}$(D) has been proposed to handle uncertainty in OWL ontologies [1]. Such formalisms have received significant research attention over the latest years, strongly driven by BioHealth and Semantic Web applications. In general, the capability of representing uncertain knowledge does not come for free: some extra reasoning complexity is usually incurred (not to mention various modeling difficulties) [2]. This problem is complicated because even classical DL reasoning is known to be worst case intractable for expressive languages, e.g., $\mathcal{SHIQ}$(D). Thus, optimization strategies are required to make the reasoning practical in real-life applications.

Optimization research can hardly be fruitful without a systematic evaluation methodology and reasonably characteristic test data. Unfortunately, there were few, if any, tools for developing or using P-$\mathcal{SHIQ}$(D) ontologies, thus no modelers have used it, and thus there are no applications using such ontologies and, indeed, no such ontologies at all. This makes the optimization research unguided and the principled comparison of different reasoning algorithms, implementations and approaches nearly impossible.

A. Sheth et al. (Eds.): ISWC 2008, LNCS 5318, pp. 213–228, 2008.

Another difficulty is the lack of reasoning problems that can be easily used for benchmarking, like, for example, classification problem in classical DL. That is, we can treat classification time as a reasonable proxy for the efficacy of reasoner optimizations (at least, as a first approximation). Conversely, P-$\mathcal{SHIQ}$(D) reasoning services are mostly query-oriented and focused on individual, antecedently given, entailments. We address this problem by the generation of queries against a bespoke ontology such that both the ontology and the queries are sensible from an application perspective.

This paper presents the first steps towards a *systematic* evaluation methodology for P-$\mathcal{SHIQ}$(D) by making the following contributions:

1. It describes a custom P-$\mathcal{SHIQ}$(D) ontology about breast cancer which we believe is a solid starting point for evaluating P-$\mathcal{SHIQ}$(D) implementations. Breast cancer risk assessment (BRCA) is a rich field with several general models, e.g., Gail model [3], and a wealth of online information and risk calculators. Thus, there are both clear statements to be formalized and deployed applications that can be used for determining characteristic queries. The ontology we developed, though not large, is very challenging to reason with. We believe that reasoners that can handle this ontology will work for an interesting range of applications.
2. It proposes a methodology for generating P-$\mathcal{SHIQ}$(D) reasoning problems including fragments of probabilistic ontologies with a series of probabilistic queries for each. The methodology has been implemented in the library PREVAL-DL[1] and applied to the BRCA ontology. The results are presented and discussed.
3. It demonstrates the utility of the methodology by evaluating the optimization strategy of lexicographic entailment in P-$\mathcal{SHIQ}$(D) that is now implemented in the new version of Pronto[2] [4]. The results clearly show both positive impacts of the strategy and the remaining issues.

The remainder of the paper is organized as follows: Section 2 briefly provides preliminaries on P-$\mathcal{SHIQ}$(D) as a representation and reasoning formalism. Section 3 describes the modeling of the BRCA ontology, the approach to generating the reasoning problems including probabilistic models and queries. It also presents the results of evaluating Pronto that help to understand the complexity of P-$\mathcal{SHIQ}$(D) in general. Section 4 sketches the developed optimization strategy and discusses the results of its evaluation using the new approach. Finally, the future work in this line is delineated in Section 5.

2 Technical Preliminaries on P-$\mathcal{SHIQ}$(D)

2.1 Syntax and Semantics of P-$\mathcal{SHIQ}$(D)

The syntactic constructs of P-$\mathcal{SHIQ}$(D) include those of $\mathcal{SHIQ}$(D) together with *conditional constraints*. Constraints are expressions of the form $(D|C)[l, u]$ where D, C are $\mathcal{SHIQ}$(D) concept expressions (called *conclusion* and *evidence* respectively) and $[l, u] \subseteq [0, 1]$ is a closed interval. Constraints can be *default* or *strict* corresponding to

[1] PREVAL-DL is an open source framework for testing and evaluating P-$\mathcal{SHIQ}$(D) reasoners: http://www2.cs.man.ac.uk/k̃linovp/projects/prevaldl/index.html

[2] Pronto 0.2: http://pellet.owldl.com/pronto

statements that are *generally* or *always* true respectively. Informally, default statements represent (probabilistic) knowledge that is true most of the time but might not apply in specific cases since details about the specific cases alters the probabilities. For example, we might have a general sense of the probability of the flu in the general population (say, low), whereas a subpopulation (say, old people and children) are more vulnerable thus have a higher probability of having the flu. There also could be a subsubpopulation (say, immunized old people and children) which has a very low probability of flu infection. P-$\mathcal{SHIQ}$(D) allows us to represent this situation using default statements.

A probabilistic TBox (PTBox) is a 2-tuple $PT = (T, P)$ where T is a classical DL TBox and P is a finite set of *default* conditional constraints (or just *defaults*). Informally, a PTBox axiom $(D|C)[l, u]$ means that "*generally*, if a randomly chosen individual belongs to C, its probability of belonging to D is in $[l, u]$". A probabilistic ABox (PABox) is a finite set of *strict* conditional constraints pertaining to a single probabilistic individual o [1]. All constraints in a PABox are of the restricted form $(D|\top)[l, u]$. Informally, they mean that "the individual o is a member of D with probability between $[l, u]$" [1]. A probabilistic knowledge base PKB is a combination of one PTBox and a set of PABoxes, one for each probabilistic individual.

The semantics of P-$\mathcal{SHIQ}$(D) is standardly explained in terms of the notion of a *possible world* which is a somewhat non-standard to DL and is defined with respect to a DL vocabulary (set of basic concepts) Φ [5]. A possible world I is a set of DL concepts from Φ such that $\{a : C|C \in I\} \cup \{a : \neg C|C \notin I\}$ is satisfiable for a fresh individual a. The set of all possible worlds with respect to Φ is denoted as $\mathcal{I}_\Phi$. A world I satisfies a concept C denoted as $I \models C$ if $C \in I$. Satisfiability of basic concepts is inductively extended to complex concepts as usual.

A world I is said to be a model of a DL axiom Ax denoted as $I \models Ax$ if $Ax \cup \{a : C|C \in I\} \cup \{a : \neg C|C \notin I\}$ is satisfiable for a fresh individual a. A world I is a model of a classical DL knowledge base KB denoted as $I \models KB$ if it is a model of all axioms of KB. Existence of a world that satisfies KB is equivalent to the satisfiability in the classical model-theoretic DL semantics [5].

We define probabilistic models in terms of the possible world semantics. A probabilistic interpretation Pr is a function $Pr : \mathcal{I}_\Phi \to [0, 1]$ such that $\sum_{I \in \mathcal{I}_\Phi} Pr(I) = 1$. Pr is said to *satisfy* a DL knowledge base KB denoted as $Pr \models KB$ iff $\forall I \in \mathcal{I}_\Phi, Pr(I) > 0 \Rightarrow I \models KB$. Next, the probability of a concept $C \in \Phi$, denoted as $Pr(C)$, is defined as $\sum_{I \models C} Pr(I)$. $Pr(D|C)$ is used as an abbreviation for $Pr(C \cap D)/Pr(C)$ given $Pr(C) > 0$. A probabilistic interpretation Pr satisfies a conditional constraint $(D|C)[l, u]$, denoted as $Pr \models (D|C)[l, u]$, iff $Pr(C) = 0$ or $Pr(D|C) \in [l, u]$. Finally, Pr satisfies a set of conditional constraints F iff it satisfies each of the constraints. A PTBox $PT = (T, P)$ is called *satisfiable* iff there exists a probabilistic interpretation that satisfies $T \cup P$.

A conditional constraint $(D|C)[l, u]$ is a *logical consequence* of a TBox T and a set of conditional constraints P, denoted as $T \cup P \models (D|C)[l, u]$, if $\forall Pr : Pr \models T \cup P \Rightarrow Pr(D|C) \in [l, u]$. It is a *tight logical consequence* of $T \cup P$ denoted as $T \cup P \models_{tight} (D|C)[l, u]$ if $l = inf_{Pr(C)>0 \wedge Pr \models T \cup P}(Pr(D|C))$ and $u = sup_{Pr(C)>0 \wedge Pr \models T \cup P}(Pr(D|C))$.

2.2 Reasoning in P-$\mathcal{SHIQ}$(D)

Lehmann's lexicographic entailment has been suggested as a non-monotonic consequence relation for P-$\mathcal{SHIQ}$(D) because of satisfying certain properties that are desirable for default reasoning [6] [7]. A few definitions are required to formulate it:

- A probabilistic interpretation Pr *verifies* a default $(D|C)[l,u]$ iff $Pr(C) = 1$ and $Pr(D|C) \in [l,u]$.
- Pr *falsifies* a default $(D|C)[l,u]$ iff $Pr(C) = 1$ and $Pr(D|C) \notin [l,u]$.
- A default d is *tolerated* by a set of defaults P under a classical TBox T iff $\exists Pr$: $Pr \models T \cup P$ and Pr verifies d.
- d *is in conflict* with P under T iff it is not tolerated by P under T.
- A default ranking σ is *admissible* for PTBox $PT = (T,P)$ iff $\forall P' \subseteq P, \forall d \in P$, d is in conflict with P' under $T \Rightarrow \exists d' \in P'$ $s.t.$ $\sigma(d') < \sigma(d)$.
- A PTBox is called *consistent* iff an admissible default ranking exists [7].

An admissible default ranking, if one exists, can be computed in the form of an ordered partition $\{P_i\}_{i=1}^k$ known as a *z-partition*. When using lexicographic entailment, those models that satisfy more defaults with higher ranks are considered *lexicographically preferable*. Models such that no other model is lexicographically preferable to them are called *lexicographically minimal*. A conditional constraint $(D|C)[l,u]$ is a *lexicographic consequence* of a PTBox $PT = (P,T)$ and a set of conditional constraints F if it is satisfied by every lexicographically minimal model of $F \cup PT$. It is a *tight* lexicographic consequence iff l (resp. u) is a minimum (resp. maximum) subject to all lexicographically minimal models [7].

It has been shown that lexicographically minimal models can be characterized via *lexicographically minimal sets* of conditional constraints [5]:

Definition 1 (Lexicographically minimal sets). *Given a consistent PTBox* $PT = (T,P)$ *with a z-partition* $\{P_i\}_{i=1}^k$ *and a set of conditional constraints* $\mathcal{F}$*, a set* $P' \subseteq P$ *is lexicographically preferable to* $P'' \subseteq P$ *given* $\mathcal{F}$ *iff:*

$$(T, P' \cup \mathcal{F}) \textit{ and } (T, P'' \cup \mathcal{F}) \textit{ are satisfiable.} \tag{1}$$

$$\textit{For some } i = \{1..k\}, |P' \cap P_i| > |P'' \cap P_i| \tag{2}$$

$$\textit{For all } j = \{i+1..k\}, P' \cap P_i = P'' \cap P_i. \tag{3}$$

The set $P' \subseteq P$ *given* $\mathcal{F}$ *is lexicographically minimal iff no* $P'' \subseteq P$ *is lexicographically preferable to* P' *given* $\mathcal{F}$.

The set of all lexicographically minimal sets of PTBox PT *given* $\mathcal{F}$ *is denoted* $LMS(PT, \mathcal{F})$

Informally, lexicographic entailment corresponds to standard logical entailment from lexicographically minimal sets. Computing $LMS(PT, F)$ is the first phase of computing the entailment. Section 4 will explain how that step can be optimized and will also present the evaluation of the proposed optimization.

The following are the core reasoning problems of P-$\mathcal{SHIQ}$(D) [7]:

- *Probabilistic Satisfiability (PSAT).* PSAT is the problem of deciding whether exists a probabilistic interpretation that satisfies given PTBox.
- *Probabilistic Generic Consistency (PGCon).* PGCon is the problem of deciding whether an admissible default ranking exists for the given PTBox.
- *Tight Logical Entailment (TLogEnt).* TLogEnt is the problem of computing the tightest probability intervals for logical consequences.
- *Tight Lexicographic Entailment (TLexEnt).* TLexEnt is the problem of computing the tightest probability intervals for lexicographic consequences.

3 Performance Evaluation Methodology

Probabilistic deduction in general and lexicographic entailment in P-$\mathcal{SHIQ}$(D) in particular are known to be computationally hard [2] [5]. Both PSAT and TLexEnt problems in P-$\mathcal{SHIQ}$(D) are EXPTIME-Complete where hardness follows from the complexity of $\mathcal{SHIQ}$(D) [8] and completeness from the small model theorem for satisfiability problem in probabilistic first-order logic [2].

These theoretical results do not necessarily say much about the practicality of reasoning in P-$\mathcal{SHIQ}$(D). It is known that even harder tableau-based algorithms for classical DL can be successfully used in applications. However, the picture is much less clear with respect to P-$\mathcal{SHIQ}$(D). It has been recently shown that reasoning tasks in P-$\mathcal{SHIQ}$(D) require a massive amount of classical DL reasoning, namely, classical SAT instances to be solved [5] [4]. At the same time the number of SATs varies greatly over probabilistic inputs so that the distribution required deeper investigation.

In this paper we use present a systematic approach to performance evaluation that is based on random sampling. Both, fragments of probabilistic ontology (samples) and probabilistic queries will be randomly generated. The main dataset for sampling will be a probabilistic ontology for breast cancer risk assessment (BRCA).

3.1 The BRCA Ontology

The BRCA ontology [3] was created as an attempt to model the problem of breast cancer risk assessment in a clear, ontological manner. The central idea behind the design the ontology was to reduce risk assessment to probabilistic entailment in P-$\mathcal{SHIQ}$(D).

The ontology consists of two major parts: a classical OWL ontology and a probabilistic part that represents domain uncertainty. It is anticipated that extensive medical vocabularies will be used as classical parts of such models. To emphasize this possibility in our experiments, we used the NCI thesaurus[4] augmented with a collection of classes to represent the risk factors used by the NCI risk calculator. The thesaurus is a large medical ontology of more than 27,500 classes.

The ontology aims at modeling two types of risk of developing breast cancer. First, it models *absolute* risk, i.e., the risk that can be measured without reference to other categories of women. Statements like "*an average woman has up to 12.3% of developing*

[3] Available at: http://www2.cs.man.ac.uk/klinovp/pronto/brc/cancer_cc.owl

[4] http://www.mindswap.org/2003/CancerOntology/nciOncology.owl

breast cancer in her lifetime" are examples of absolute risk [9]. Such risk is modeled using subclasses of $WomanUnderAbsoluteBRCRisk$. Subclasses distinguish between the risk of developing cancer over a lifetime vs. in the short term (e.g., ten years).

Second, the ontology models relative breast cancer risk. This is useful for representing the impact of various risk factors by describing how they increase or decrease the risk compared to an average woman. Statements like "*having BRCA1 gene mutation increases the risk of developing breast cancer by a factor of four*" express relative risk [9]. The ontology provides classes for different categories of relative risk, e.g., for increased risk or decreased risk.

The ontology defines risk factors that are relevant to breast cancer using subclasses of $RiskFactor$. It makes the distinction between the factors that should be known to a woman, e.g., age, family cancer history, breastfeeding and those that can only be inferred on the basis of other factors or by examination, e.g., BRCA gene mutation, breast and bone densities, etc. It also defines different categories of women: first, those that have certain risk factors (subclasses of $WomanWithRiskFactors$); and, second, those distinct in terms of the risk of developing cancer (subclasses of $WomanUnderBRCRisk$).

With this classical ontology, it is possible to define the task of assessing the risk in terms of probabilistic entailment. The problem is to compute the conditional probability that a certain woman is an instance of some subclass of $WomanUnderBRCRisk$ given probabilities that she is an instance of some subclasses of $WomanWithRiskFactors$. This requires probabilistic entailment of PABox axioms. In addition, it might also be useful to infer the generic probabilistic relationships between classes under $WomanUnderBRCRisk$ and under $WomanWithRiskFactors$. This can be done by computing TLexEnt for the corresponding PTBox axioms.

Following the assumption that the subjective probabilities representing risk factors for a certain individual can be combined with objective probabilities representing the statistical knowledge, the model contains a set of PABox and PTBox axioms. The PABox axioms define risk factors that are relevant to a particular individual. The PTBox axioms model generic probabilistic relationships between classes in the ontology, i.e., those that are assumed to hold for a randomly chosen individual.

The model represents absolute risk using the subclasses of $WomanUnderAbsolute$ $BRCRisk$ as conclusions in conditional constraints. For example, the above statement that an average woman has risk up to 13.2% can be expressed as the following TBox axiom:

$$(WomanUnderAbsoluteBRCRisk|Woman)[0, 0.132].$$

Similarly, the model represents the impact of various risk factors by PTBox constraints with subclasses $WomanWithRiskFactors$ as evidence. For example, the influence of age can be represented by the following constraint:

$$(WomanWithBRCInShortTerm|Woman50Plus)[0.027, 0.041]$$

which expresses that a woman after the age of fifty has a certain risk of developing breast cancer in short term. Relative risk can be captured analogously by using the subclasses of $WomanUnderRelativeBRCRisk$ as conclusions. For example, the impact of BRCA gene mutation can be described as:

$$(WomanUnderStrongBRCRisk|WomanWithBRCAMutation)[0.9, 1]$$

which means that a woman having BRCA (BRCA1 or BRCA2) gene mutation is almost certainly in the highest risk category.

The model also allows one to express various inter-relationships between risk factors. One possibility is to represent how the presence of one risk factor allows one to guess on the presence of others. This is the principal way to use *inferred* risk factors, i.e., those unknown to a woman. For example, it is statistically true that Ashkenazi Jews are more likely to develop the BRCA gene mutation [9]. Although the person being questioned may not be aware of her chances of having a gene mutation, they can be estimated based on her ethnicity or other factors. Such relationships are captured using the PTBox constraints with evidence and conclusions being subclasses of $Woman$ or $WomanUnderBRCRisk$, such as:

$$(WomanWithBRCAMutation|AshkenaziJewishWoman)[0.025, 0.025]$$

In addition, the model allows to represent how different risk factors strengthen or weaken each other. The classical part of the ontology provides classes that are combinations of multiple risk factors. For example, $Woman50PlusMotherBRCA$ is a subclass of both $WomanAged50Plus$ and $WomanWithMotherBRCA$, i.e., it represents women after the age of 50 whose mothers developed breast cancer in the past. The model can define the risk for such women to be much higher than if they had just one of the factors. This is possible using the previously described *overriding* feature. Informally, PTBox axioms for the combination of factors, such as:

$$(WomanUnderStrongBRCRisk|Woman50PlusMotherBRCA)[0.9, 1]$$

overrides the axioms for each individual factor, thus allowing the system to make a more relevant and objective inference. It is theoretically possible to define an exponential number of such risk factor combinations but in practice only some of them require special attention.

Finally, the ontology contains a number of PABoxes that represent risk factors for specific individuals. The motivation is that while the generic probabilistic model that provides all the necessary statistics that can be developed and maintained by a central cancer research institute, individual women can supply the knowledge about the risk factors that are known to them, e.g., age. It is also possible to express uncertainty in having some particular risk factor. This is particularly important for inferred risk factors, for example, breast or bone density.

3.2 Random Sampling

Given the test data (BRCA ontology) the next step is to generate instances of reasoning problems to evaluate the performance. We chose to generate instances of PSAT and TLexEnt where TLexEnt also includes PGCon as a sub-problem.

Currently the full version of BRCA ontology cannot be handled by P-$\mathcal{SHIQ}$(D) reasoners mainly because linear system even for a single PSAT becomes too large (i.e., exponential in the number of conditional constraints). Therefore we decided to evaluate the performance on selected fragments of the ontology. As mentioned above, the performance varies significantly over fragments and it was originally unclear which fragments are "hard" and which are "easy". Thus it was natural to begin with the random sampling method.

In all the following experiments, the performance (or hardness) is measured in the number of classical DL SAT instances that need to be solved during probabilistic reasoning. This helps to abstract from platform-dependent metrics such as time.

Instances of PSAT have been generated using *simple random sampling*. Each sample was an independent probabilistic KB with the full classical part of the BRCA ontology and a subset of the PTBox constraints. The number of conditional constraints varied from 10 to 15 to maintain the balance between the size of each sample and the number of trials for each size. The latter was 200.

Instances of TLexEnt are less straighforward to generate. First note that entailments of PABox constraints are usually harder than PTBox because of interactions between default PTBox knowledge and strict PABox knowledge during non-monotonic reasoning. Simple random samples of PKB are insufficient for generating PABox queries. It is also required to have a probabilistic individual with PABox costraints. For example, in the case of BRCA ontology, such individual would be a woman with her personal probabilistic facts (risk factors that apply to her).

Such individual can be selected from the collection of predefined PABoxes (analogously to selecting a fragment of PTBox). But in this case it is hard to ensure the interaction between a randomly selected fragment of PTBox and a independently selected probabilistic individual. Intuitively, it is desirable to generate realistic problem instances so that the strict knowledge about the individual can be usefully combined with the statistical knowledge in the PTBox. Again, in the case of BRCA ontology, there should be PTBox constraints that represent statistics about the risk factors that are relevant to some probabilistic individual. Otherwise the latter are useless for assessing the breast cancer risk.

Our approach to generating such reaslistic TLexEnt instances is summarized by the following steps:

- Generate fragments of the PTBox using simple random sampling.
- Generate a probabilistic individual and the corresponding PABox. Each PABox constraint $(C|\top)[l, u]$ is generated such that C is a class appearing in some of the previously selected PTBox constraints and $[l, u]$ is a random interval.
- Generate a PABox query of the form $(C|\top)[?, ?]$ where C is selected from a domain-specific set of classes. In BRCA that set includes classes that represent women under absolute or relative breast cancer risk.

The effect of the first two steps is that the reasoner has to consider both PTBox and PABox constraints during reasoning, instead of eliminating some as irrelevant (which might have been the case if they had been generated completely independently). The third step ensures that the queries will be meaningful in that particular domain.

3.3 Results

We have applied the methodology to the latest version of Pronto. As expected it was observed that the hardness of PSAT grows exponentially with the number of conditional constraints. The interesting fact was that the exponential blowup did not happen in all cases. Moreover, some samples of size k (*k-samples*) happened to be easier that some samples of size $k-1$. For example, for $k = 10$, the number of SATs varied from 699 to $14,200$ whereas for $k = 11$ it varied from $1,091$ to $38,522$. Such variation is important to investigate in order to understand what exactly makes probabilistic KBs hard or easy for reasoning. This might lead to developing reasoning algorithms that can exploit such characteristics of PKBs.

The lower bound on the number of needed SATs is the number of variables in the linear system generated during PSAT. Each variable corresponds to some world and a SAT should be solved in order to show that the world is *possible*, i.e., satisfies the classical part of the KB. Thus it is natural to investigate how the number of variables (or size of the *index set* [5]) varies over the random samples and what factors have an impact on it.

The variation of the index set size is similar to the variation of the number of SAT as expected: for 10-samples the minimal size was 447 and maximal was $8,064$. The more interesting problem is to identify what factors determine the size of the index set. Then it would be possible to assess hardness of samples *in advance* and potentially exploit this information during reasoning.

With this aim in mind we attempted to develop a metric for estimating hardness of a PTBox. It can be conjectured that the size of the index set should depend on the number of relations (e.g., subsumption, disjointness, etc.) that can be proven for classes appearing in conditional constraints [1]. In the extreme case, if no such relation exists, the size would be 3^N where N is the number of constraints [1]. In practice, however, many index set items can correspond to classical models that do not satisfy classical part of KB and should be pruned. As an example, consider TBox $T = \{Penguin \sqsubseteq Bird\}$ and the world $\{Penguin, \neg Bird\}$. Clearly this world is not possible. Thus the metric should reflect the number of such relations between classes in constraints which we call the *connectivity* of PTBox). We have implemented and experimented with this metric by computing, for each pair of constraints, the number of subsumptions between classes and their negations (i.e. 9 SAT tests for each pair of constraints). The results for 200 samples were compared with the actual hardness of PTBox, i.e., the number of SATs solved during PSAT, in Figure 1.

The results show the anticipated correspondence between the connectivity of PTBox and its actual hardness which means that some prediction of reasoning complexity can be done in advance. It is an interesting question whether the *phase transition* phenomenon [10] can be observed for PSAT. Phase transition is a property of many known NP-hard problems which says that the hardest instances are groupped in a relatively small region of the problem space which is characterized by a critical value of some order parameter. For example, for SAT in propositional logic such parameter would be the average number of literals in clauses. So around the critical value there is a transition from the set of underconstrained problems to overconstrained ones. Reasonable algorithms are often capable of solving most of the problems that do not fall into the

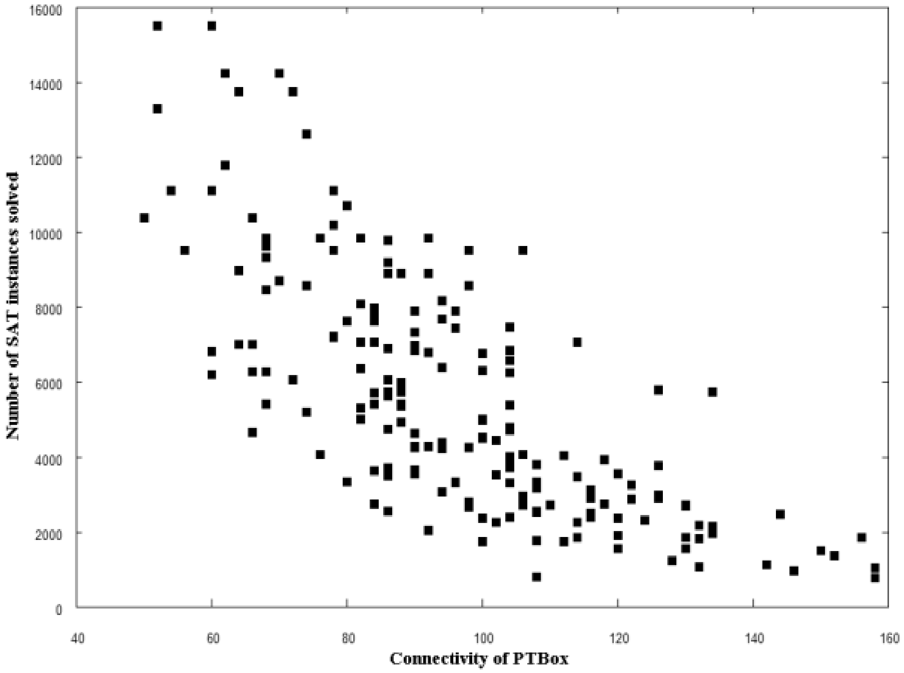

Fig. 1. Performance of PSAT plotted against the predicted hardness of PTBox

hard region efficiently, for example, such problems as 3-SAT, graph coloring, etc. are often tractable in practice.

The diagrams above do not show typical phase transition pattern although the connectivity metric is related to the extent to which a given instance is constrained. Less connectivity means that the classes in the conditional constraints are weakly related to each other so that the chance of conflicts is small. This is similar for underconstrained instances of SAT in propositional logic. Similarly, highly connected instances of PSAT are overconstrained. Thus it is reasonable to expect that some sort of phase transition phenomenon would occur. Why does it not happen?

The answer is that the PSAT algorithm [7] does not exploit the heuristical estimation of hardness in any way. Differently from many known algorithms for NP-complete problems it does not try to quickly find a solution for an underconstrained problem or quickly prove inexistence of solutions for an overconstrained problem. This might be one possible reason why PSAT is intractable for P-$\mathcal{SHIQ}$(D), and thus can be a promising direction for the optimization research on P-$\mathcal{SHIQ}$(D). More sophisticated evaluation techniques may need to be developed to support or falsify this conjecture.

The results for TLexEnt look similar to the results for single PSAT. The same metric proved to be predictive for a different problem. This is natural to expect because complexity of TLexEnt strongly depends on the complexity of PSAT which is its subproblem. The results plotted on the Figure 2 (again 200 samples were taken).

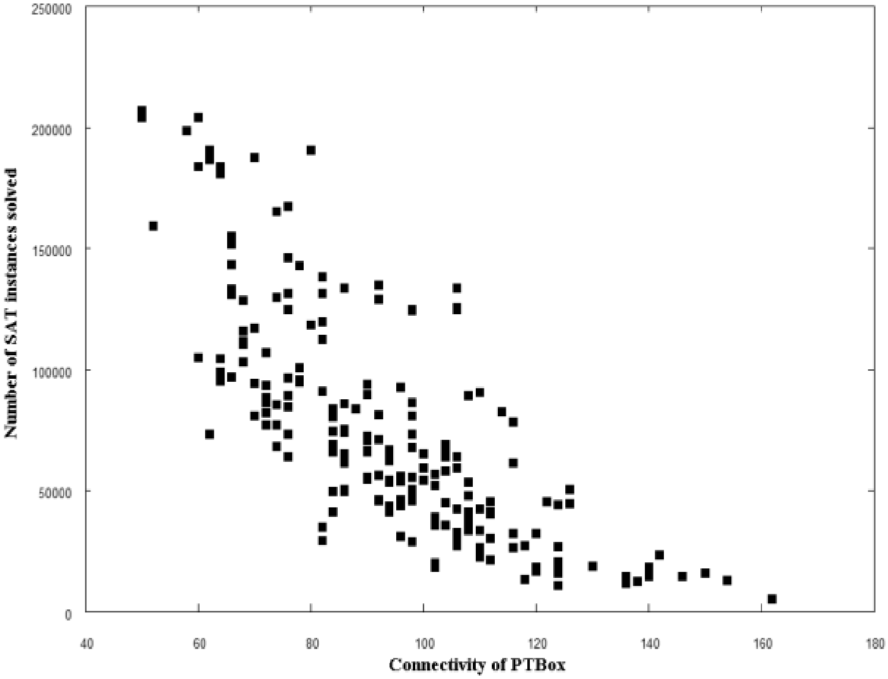

Fig. 2. Performance of TLexEnt plotted against the predicted hardness of PTBox

The important outcome is that for the latest TLexEnt algorithm (see Section 4 for details) there do not seem to be other factors except PSAT that affect its complexity. Interestingly this is not the case for the original algorithms that are due to Lukasiewicz [1] [5]. The same evaluation methodology can show that the original algorithm performs on some PKBs much worse than predicted by the metric. The reason is that the naive computation of lexicographically minimal sets during the non-monotonic phase of reasoning causes too many PSATs to be solved.

4 Evaluating Optimization Strategies

This section will demonstrate how new optimization strategies can be evaluated and compared to th existing algorithms using the proposed evaluation methodology. We start by briefly describing the optimization technique for computing lexicographically minimal models during TLexEnt.

4.1 Optimized TLexEnt Algorithm

The original TLexEnt algorithm computes the tightest interval for probabilistic query $(D|C)[?, ?]$ in two phases [7] [1]:

1. *Model selection*. Conclusions in P-$\mathcal{SHIQ}$(D) are drawn from the set of lexicographically minimal models that are selected by computing lexicographically minimal sets (*LM-sets*) of constraints (see Definition 1).
2. *Entailment from preferred models*. Once models have been selected, the tightest interval can be computed by performing linear optimizations.

The complexity of the first phase determines the overall complexity of TLexEnt. Models selection can be done by solving $O(e^N)$ instances of PSAT each of which requires $O(e^N)$ instances SAT. Such a high complexity is caused by an uninformed search for LM-sets that runs over the powerset of constraints [5]. This is avoided in the improved algorithm that proceeds by eliminating the *minimal conflicting subsets*.

Definition 2 (Minimal conflict sets). *For a set of conditional constraints $\mathcal{F}$ and PTBox $PT = (T, P)$, a conflict set of PT given $\mathcal{F}$ is a set of conditional constraints Q s.t. $Q \subseteq P$ and $(T, Q \cup \mathcal{F})$ is unsatisfiable.*

A conflict set Q of $PT = (T, P)$ given $\mathcal{F}$ is minimal if $\forall Q' \subset Q$, $(T, Q' \cup \mathcal{F})$ is satisfiable.

The set of all minimal conflict sets of PT given $\mathcal{F}$ is denoted as $MCS(PT, \mathcal{F})$.

Informally, conflict sets identify those fragments of a probabilistic ontology that require conflict resolution during default reasoning. See the example below:

Example 1. Consider the following PTBox

$$
\begin{aligned}
PT =& (\{Penguin \sqsubseteq Bird\}, \\
& \{(Fly|Bird)[0.9, 0.95], && (1)\\
& (Fly|Penguin)[0, 0.05], && (2)\\
& (Wings|Bird)[0.95, 1]\}) && (3)
\end{aligned}
$$

$MCS(PT, \{(Penguin|\top)[1, 1]\}) = \{1, 2\}$, but $MCS(PT, \{(Bird|\top)[1, 1]\}) = \{\}$

As it will be shown below, conflict sets can be very useful for computing LM-sets.

Computing Minimal Conflict Sets. Finding *all* MCS is an NP-complete problem, so it may seem that an exponential number of PSAT instances will need to be generated and solved. However, it turns out that it is necessary to generate only a *single* PSAT instance to find all MCS thus avoiding a double exponential number of classical SAT tests. At the same time, it may be required to check an exponential number of linear systems for solvability. Fortunately, that step is computationally easier as it does not involve any classical DL reasoning.

The idea is as follows: First, *some initial* MCS is found by repeatedly removing linear inequalities from the linear system corresponding to $(T, P \cup \mathcal{F})$. The resulting system contains only those inequalities that correspond to conflicting constraints in MCS. Then it is possible to employ a standard technique for computing all explanation sets in classical DLs [11]. Each next MCS can be found by eliminating some constraints from all the previous MCS from the original PTBox and repeating the process of removing inequalities. The entire process terminates when no further MCS can be found.

It can be seen that there is only a *single* PSAT instance is generated during the computation of the first MCS. All other MCS are discovered by performing operations on linear systems and do not require any SAT tests at all.

Computing Lexicographically Minimal Sets. As mentioned before, the main goal of the optimization is to avoid solving an exponential number of PSATs during the search for lexicographically minimal sets while solving TLexEnt. It appears that it can be done by using the idea of conflict sets to compute maximal satisfiable subsets of PTBox by generating only a linear number of PSATs.

Definition 3 (Maximal satisfiable subsets). *Given a PTBox $PT = (T, P)$ and a set of conditional constraints $\mathcal{F}$, set $R \subseteq P$ is the maximal satisfiable subset of PT given $\mathcal{F}$ iff $(T, R \cup \mathcal{F})$ is satisfiable but $(T, S \cup \mathcal{F})$ is not for every $S \subseteq P$ s.t. $R \subset S$.*

The set of all maximal satisfiable subsets of PT given $\mathcal{F}$ is denoted as $MSS(PT, \mathcal{F})$.

The crucial observation is that lexicographically minimal sets can be computed by iterating over the z-partition and computing MSS at each subset. More formally:

Lemma 1. *Given a consistent PTBox $PT = (T, P)$ with z-partition $\{P_0, ..., P_k\}$ and a set of constraints $\mathcal{F}$, $LMS(PT, \mathcal{F})$ is equivalent to the set of all unions $\bigcup_{i=0}^{k} M_i$ where $M_k \in MSS((T, P_k), \mathcal{F})$ and $M_i \in \{MSS((T, P_i), M_{i+1}) | MSS((T, P_i), M_{i+1})$ has subsets of maximal cardinality subject to all $M_{i+1}\}$*

Lemma 1 essentially describes the algorithm for computing $LMS(PT, \mathcal{F})$. It is sufficient to iterate over all subsets of the z-partition in the order of decreasing specificity and compute MSS at each subset of the partition. It only remains to show how to compute $MSS(PT, \mathcal{F})$. It is well known that maximal satisfiable subsets are related with minimal unsatisfiable subsets in the following sense [12]:

Lemma 2. *Given a PTBox $PT = (T, P)$ and a set of constraints $\mathcal{F}$, $MSS(PT, \mathcal{F})$ is the set of all $M \subseteq P$ s.t. for every M there exists a set H s.t. $M = P \setminus H$, $H \cap Q \neq \emptyset$ for all $Q \in MCS(PT, \mathcal{F})$ and for any $H' \subset H$ there exists $Q \in MCS(PT, \mathcal{F})$ s.t. $H' \cap Q = \emptyset$*

Such sets H are called *minimal hitting sets* in the literature. Lemma 2 states a known approach to computing MSS that is based on computing all minimal hitting sets for all minimal conflict sets and then removing them from the initial set [12].

Using this technique the optimized algorithm computes a set of MSS that is linear in the number of subsets in the z-partition. Each MSS can be reduced to the computation of MCS and minimal hitting sets over the MCS. The latter is a known NP-complete problem but fortunately it is limited in its size and does not involve any classical DL reasoning. So, this algorithm computes lexicographically minimal sets by generating only a linear number of PSATs as opposed to the exponential number required by the Lukasiewicz algorithm. A simple example illustrates the advantage:

Example 2. Consider the following PTBox:

$$
\begin{aligned}
PT = (&\{Penguin \sqsubseteq Bird\},\\
&\{(Fly|Bird)[0.9, 0.95], && (1)\\
&(Fly|Penguin)[0, 0.05], && (2)\\
&(Wings|Bird)[0.95, 1]\}) && (3)\\
\mathcal{F} = &\{(Penguin|\top)[1, 1]\}
\end{aligned}
$$

The z-partition is $\{\{1,3\},\{2\}\}$. Lukasiewicz's algorithm would compute $LMS(PT, \mathcal{F})$ in the following steps (* means that a PSAT instance has to be generated):

1. Check satisfiability of $(T, \mathcal{F})$*. Result: true.
2. Check satisfiability of $(T, \mathcal{F} \cup \{2\})$*. Result: true.
3. Check satisfiability of $(T, \mathcal{F} \cup \{2, 1, 3\})$*. Result: false.
4. Check satisfiability of $(T, \mathcal{F} \cup \{2, 1\})$*. Result: false.
5. Check satisfiability of $(T, \mathcal{F} \cup \{2, 3\})$*. Result: true.
6. $LMS(PT, \mathcal{F}) := \mathcal{F} \cup \{2, 3\}$

There are two negative PSAT tests that are avoided in the new algorithm:

1. Check satisfiability of $(T, \mathcal{F})$*. Result: true.
2. Compute $MSS((T, \{2\}), \mathcal{F})$*. Result: $\{2\}$
3. Compute $MSS((T, \{1, 3\}), \mathcal{F})$*. Result: $\{3\}$
4. $LMS(PT, \mathcal{F}) := \mathcal{F} \cup \{2, 3\}$

4.2 Evaluation of the Optimized Algorithm

The developed methodology enables us to systematically evaluate the performance of the optimized algorithm. The methodology can be applied to both algorithms and the results are easily comparable. This has been done by running both algorithms on random instances of TLexEnt generated as explained in the Section 3.2. We performed 200 runs where each PKB had 10 PTBox and 3 PABox constraints. The results are plotted on the Figure 3.

Simple visual comparison yields a few important observations. First, the new algorithm performs better as expected. Second, its behavior is more amenable to predictions using our connectivity metric. In other words, the relationship between the metric values and the actual hardness is apparent and resembles the same graph for PSAT. The naive algorithm, in contrast, produced a lot more outliers. There are some "hard" outliers — instances of TLexEnt that involve much more classical SATs than expected.

Finally, it can be noted that the fraction of such hard outliers is not large. This is a direct consequence of simple random sampling method which selects subsets of PTBox constraints with equal probability. Therefore, the chance that there will be conflicts similar to those shown in Example 2 is relatively small.

There is a question, however, whether such conflicts would be frequent in practice. At this point it is not fully clear because no P-$\mathcal{SHIQ}$(D) ontologies are employed in real applications. The BRCA ontology is the first attempt we know of to provide such model. In this ontology, conflicts can be expected because strict knowledge about particular women or their categories can often override general statistical knowledge. One example is African American and Ashkenazi Jew women for whom the statistical relationships from the Gail model are known to be imprecise or even incorrect.

In any case the evaluation methodology is useful because, first, it can systematically generate and run many random samples and therefore help to find "interesting cases", i.e., hard or easy outliers. Second, it can be used to compare different reasoning techniques and find inputs on which the techniques demonstrate similar or drastically different performance. At the same time it may be required to have a more intelligent problem generation method rather than random sampling. For example, a possible next step in the development of a benchmarking suite might be generation of only hard instances analogously to how it was done for other logics [13].

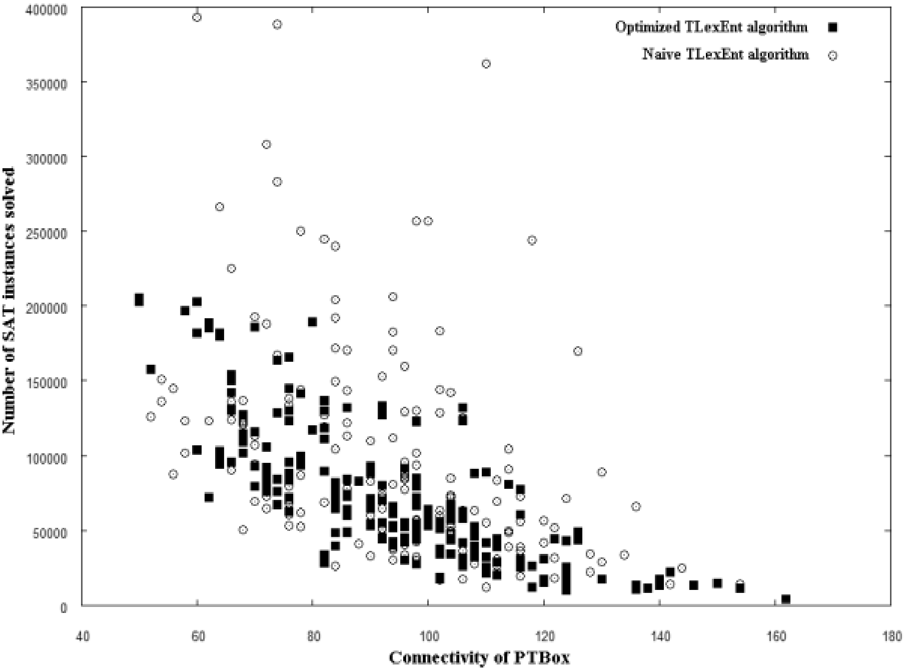

Fig. 3. Performance of TLexEnt plotted against the predicted hardness of PTBox. 200 runs.

5 Summary

The paper described first steps towards a systematic performance evaluation methodology for P-$\mathcal{SHIQ}$(D) reasoners. We have developed an approach to generating instances of the most important reasoning problems in P-$\mathcal{SHIQ}$(D) and provided a probabilistic ontology to serve as a basis for the generation. The methodology has been used to illustrate benefits of our optimizations for computing entailments.

Even though our approach is methodologically straighforward, to our knowledge, it has not been applied in this area before. Our experimental results show that being systematic in the evaluation of performances validates our analytical understanding of the reasoning tasks and algorithms but also yields important insights, such as the notion of connectivity for a set of conditional constraints and its impact on reasoning complexity.

The approach is flexible and extensible in the sense that one can contribute problem generators for their specific reasoning tasks. For example, as learned from the evaluation of the improved TLexEnt algorithm, a bias towards "hard" problem instances might be desirable. Also, there might be domain-specific evaluation. For instance, in the BRCA domain, it would be natural to generate PABoxes that only have constraints describing individual risk factors as opposed to randomly generated constraints. All such extensions can be smoothly plugged into the framework.

It is our expectation that the approach will also stimulate further reasoning optimization research for P-$\mathcal{SHIQ}$(D). The most important reasoning task to be optimized is

PSAT because it is currently responsible for the limited scalability of reasoners, e.g., Pronto. The evaluation strategy can highlight the problem instances on which the algorithm performs poorly so that specific optimization techniques might be developed to alleviate it. In this respect, current results can be considered as an important step towards practical reasoning in P-$\mathcal{SHIQ}$(D).

References

1. Giugno, R., Lukasiewicz, T.: $P{-}\mathcal{SHOQ}(D)$: A probabilistic extension of $\mathcal{SHOQ}$(D) for probabilistic ontologies in the semantic web. Technical Report Nr. 1843-02-06, Institut fur Informationssysteme, Technische Universitat Wien (2002)
2. Lukasiewicz, T.: Probabilistic logic programming with conditional constraints. ACM Transactions on Computational Logic 2(3), 289–339 (2001)
3. Gail, M.H., Brinton, L.A., Byar, D.P., Corle, D.K., Green, S.B., Shairer, C., Mulvihill, J.J.: Projecting individualized probabilities of developing breast cancer for white females who are being examined annually. Journal of the National Cancer Institute 81(25), 1879–1886 (1989)
4. Klinov, P.: Pronto: a non-monotonic probabilistic description logic reasoner. In: Proceeding of the European Semantic Web Conference (2008)
5. Lukasiewicz, T.: Expressive probabilistic description logics. Artificial Intelligence 172(6-7), 852–883 (2008)
6. Lehmann, D.: Another perspective on default reasoning. Annals of Mathematics and Artificial Intelligence 15(1), 61–82 (1995)
7. Lukasiewicz, T.: Probabilistic default reasoning with conditional constraints. Annals of Mathematics and Artificial Intelligence 34(1-3), 35–88 (2002)
8. Horrocks, I., Sattler, U., Tobies, S.: Practical reasoning for very expressive description logics. Journal of the IGPL 8(3) (2000)
9. Komen, S.G.: Breast cancer risk factors table (2007), Retrieved from: `http://cms.komen.org/Komen/AboutBreastCancer/`
10. Cheeseman, P., Kanefsky, B., Taylor, W.M.: Computational complexity and phase transitions. In: Proceedings of IJCAI, pp. 331–337 (1991)
11. Kalyanpur, A., Parsia, B., Horridge, M., Sirin, E.: Finding all justifications of OWL DL entailments. In: Proceedings of IJCAI, pp. 267–280 (2007)
12. Bailey, J., Stuckey, P.: Discovery of Minimal Unsatisfiable Subsets of Constraints Using Hitting Set Dualization. In: Hermenegildo, M.V., Cabeza, D. (eds.) PADL 2004. LNCS, vol. 3350, pp. 174–186. Springer, Heidelberg (2005)
13. Horrocks, I., Patel-Schneider, P.F.: Generating hard modal problems for modal decision procedures. In: First Methods for Modalities Workshop (1999)