INFSYS

RESEARCH

REPORT

INSTITUT FÜR INFORMATIONSSYSTEME

ARBEITSBEREICH WISSENSBASIERTE SYSTEME

NONMONOTONIC PROBABILISTIC LOGICS UNDER VARIABLE-STRENGTH INHERITANCE WITH OVERRIDING: ALGORITHMS AND IMPLEMENTATION IN NMPROBLOG

THOMAS LUKASIEWICZ

INFSYS RESEARCH REPORT 1843-05-03

MAY 2005; JUNE 2006

Institut für Informationssysteme
AB Wissensbasierte Systeme
Technische Universität Wien
Favoritenstraße 9-11
A-1040 Wien, Austria
Tel: +43-1-58801-18405
Fax: +43-1-58801-18493
sek@kr.tuwien.ac.at
www.kr.tuwien.ac.at

TU VIENNA

INFSYS RESEARCH REPORT

INFSYS RESEARCH REPORT 1843-05-03, MAY 2005; JUNE 2006

NONMONOTONIC PROBABILISTIC LOGICS UNDER VARIABLE-STRENGTH INHERITANCE WITH OVERRIDING: ALGORITHMS AND IMPLEMENTATION IN NMPROBLOG

JUNE 22, 2006

Thomas Lukasiewicz [1]

Abstract. In previous work, I have introduced nonmonotonic probabilistic logics under variable-strength inheritance with overriding. They are formalisms for probabilistic reasoning from sets of strict logical, default logical, and default probabilistic sentences, which are parameterized through a value $\lambda \in [0,1]$ that describes the strength of the inheritance of default probabilistic knowledge. In this paper, I continue this line of research. I give a precise picture of the complexity of deciding consistency of strength λ and of computing tight consequences of strength λ. I also present algorithms for these tasks, which are based on reductions to the standard problems of deciding satisfiability and of computing tight logical consequences in model-theoretic probabilistic logic. Finally, I describe the system NMPROBLOG, which includes a prototype implementation of these algorithms.

[1]Dipartimento di Informatica e Sistemistica, Università di Roma "La Sapienza", Via Salaria 113, I-00198 Rome, Italy; e-mail: lukasiewicz@dis.uniroma1.it. Institut für Informationssysteme, Technische Universität Wien, Favoritenstraße 9-11, A-1040 Vienna, Austria; e-mail: lukasiewicz@kr.tuwien.ac.at.

Acknowledgements: This work was supported by a Heisenberg Professorship of the German Research Foundation (DFG). I am thankful to the reviewers of the ISIPTA-2005 abstract of this paper for their constructive comments, which helped to improve this work.

Contents

1 Introduction

During the recent decades, reasoning about probabilities has started to play an important role in artificial intelligence. In particular, reasoning about interval restrictions for conditional probabilities, also called *conditional constraints* [38], has been a subject of extensive research efforts. Roughly, a conditional constraint is of the form $(\psi|\phi)[l, u]$, where ψ and ϕ are events, and $[l, u]$ is a subinterval of the unit interval $[0, 1]$. It encodes that the conditional probability of ψ given ϕ lies in $[l, u]$.

An important approach for handling conditional constraints is model-theoretic probabilistic logic, which has its origin in philosophy and logic, and whose roots can be traced back to already Boole in 1854 [13]. There is a wide spectrum of formal languages that have been explored in model-theoretic probabilistic logic, ranging from constraints for unconditional and conditional events to rich languages that specify linear inequalities over events (see especially the work by Nilsson [44], Fagin et al. [17], Dubois and Prade et al. [14, 15], Frisch and Haddawy [19], and the author [37, 38]). The main decision and optimization problems in model-theoretic probabilistic logic are deciding satisfiability, deciding logical consequence, and computing tight logically entailed intervals.

For example, a simple collection of conditional constraints KB may encode the *strict logical knowledge* "all eagles are birds" and "all birds have feathers" as well as the *purely probabilistic knowledge* "birds fly with a probability of at least 0.95". This KB is satisfiable, and some logical consequences in model-theoretic probabilistic logic from KB are "all birds have feathers", "birds fly with a probability of at least 0.95", "all eagles have feathers", and "eagles fly with a probability between 0 and 1"; in fact, these are the tightest intervals that follow from KB. That is, we especially cannot conclude anything from KB about the ability to fly of eagles.

A closely related research area is default reasoning from conditional knowledge bases, which consist of a collection of strict statements in classical logic and a collection of defeasible rules, also called defaults. The former must always hold, while the latter are rules of the kind $\psi \leftarrow \phi$, which read as "generally, if ϕ then ψ." Such rules may have exceptions, which can be handled in different ways.

The literature contains several different proposals for default reasoning from conditional knowledge bases and extensive work on its desired properties. The core of these properties are the rationality postulates of System P by Kraus, Lehmann, and Magidor [32], which constitute a sound and complete axiom system for several model-theoretic entailment relations under uncertainty measures on worlds. They characterize model-theoretic entailment under preferential structures, infinitesimal probabilities, possibility measures, and world rankings. As shown by Friedman and Halpern [18], many of these uncertainty measures on worlds are expressible as plausibility measures. See [7, 21] for a survey of the above relationships.

Mainly to solve problems with irrelevant information, the notion of rational closure as a more adventurous notion of entailment was introduced by Lehmann [35]. It is equivalent to entailment in System Z by Pearl [47] and to the least specific possibility entailment by Benferhat et al. [5]. Finally, mainly to solve problems with property inheritance from classes to exceptional subclasses, some more sophisticated notions of entailment were proposed, including the notion of lexicographic entailment by Lehmann [36] and Benferhat et al. [4].

For example, a conditional knowledge base KB may encode the *strict logical knowledge* "all ostriches are birds" and the *default logical knowledge* "generally, birds fly", "generally, ostriches do not fly", and "generally, birds have wings". Some desirable conclusions from KB [26] are "generally, birds fly" and "generally, birds have wings" (which both belong to KB), "generally, ostriches have wings" (since the set of all ostriches is a subclass of the set of all birds, and thus ostriches should inherit all properties of birds), "generally, ostriches do not fly" (since properties of more specific classes should override inherited properties of less specific classes), and "generally, red birds fly" (since "red" is not mentioned at all in KB and thus should be irrelevant to the ability to fly of birds).

There are several works in the literature on probabilistic foundations for default reasoning from conditional knowledge bases [1, 46, 12], on combinations of Reiter's default logic with statistical inference [34, 49], and on a rich first-order formalism for deriving degrees of belief from statistical knowledge including default statements [3]. A series of recent papers has proposed combinations of model-theoretic probabilistic reasoning from conditional constraints with default reasoning from conditional knowledge bases, which are summarized as follows:

- The paper [43] presents *weak nonmonotonic probabilistic logics*, which are extensions of probabilistic logic by defaults as in conditional knowledge bases under Kraus et al.'s System P [32], Pearl's System Z [47], and Lehmann's lexicographic entailment [36]. The new formalisms allow for expressing in a uniform framework *strict logical knowledge* and *purely probabilistic knowledge* from probabilistic logic, as well as *default logical knowledge* from default reasoning from conditional knowledge bases. For example, consider the *strict logical knowledge* "all penguins are birds", the *default logical knowledge* "generally, birds have legs" and "generally, birds fly", and the *purely probabilistic knowledge* "penguins fly with a probability of at most 0.05". Clearly, some desired conclusions are "generally, birds have legs", "generally, birds fly", and "penguins fly with a probability of at most 0.05", since these sentences are explicitly stated above. Two other desired conclusions are "generally, penguins have legs" (since the property of having legs of birds should be inherited to the subclass of all penguins) and "generally, red birds fly" (since being red is not mentioned at all and so should be irrelevant to the ability to fly). In weak nonmonotonic probabilistic logics, we can deal with all the above sentences. In particular, *probabilistic lexicographic entailment* [43] also produces all the above desired conclusions.

- A companion paper [41] introduces *strong nonmonotonic probabilistic logics*, which are similar probabilistic generalizations of default reasoning from conditional knowledge bases. They are, however, quite different from the ones in [43] in that they allow for handling *default purely probabilistic knowledge*, rather than *(strict) purely probabilistic knowledge*, in addition to strict logical knowledge and default logical knowledge. For example, they allow for expressing sentences "*generally*, birds fly with a probability of at least 0.95" rather than "birds fly with a probability of at least 0.95". Intuitively, the former means that being able to fly with a probability of at least 0.95 should apply to all birds and all subclasses of birds, as long as this is consistent, while the latter says that being able to fly with a probability of at least 0.95 should only apply to all birds. This is why the formalisms in [41] are generally much stronger than the ones in [43].

- Finally, [42] defines *nonmonotonic probabilistic logics under variable-strength inheritance with overriding*, which are a general approach to nonmonotonic probabilistic reasoning, which subsumes the approaches in [43] and [41] as special cases. Roughly, these formalisms also allow for handling *strict logical knowledge*, *default logical knowledge*, and *default purely probabilistic knowledge*, but the inheritance of purely probabilistic knowledge is controlled by a strength $\lambda \in [0,1]$. For $\lambda = 0$ (resp., $\lambda = 1$), these formalisms coincide with the weak (resp., strong) nonmonotonic ones in [43] (resp., [41]). For example, suppose that we have the default probabilistic knowledge "generally, yellow objects are easy to see with a probability between 0.8 and 0.9". In nonmonotonic probabilistic reasoning of strength 0 (resp., 0.5 and 1), we then conclude "generally, yellow birds are easy to see with a probability in $[0,1]$ (resp., $[0.6,1]$ and $[0.8,0.9]$)".

To date, however, there have been no works on the computational aspects of nonmonotonic probabilistic logics under variable-strength inheritance with overriding. In particular, there have been no implementations, neither of these unifying formalisms, nor of the special cases of weak and strong nonmonotonic probabilistic logics. In this paper, I try to fill this gap. The main contributions are as follows:

- I recall the nonmonotonic probabilistic logics under variable-strength inheritance with overriding presented in [42], namely, *probabilistic entailment in System Z of strength λ* (or z_λ*-entailment*) and *probabilistic lexicographic entailment of strength λ* (or lex_λ*-entailment*). I also provide several new examples.

- I give a precise picture of the complexity of deciding consistency of strength λ and of computing tight entailed intervals under z_λ- and lex_λ-entailment. Furthermore, I present an algorithm for deciding consistency of strength λ, which is based on a reduction to deciding satisfiability in model-theoretic probabilistic logic. I also present algorithms for computing tight entailed intervals under z_λ- and lex_λ-entailment, based on reductions to deciding satisfiability and computing tight logically entailed intervals in model-theoretic probabilistic logic.

- I describe the system NMPROBLOG, which includes a prototype implementation of the above algorithms. Deciding satisfiability (resp., computing tight logically entailed intervals) in model-theoretic probabilistic logic are reduced to deciding the solvability of a system of linear constraints (resp., solving linear programs), which is done by the linear programming solver "lp_solve 5.1" [9].

The rest of this paper is organized as follows. Section 2 gives some technical preliminaries. In Section 3, we recall the notions of z_λ-and lex_λ-entailment, and their semantic properties. Section 4 provides further examples to illustrate the notions of z_λ- and lex_λ-entailment. Section 5 describes the complexity results and the algorithms for deciding consistency of strength λ and computing tight entailed intervals under z_λ- and lex_λ-entailment. In Section 6, we present the system NMPROBLOG. Section 7 summarizes the main results and gives an outlook on future research.

2 Preliminaries

In this section, we recall probabilistic knowledge bases and the main concepts from model-theoretic probabilistic logic. Furthermore, we define the monotonic notion of logical entailment of strength $\lambda \in [0, 1]$.

2.1 Probabilistic Knowledge Bases

We now recall probabilistic knowledge bases. We start by defining logical constraints and probabilistic formulas, which are interpreted by probability distributions over a set of possible worlds.

We assume a set of *basic events* $\Phi = \{p_1, \ldots, p_n\}$ with $n \geqslant 1$. We use $\bot$ and $\top$ to denote *false* and *true*, respectively. We define *events* by induction as follows. Every element of $\Phi \cup \{\bot, \top\}$ is an event. If ϕ and ψ are events, then also $\neg\phi$ and $(\phi \wedge \psi)$. A *conditional event* is an expression of the form $\psi|\phi$, where ψ and ϕ are events. A *conditional constraint* has the form $(\psi|\phi)[l, u]$, where ψ and ϕ are events, and $l, u \in [0, 1]$ are reals. We define *probabilistic formulas* by induction as follows. Every conditional constraint is a probabilistic formula. If F and G are probabilistic formulas, then also $\neg F$ and $(F \wedge G)$. We use $(F \vee G)$ and $(F \Leftarrow G)$ to abbreviate $\neg(\neg F \wedge \neg G)$ and $\neg(\neg F \wedge G)$, respectively, where F and G are either two events or two probabilistic formulas, and adopt the usual conventions to eliminate parentheses. A *logical constraint* is an event of the form $\psi \Leftarrow \phi$.

A *world* I is a truth assignment to the basic events in Φ (that is, a mapping $I\colon \Phi \rightarrow \{\mathbf{true}, \mathbf{false}\}$), which is inductively extended to all events as usual (that is, $I(\bot) = \mathbf{false}$, $I(\top) = \mathbf{true}$, $I(\neg\phi) = \mathbf{true}$ iff $I(\phi) = \mathbf{false}$, and $I(\phi \wedge \psi) = \mathbf{true}$ iff $I(\phi) = I(\psi) = \mathbf{true}$). We denote by $\mathcal{I}_\Phi$ the set of all worlds for Φ. A world I *satisfies* an event ϕ, or I is a *model* of ϕ, denoted $I \models \phi$, iff $I(\phi) = \mathbf{true}$. A *probabilistic interpretation* Pr is a probability function on $\mathcal{I}_\Phi$ (that is, a mapping $Pr\colon \mathcal{I}_\Phi \rightarrow [0,1]$ such that all $Pr(I)$ with $I \in \mathcal{I}_\Phi$ sum up to 1). The *probability* of an event ϕ in Pr, denoted $Pr(\phi)$, is the sum of all $Pr(I)$ such that $I \in \mathcal{I}_\Phi$ and $I \models \phi$. For events ϕ and ψ with $Pr(\phi) > 0$, we write $Pr(\psi|\phi)$ to abbreviate $Pr(\psi \wedge \phi) / Pr(\phi)$. The *truth* of logical constraints and probabilistic formulas F in Pr, denoted $Pr \models F$, is inductively defined by: (i) $Pr \models \psi \Leftarrow \phi$ iff $Pr(\psi \wedge \phi) = Pr(\phi)$, (ii) $Pr \models (\psi|\phi)[l,u]$ iff $Pr(\phi) = 0$ or $Pr(\psi|\phi) \in [l,u]$, (iii) $Pr \models \neg F$ iff not $Pr \models F$, and (iv) $Pr \models (F \wedge G)$ iff $Pr \models F$ and $Pr \models G$. We say Pr *satisfies* F, or Pr is a *model* of F, iff $Pr \models F$. It *satisfies* a set of logical constraints and probabilistic formulas $\mathcal{F}$, or Pr is a *model* of $\mathcal{F}$, denoted $Pr \models \mathcal{F}$, iff Pr is a model of all $F \in \mathcal{F}$. We say $\mathcal{F}$ is *satisfiable* iff a model of $\mathcal{F}$ exists. A conditional constraint $C = (\psi|\phi)[l,u]$ is a *logical consequence* of $\mathcal{F}$, denoted $\mathcal{F} \Vdash C$, iff each model of $\mathcal{F}$ is also a model of C. It is a *tight logical consequence* of $\mathcal{F}$, denoted $\mathcal{F} \Vdash_{tight} C$, iff $l = \inf Pr(\psi|\phi)$ (resp., $u = \sup Pr(\psi|\phi)$) subject to all models Pr of $\mathcal{F}$ with $Pr(\phi) > 0$. Here, we define $l = 1$ and $u = 0$, when no such model Pr exists.

A *probabilistic knowledge base* $KB = (L, P)$ consists of a finite set of logical constraints L and a finite set of conditional constraints P. We say KB is *satisfiable* iff $L \cup P$ is satisfiable. A conditional constraint C is a *logical consequence* of KB, denoted $KB \Vdash C$, iff $L \cup P \Vdash C$. It is a *tight logical consequence* of KB, denoted $KB \Vdash_{tight} C$, iff $L \cup P \Vdash_{tight} C$. The following example illustrates the syntactic notion of a probabilistic knowledge base.

Example 2.1 The strict logical knowledge "all penguins are birds", the default logical knowledge "generally, birds have legs", and the default purely probabilistic knowledge "generally, yellow objects are easy to see with a probability between 0.8 and 0.9", "generally, birds fly with a probability of at least 0.9", and "generally, penguins fly with a probability of at most 0.1" can be expressed by the probabilistic knowledge base $KB = (L, P)$, where $L = \{\mathit{bird} \Leftarrow \mathit{penguin}\}$ and $P = \{(\mathit{legs}|\mathit{bird})[1,1], (\mathit{see}|\mathit{yellow})[.8,.9], (\mathit{fly}|\mathit{bird})[.9,1], (\mathit{fly}|\mathit{penguin})[0,.1]\}$.

2.2 Logical Entailment of Strength λ

As a first step towards z_λ- and lex_λ-entailment in Section 3, we now define the monotonic notion of *logical entailment of strength* $\lambda \in [0,1]$. It already realizes an inheritance of default purely probabilistic knowledge controlled by λ. Intuitively, the larger the strength λ, the stronger is the inheritance of default purely probabilistic knowledge, and thus the stronger is the notion of logical entailment of strength λ (see Example 2.2). In the extreme case $\lambda = 0$ (resp., $\lambda = 1$), default purely probabilistic knowledge is not inherited at all (resp., fully inherited). But, in contrast to z_λ- and lex_λ-entailment, logical entailment of strength λ has no overriding mechanism, and this often produces *local inconsistencies* (see Example 2.2).

In the sequel, we use $\phi \succcurlyeq \lambda$ to abbreviate the probabilistic formula $\neg(\phi|\top)[0,0] \wedge (\phi|\top)[\lambda,1]$. Informally, for any probabilistic interpretation Pr that satisfies $\phi \succcurlyeq \lambda$, it holds that $Pr(\phi) > 0$, if $\lambda = 0$, and $Pr(\phi) \geqslant \lambda$, otherwise. We define the notion of *logical entailment of strength* $\lambda \in [0,1]$ (or simply λ-*logical entailment*) as follows. A conditional constraint $C = (\psi|\phi)[l,u]$ is a λ-*logical consequence* of $KB = (L, P)$, denoted $KB \Vdash^\lambda C$, iff $L \cup P \cup \{\phi \succcurlyeq \lambda\} \Vdash C$. It is a *tight* λ-*logical consequence* of KB, denoted $KB \Vdash^\lambda_{tight} C$, iff $L \cup P \cup \{\phi \succcurlyeq \lambda\} \Vdash_{tight} C$.

As shown by the following example, the notion of λ-logical entailment already realizes an inheritance of default purely probabilistic knowledge controlled by λ. Intuitively, the larger the strength λ, the stronger are the tight λ-logical consequences $(\psi|\phi)[l,u]$ of KB influenced by $(\psi'|\phi')[l',u'] \in P$ with $L \models \phi' \Leftarrow \phi$.

Example 2.2 Let KB be as in Example 2.1. Some tight logical consequences of strength λ among 0, 0.2, 0.4, 0.6, 0.8, and 1 are shown in Table 1 (less desired intervals are bold). We observe an inheritance of default logical knowledge along subclass relationships, which is independent from λ. E.g., the default logical property of having legs is inherited from birds down to yellow birds. Furthermore, we observe an inheritance of default purely probabilistic knowledge along subclass relationships, which depends on the strength λ. E.g., being easy to see with a probability in $[.8,.9]$ is inherited from yellow objects down to yellow birds, but the new intervals are $[0,1]$, $[0,1]$, $[.5,1]$, $[.67,1]$, $[.75,1]$, and $[.8,.9]$, respectively. Finally, for $\lambda > 1/9$, there are local inconsistencies related to penguins (as being able to fly with a probability of at least 0.9 is inherited from birds down to penguins, and there it is inconsistent with being able to fly with a probability of at most 0.1).

Table 1: Some tight λ-logical consequences.

	$\lambda=0$	$\lambda=0.2$	$\lambda=0.4$	$\lambda=0.6$	$\lambda=0.8$	$\lambda=1$
$legs \mid bird$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$
$legs \mid yellow \wedge bird$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$
$legs \mid penguin$	$[1,1]$	$[1,0]$	$[1,0]$	$[1,0]$	$[1,0]$	$[1,0]$
$legs \mid yellow \wedge penguin$	$[1,1]$	$[1,0]$	$[1,0]$	$[1,0]$	$[1,0]$	$[1,0]$
$fly \mid bird$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$
$fly \mid yellow \wedge bird$	$[0,1]$	$[.5,1]$	$[.75,1]$	$[.83,1]$	$[.88,1]$	$[.9,1]$
$fly \mid penguin$	$[0,.1]$	$[1,0]$	$[1,0]$	$[1,0]$	$[1,0]$	$[1,0]$
$fly \mid yellow \wedge penguin$	$[0,1]$	$[1,0]$	$[1,0]$	$[1,0]$	$[1,0]$	$[1,0]$
$see \mid yellow$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$
$see \mid yellow \wedge bird$	$[0,1]$	$[0,1]$	$[.5,1]$	$[.67,1]$	$[.75,1]$	$[.8,.9]$
$see \mid yellow \wedge penguin$	$[0,1]$	$[1,0]$	$[1,0]$	$[1,0]$	$[1,0]$	$[1,0]$

3 Nonmonotonic Probabilistic Logics

In this section, we recall the notions of z_λ- and lex_λ-entailment from [42]. They are parameterized through a value $\lambda \in [0,1]$ that describes the *strength* of the inheritance of default purely probabilistic knowledge.

3.1 Consistency of Strength λ

We first describe the notion of consistency of strength λ (or simply λ-consistency), where $\lambda \in [0,1]$, for probabilistic knowledge bases $KB = (L,P)$.

A probabilistic interpretation Pr λ-*verifies* (resp., λ-*falsifies*) a conditional constraint $(\psi|\phi)[l,u]$ iff Pr satisfies $(\psi|\phi)[l,u]$ (resp., $\neg(\psi|\phi)[l,u]$) and $\phi \succcurlyeq \lambda$. Recall that Pr satisfies $(\psi|\phi)[l,u]$ iff $Pr(\phi)=0$ or $Pr(\psi|\phi) \in [l,u]$, and thus Pr λ-verifies (resp., λ-falsifies) $(\psi|\phi)[l,u]$ iff $Pr \models \phi \succcurlyeq \lambda$ and $Pr(\psi|\phi) \in [l,u]$ (resp., $Pr(\psi|\phi) \notin [l,u]$). A set of conditional constraints P λ-*tolerates* a conditional constraint C under a set of logical constraints L iff $L \cup P$ has a model that λ-verifies C. We say P is under L *in* λ-*conflict* with C

iff no model of $L \cup P$ λ-verifies C. A *conditional constraint ranking* σ on $KB = (L, P)$ maps each element of P to a nonnegative integer. If $P \neq \emptyset$, then we say that σ is λ-*admissible* with KB iff every $P' \subseteq P$ that is under L in λ-conflict with some $C \in P$ contains some C' such that $\sigma(C') < \sigma(C)$; if $P = \emptyset$, then σ is λ-*admissible* with KB iff L is satisfiable.

We define the notion of λ-consistency as follows. We say that KB is λ-*consistent* iff there exists a conditional constraint ranking σ on KB that is λ-admissible with KB. Otherwise, KB is λ-*inconsistent*. The following theorem characterizes the λ-consistency of KB through the existence of an ordered partition of P.

Theorem 3.1 *A probabilistic knowledge base $KB = (L, P)$ is λ-consistent iff (i) L is satisfiable and (ii) an ordered partition $(P_0, \ldots, P_k)$ of P exists such that each P_i, $0 \leqslant i \leqslant k$, is the set of all $C \in \bigcup_{j=i}^{k} P_j$ that are λ-tolerated under L by $\bigcup_{j=i}^{k} P_j$.*

The unique ordered partition $(P_0, \ldots, P_k)$ of P in Theorem 3.1 is called the z_λ-*partition* of $KB = (L, P)$. Hence, KB is λ-consistent iff (i) L is satisfiable and (ii) the z_λ-partition of KB exists. The following example shows some z_λ-partitions and a case of a λ-inconsistent probabilistic knowledge base.

Example 3.2 Let $KB = (L, P)$ be as in Example 2.1. For every $\lambda \in [0, 1/9]$, the z_λ-partition of KB is given by $(P_0) = (P)$. For every $\lambda \in (1/9, 1]$, the z_λ-partition of KB is given by $(P_0, P_1) = (P \setminus \{(\mathit{fly}|\mathit{penguin})[0, .1]\}, \{(\mathit{fly}|\mathit{penguin})[0, .1]\})$. Thus, KB is λ-consistent, for all $\lambda \in [0, 1]$, while the probabilistic knowledge base $KB = (L, P) = (\emptyset, \{(a|b)[0, .3], (a|b)[.6, 1]\})$ is λ-inconsistent, for all $\lambda \in [0, 1]$.

Note that the existence of a conditional constraint ranking σ on a probabilistic knowledge base KB that is λ-admissible with KB is also equivalent to the existence of a probability ranking that is λ-admissible with KB. Here, probability rankings and the λ-admissibility of probability rankings with probabilistic knowledge bases are defined as follows. A *probability ranking* κ maps each probabilistic interpretation on $\mathcal{I}_\Phi$ to a member of $\{0, 1, \ldots\} \cup \{\infty\}$ such that $\kappa(Pr) = 0$ for at least one interpretation Pr. It is extended to all logical constraints and probabilistic formulas F as follows. If F is satisfiable, then $\kappa(F) = \min\{\kappa(Pr) \mid Pr \models F\}$; otherwise, $\kappa(F) = \infty$. A probability ranking κ is λ-*admissible* with a probabilistic knowledge base $KB = (L, P)$ iff $\kappa(\neg F) = \infty$ for all $F \in L$ and $\kappa(\phi \succcurlyeq \lambda) < \infty$ and $\kappa(\phi \succcurlyeq \lambda \wedge (\psi|\phi)[l, u]) < \kappa(\phi \succcurlyeq \lambda \wedge \neg(\psi|\phi)[l, u])$ for all $(\psi|\phi)[l, u] \in P$.

3.2 System Z of Strength λ

We next define the notion of z_λ-entailment, where $\lambda \in [0, 1]$, for λ-consistent probabilistic knowledge bases $KB = (L, P)$. It is linked to a conditional constraint ranking z_λ on KB and a probability ranking κ^{z_λ}. Let $(P_0, \ldots, P_k)$ be the z_λ-partition of KB. For every $j \in \{0, \ldots, k\}$, each $C \in P_j$ is assigned the value j under z_λ. Then, κ^{z_λ} on all probabilistic interpretations Pr is defined as follows:

$$\kappa^{z_\lambda}(Pr) \;=\; \begin{cases} \infty & \text{if } Pr \not\models L \\ 0 & \text{if } Pr \models L \cup P \\ 1 + \max\limits_{C \in P\colon\, Pr \not\models C} z_\lambda(C) & \text{otherwise.} \end{cases}$$

Note that the rankings z_λ and κ^{z_λ} are both λ-admissible with KB. The ranking κ^{z_λ} defines a preference relation on probabilistic interpretations: For probabilistic interpretations Pr and Pr', we say Pr is z_λ-*preferable* to Pr' iff $\kappa^{z_\lambda}(Pr) < \kappa^{z_\lambda}(Pr')$. A model Pr of a set of logical constraints and probabilistic formulas $\mathcal{F}$ is a z_λ-*minimal model* of $\mathcal{F}$ iff no model of $\mathcal{F}$ is z_λ-preferable to Pr.

Table 2: Some tight z_λ-consequences.

	$\lambda=0$	$\lambda=0.2$	$\lambda=0.4$	$\lambda=0.6$	$\lambda=0.8$	$\lambda=1$
$legs \mid bird$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$
$legs \mid yellow \wedge bird$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$
$legs \mid penguin$	$[1,1]$	$[0,1]$	$[0,1]$	$[0,1]$	$[0,1]$	$[0,1]$
$legs \mid yellow \wedge penguin$	$[1,1]$	$[0,1]$	$[0,1]$	$[0,1]$	$[0,1]$	$[0,1]$
$fly \mid bird$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$
$fly \mid yellow \wedge bird$	$[0,1]$	$[.5,1]$	$[.75,1]$	$[.83,1]$	$[.88,1]$	$[.9,1]$
$fly \mid penguin$	$[0,.1]$	$[0,.1]$	$[0,.1]$	$[0,.1]$	$[0,.1]$	$[0,.1]$
$fly \mid yellow \wedge penguin$	$[0,1]$	$[0,.5]$	$[0,.25]$	$[0,.17]$	$[0,.13]$	$[0,.1]$
$see \mid yellow$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$
$see \mid yellow \wedge bird$	$[0,1]$	$[0,1]$	$[.5,1]$	$[.67,1]$	$[.75,1]$	$[.8,.9]$
$see \mid yellow \wedge penguin$	$[0,1]$	$[0,1]$	$[0,1]$	$[0,1]$	$[0,1]$	$[0,1]$

We are now ready to define the notion of z_λ-*entailment* as follows. A conditional constraint $C = (\psi|\phi)[l,u]$ is a z_λ-*consequence* of KB, denoted $KB \,|\!\!\sim^{z_\lambda} C$, iff every z_λ-minimal model of $L \cup \{\phi \succcurlyeq \lambda\}$ satisfies C. It is a *tight* z_λ-*consequence* of KB, denoted $KB \,|\!\!\sim^{z_\lambda}_{tight} C$, iff $l = \inf Pr(\psi|\phi)$ (resp., $u = \sup Pr(\psi|\phi)$) subject to all z_λ-minimal models Pr of $L \cup \{\phi \succcurlyeq \lambda\}$.

The following example shows that the notion of z_λ-entailment realizes an inheritance of default logical and default purely probabilistic properties from classes to non-exceptional subclasses, where the inheritance of default purely probabilistic properties depends on the strength λ. But z_λ-entailment does not inherit properties from classes to subclasses that are exceptional relative to some other property (and thus, like its classical counterpart, shows the problem of *inheritance blocking*).

Example 3.3 Consider again the probabilistic knowledge base $KB = (L, P)$ given in Example 2.1. Some tight z_λ-consequences, where $\lambda \in \{0, 0.2, 0.4, 0.6, 0.8, 1\}$, are shown in Table 2. Observe that, in contrast to Table 1, there are no empty intervals "$[1,0]$", that is, no local inconsistencies. Then, observe that the default logical property of having legs is inherited from the class of birds down to yellow birds, independently from λ, while the default purely probabilistic property of being easy to see with a probability between 0.8 and 0.9 is also inherited from the class of yellow objects to yellow birds, but this is controlled by λ. Furthermore, for every strength $\lambda > 1/9$, these properties are not inherited down to the exceptional classes of penguins and yellow penguins, respectively. Note that for every strength $\lambda \leqslant 1/9$, the default logical property of having legs is inherited down to penguins, since there is only some weak inheritance of default purely probabilistic knowledge, and thus no conflict between the abilities to fly of birds and penguins.

3.3 Lexicographic Entailment of Strength λ

We now describe the notion of lex_λ-entailment, where $\lambda \in [0,1]$, for λ-consistent probabilistic knowledge bases $KB = (L, P)$.

We use the z_λ-partition $(P_0, \ldots, P_k)$ of KB to define a lexicographic preference relation on probabilistic interpretations: For probabilistic interpretations Pr and Pr', we say Pr is lex_λ-*preferable* to Pr' iff some $i \in \{0, \ldots, k\}$ exists such that $|\{C \in P_i \mid Pr \models C\}| > |\{C \in P_i \mid Pr' \models C\}|$ and $|\{C \in P_j \mid Pr \models C\}| =$

Table 3: Some tight lex_λ-consequences.

	$\lambda=0$	$\lambda=0.2$	$\lambda=0.4$	$\lambda=0.6$	$\lambda=0.8$	$\lambda=1$
$legs\|bird$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$
$legs\|yellow\wedge bird$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$
$legs\|penguin$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$
$legs\|yellow\wedge penguin$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$
$fly\|bird$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$
$fly\|yellow\wedge bird$	$[0,1]$	$[.5,1]$	$[.75,1]$	$[.83,1]$	$[.88,1]$	$[.9,1]$
$fly\|penguin$	$[0,.1]$	$[0,.1]$	$[0,.1]$	$[0,.1]$	$[0,.1]$	$[0,.1]$
$fly\|yellow\wedge penguin$	$[0,1]$	$[0,.5]$	$[0,.25]$	$[0,.17]$	$[0,.13]$	$[0,.1]$
$see\|yellow$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$
$see\|yellow\wedge bird$	$[0,1]$	$[0,1]$	$[.5,1]$	$[.67,1]$	$[.75,1]$	$[.8,.9]$
$see\|yellow\wedge penguin$	$[0,1]$	$[0,1]$	$[.5,1]$	$[.67,1]$	$[.75,1]$	$[.8,.9]$

$|\{C\in P_j \mid Pr'\models C\}|$ for all $i<j\leqslant k$. A model Pr of a set of logical constraints and probabilistic formulas $\mathcal{F}$ is a lex_λ-*minimal model* of $\mathcal{F}$ iff no model of $\mathcal{F}$ is lex_λ-preferable to Pr.

We define the notion of lex_λ-*entailment* as follows. A conditional constraint $C=(\psi|\phi)[l,u]$ is a lex_λ-*consequence* of KB, denoted $KB \Vdash\!\sim^{lex_\lambda} C$, iff every lex_λ-minimal model of $L\cup\{\phi\succcurlyeq\lambda\}$ satisfies C. It is a *tight* lex_λ-*consequence* of KB, denoted $KB \Vdash\!\sim^{lex_\lambda}_{tight} C$, iff $l=\inf Pr(\psi|\phi)$ (resp., $u=\sup Pr(\psi|\phi)$) subject to all lex_λ-minimal models Pr of $L\cup\{\phi\succcurlyeq\lambda\}$. Note that the notion of lex_λ-entailment can also be defined in terms of a unique probability ranking for KB.

The following example shows that the notion of lex_λ-entailment realizes an inheritance of default properties, without showing the problem of inheritance blocking.

Example 3.4 Consider again the probabilistic knowledge base $KB=(L,P)$ given in Example 2.1. Some tight lex_λ-consequences, where $\lambda\in\{0,0.2,0.4,0.6,0.8,1\}$, are shown in Table 3. In particular, for every strength $\lambda\in[0,1]$, the default logical property of having legs is inherited from the class of birds to the exceptional subclass of penguins, while the default purely probabilistic property of being easy to see with a probability between 0.8 and 0.9 is also inherited from the class of yellow objects to the exceptional subclass of yellow penguins.

3.4 Semantic Properties

We finally briefly summarize some semantic properties of λ-logical, z_λ-, and lex_λ-entailment. More precisely, we describe the general nonmonotonic properties of the formalisms, the relationships between the formalisms, and their probabilistic and classical special cases.

As for general nonmonotonic properties, λ-logical, z_λ-, and lex_λ-entailment all satisfy probabilistic versions of the postulates *Right Weakening*, *Reflexivity*, *Left Logical Equivalence*, *Cut*, *Cautious Monotonicity*, and *Or* proposed by Kraus et al. [32], which are commonly regarded as being particularly desirable for any reasonable notion of nonmonotonic entailment [42]. All three notions also satisfy the property of *Rational Monotonicity* [32], which describes a restricted form of monotony and allows to ignore certain kinds of irrelevant knowledge.

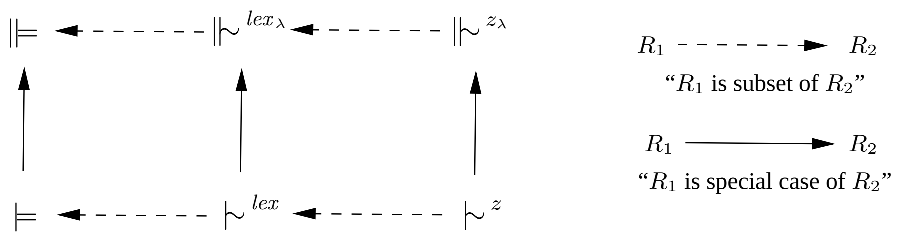

Figure 1: Relationships between probabilistic and ordinary formalisms.

Concerning the relationships between the three formalisms, λ-logical entailment is stronger than both lex_λ- and z_λ-entailment. Moreover, lex_λ-entailment is stronger than z_λ-entailment. These relationships between λ-logical, z_λ-, and lex_λ-entailment are illustrated in Fig. 1, upper part. In general, λ-logical entailment is strictly stronger than lex_λ-entailment, which in turn is strictly stronger than z_λ-entailment. However, in the special case when $\phi = \top$, the three notions of λ-logical, z_λ-, and lex_λ-entailment of $(\psi|\phi)[l,u]$ from λ-consistent $KB = (L,P)$ all coincide. Furthermore, also when $L \cup P \cup \{\phi \succcurlyeq \lambda\}$ is satisfiable, the three notions of λ-logical, z_λ-, and lex_λ-entailment of $(\psi|\phi)[l,u]$ from λ-consistent $KB = (L,P)$ all coincide.

Some probabilistic special cases are summarized as follows. For $\lambda = 0$, the notion of λ-logical entailment from KB coincides with standard logical entailment from KB. For $\lambda = 0$ (resp., $\lambda = 1$), the notions of z_λ- and lex_λ-entailment coincide with weak (resp., strong) probabilistic z- and lex-entailment introduced in [43] (resp., [41]). Furthermore, for $\lambda = 0$, the notion of λ-consistency coincides with the notion of g-coherence (see, e.g., [12]). As for classical special cases, z_λ- and lex_λ-entailment of $(\beta|\alpha)[1,1]$ from λ-consistent probabilistic knowledge bases of the form $KB = (L,P)$, where $P = \{(\psi_i|\phi_i)[1,1] \mid i \in \{1,\ldots,n\}\}$, coincide with the classical notions of Pearl's entailment in System Z and Lehmann's lexicographic entailment of the default $\beta \leftarrow \alpha$ from the default counterpart of KB. Moreover, λ-logical entailment of $(\beta|\alpha)[1,1]$ from such KB coincides with propositional logical entailment of $\beta \Leftarrow \alpha$ from the propositional counterpart of KB (see Fig. 1). Furthermore, the notion of λ-consistency for such KB coincides with the notion of ε- (or also p-) consistency for the default counterpart of KB. Finally, notice also that for such KB, the notion of z_λ-partition of KB does not depend on λ.

4 Further Examples

In this section, we provide some other examples. The first one deals with reasoning from statistical knowledge and degrees of belief, where z_1- and lex_1-entailment show a similar behavior as reference-class reasoning [33, 48] in a number of uncontroversial examples, but also avoid many drawbacks of reference-class reasoning. More precisely, they can handle complex scenarios and even purely probabilistic subjective knowledge as input. Furthermore, conclusions are drawn in a global way from all the available knowledge as a whole. See [41] for further details.

Example 4.1 Suppose the statistical knowledge "all penguins are birds", "between 90% and 95% of all birds fly", "at most 5% of all penguins fly", and "at least 95% of all yellow objects are easy to see". Moreover, assume we believe "Sam is a yellow penguin". What do we then conclude about Sam's property of being easy to see? Under reference-class reasoning, which is a machinery for dealing with statistical knowledge and degrees of belief, we conclude "Sam is easy to see with a probability of at least 0.95". This is exactly what we obtain using lex_1-entailment. The above statistical knowledge can be represented by the probabilistic knowledge base $KB=(L,P)$, where $L=\{bird \Leftarrow penguin\}$ and $P=\{(fly|bird)[.9,.95], (fly|penguin)[0,.05], (see|yellow)[.95,1]\}$. It is then not difficult to verify that KB is 1-consistent, and that $(see|yellow \wedge penguin)[0.95,1]$ is a tight conclusion from KB under lex_1-entailment. Some other tight intervals for $see|yellow \wedge penguin$ from KB under λ-logical, z_λ-, and lex_λ-entailment are shown in Table 4.

Table 4: Tight intervals for $see|yellow \wedge penguin$.

	$\lambda=0$	$\lambda=0.2$	$\lambda=0.4$	$\lambda=0.6$	$\lambda=0.8$	$\lambda=1$
tight λ-logical entailment	$[0,1]$	$[1,0]$	$[1,0]$	$[1,0]$	$[1,0]$	$[1,0]$
tight lex_λ-entailment	$[0,1]$	$[.75,1]$	$[.88,1]$	$[.92,1]$	$[.94,1]$	$[.95,1]$
tight z_λ-entailment	$[0,1]$	$[0,1]$	$[0,1]$	$[0,1]$	$[0,1]$	$[0,1]$

The next example is from the area of medical diagnosis [31].

Example 4.2 In a hospital, physicians have to diagnose whether patients with acute abdominal pain are suffering from appendicitis or not. Diagnosing appendicitis is a difficult task, since a lot of different symptoms (such as, for example, high temperature, a high rate of leucocytes, vomiting, and various types of pains) can indicate appendicitis, but often only the joint occurrence of several of these symptoms reliably supports the diagnosis. Here, we only consider four possible symptoms of appendicitis (*app*), namely a high rate of leucocytes (*leuco_high*) and the following three different types of pain: rectal pain (*rec_pain*), pain when released (*pain_rel*), and rebound tenderness (*reb_tender*). Thus, our view on this area is a very simplified one. Let our knowledge about the relationships between *app*, *leuco_high*, and the three types of pain be expressed by the following probabilistic knowledge base $KB=(L,P)$, where $L=\emptyset$ and P is given as follows:

$$\begin{aligned} P \;=\; & \{(\mathit{reb\_tender} \,|\, \mathit{pain\_rel})[.70,.75], (\mathit{reb\_tender} \,|\, \mathit{leuco\_high})[.70,.75], \\ & (\mathit{app} \,|\, \mathit{rec\_pain} \wedge \mathit{pain\_rel})[.70,.75], (\mathit{app} \,|\, \mathit{rec\_pain} \wedge \mathit{reb\_tender})[.65,.70], \\ & (\mathit{app} \,|\, \mathit{pain\_rel} \wedge \mathit{reb\_tender} \wedge \mathit{leuco\_high})[.80,.85]\}\,. \end{aligned}$$

Suppose Judy is a patient showing the symptoms *leuco_high* and *pain_rel*. Which is the probability that Judy has appendicitis? Which is the probability that she has appendicitis given that she also feels rectal pain? Some tight intervals for *app*|*leuco_high* ∧ *pain_rel* and *app*|*leuco_high* ∧ *pain_rel* ∧ *rec_pain* from KB under λ-logical, z_λ-, and lex_λ-entailment are shown in Tables 5 and 6, respectively.

Table 5: Tight intervals for *app*|*leuco_high* ∧ *pain_rel*.

	$\lambda=0$	$\lambda=0.2$	$\lambda=0.4$	$\lambda=0.6$	$\lambda=0.8$	$\lambda=1$
tight λ-logical entailment	$[0,1]$	$[.08,.99]$	$[.38,.93]$	$[.48,.91]$	$[.53,.9]$	$[.56,.9]$
tight lex_λ-entailment	$[0,1]$	$[.08,.99]$	$[.38,.93]$	$[.48,.91]$	$[.53,.9]$	$[.56,.9]$
tight z_λ-entailment	$[0,1]$	$[.08,.99]$	$[.38,.93]$	$[.48,.91]$	$[.53,.9]$	$[.56,.9]$

Table 6: Tight intervals for *app*|*leuco_high*∧*pain_rel*∧*rec_pain*.

	$\lambda = 0$	$\lambda = 0.2$	$\lambda = 0.4$	$\lambda = 0.6$	$\lambda = 0.8$	$\lambda = 1$
tight λ-logical entailment	$[0, 1]$	$[0, 1]$	$[.41, 1]$	$[.57, 1]$	$[.66, .92]$	$[1, 0]$
tight lex_λ-entailment	$[0, 1]$	$[0, 1]$	$[.41, 1]$	$[.57, 1]$	$[.66, .92]$	$[.7, .75]$
tight z_λ-entailment	$[0, 1]$	$[0, 1]$	$[.41, 1]$	$[.57, 1]$	$[.66, .92]$	$[0, 1]$

5 Algorithms and Complexity

In this section, we provide algorithms for solving the main computational problems in nonmonotonic probabilistic logics under variable-strength inheritance with overriding, and we give a precise picture of the complexity of these problems.

5.1 Problem Statements

The main decision and optimization problems of nonmonotonic probabilistic logics under variable-strength inheritance with overriding are summarized as follows:

CONSISTENCY: Given a probabilistic knowledge base $KB = (L, P)$ and a strength $\lambda \in [0, 1]$, decide whether KB is λ-consistent.

TIGHT s-CONSEQUENCE: Given a λ-consistent probabilistic knowledge base $KB = (L, P)$, a conditional event $\beta|\alpha$, and a strength $\lambda \in [0, 1]$, compute $l, u \in [0, 1]$ such that $KB \Vdash\!\sim^{s_\lambda} (\beta|\alpha)[l, u]$, for some fixed semantics $s \in \{z, lex\}$.

For the complexity results below, we assume that the strength $\lambda \in [0, 1]$ and all numbers in probabilistic knowledge bases $KB = (L, P)$ are rational.

5.2 Algorithms

Algorithm consistency in Fig. 2 decides whether a given probabilistic knowledge base $KB = (L, P)$ is λ-consistent for a given strength $\lambda \in [0, 1]$. If KB is λ-consistent, then the algorithm also returns the z_λ-partition of KB. It is a variable-strength generalization of an algorithm for deciding g-coherence by Biazzo et al. [11], which in turn is a probabilistic generalization of an algorithm for deciding ε-consistency in default reasoning by Goldszmidt and Pearl [25]. The algorithm consistency works as follows. If $P = \emptyset$, then Step 1 returns the empty partition $()$, if L is satisfiable; and nil, otherwise. If $P \neq \emptyset$, then Steps 2-7 try to compute the z_λ-partition $(P_0, \ldots, P_k)$ of KB, and Step 8 returns $(P_0, \ldots, P_k)$, if this succeeds; and nil, otherwise.

Algorithms tight-s-consequence, where $s = \mathsf{z}$ and $s = \mathsf{lex}$, in Figs. 3 and 4 compute tight intervals for a given conditional event $\beta|\alpha$ under z_λ- and lex_λ-entailment, respectively, and a given strength $\lambda \in [0, 1]$ from a given λ-consistent probabilistic knowledge base $KB = (L, P)$. They are variable-strength generalizations and improvements of algorithms in [43] for computing tight entailed intervals under weak probabilistic z- and lex-entailment, and are also related to algorithms for inference in System Z [47] and lexicographic inference [8], respectively. Algorithm tight-s-consequence, where $s = \mathsf{z}$ (resp., $s = \mathsf{lex}$), works as follows. If $L \cup \{\alpha \succcurlyeq \lambda\}$ is unsatisfiable, then $[1, 0]$ is returned in Step 1. Otherwise, we use Theorem 5.1 below saying that then a set $\mathcal{D}^s_\alpha(KB) \subseteq 2^P$, $s \in \{z_\lambda, lex_\lambda\}$, exists such that $KB \Vdash\!\sim^s (\beta|\alpha)[l, u]$

Algorithm consistency

Input: probabilistic knowledge base $KB = (L, P)$ and strength $\lambda \in [0, 1]$.
Output: z_λ-partition of KB, if KB is λ-consistent; *nil*, otherwise.

1. **if** $P = \emptyset$ **then if** L is satisfiable **then return** $()$ **else return** *nil*;
2. $R := P;\ i := -1;$
3. **repeat**
4. $\quad i := i + 1;$
5. $\quad D[i] := \{(\psi|\phi)[l, u] \in R \mid L \cup R \cup \{\phi \succcurlyeq \lambda\}$ is satisfiable$\}$;
6. $\quad R := R \setminus D[i]$
7. **until** $R = \emptyset$ **or** $D[i] = \emptyset$;
8. **if** $R = \emptyset$ **then return** $(D[0], \ldots, D[i])$ **else return** *nil*.

Figure 2: Algorithm consistency.

iff $L \cup H \cup \{\alpha \succcurlyeq \lambda\} \Vert\!\!\models (\beta|\alpha)[l, u]$ for all $H \in \mathcal{D}^s_\alpha(KB)$. In this case, we compute $\mathcal{D}^s_\alpha(KB)$ along the z_λ-partition of KB by binary search in Steps 2–6 (resp., 2–14), and the requested tight interval in Step 7 (resp., Steps 15–19). In particular, tight-lex-consequence computes $\mathcal{D}^s_\alpha(KB)$ stepwise along the components $P_k, \ldots, P_0$ of the z_λ-partition $(P_0, \ldots, P_k)$ of KB: Keeping track of the already computed parts of the members of $\mathcal{D}^s_\alpha(KB)$, a binary search is done for every P_i.

For $G, H \subseteq P$, we say G is z_λ*-preferable* to H iff some $i \in \{0, \ldots, k\}$ exists such that $P_i \subseteq G$, $P_i \not\subseteq H$, and $P_j \subseteq G$ and $P_j \subseteq H$ for all $i < j \leqslant k$. We say G is lex_λ*-preferable* to H iff some $i \in \{0, \ldots, k\}$ exists such that $|G \cap P_i| > |H \cap P_i|$ and $|G \cap P_j| = |H \cap P_j|$ for all $i < j \leqslant k$. For $\mathcal{D} \subseteq 2^P$ and $s \in \{z_\lambda, lex_\lambda\}$, we say G is s*-minimal* in $\mathcal{D}$ iff $G \in \mathcal{D}$ and no $H \in \mathcal{D}$ is s-preferable to G.

Theorem 5.1 *Let $KB = (L, P)$ be λ-consistent, and let $\beta|\alpha$ be a conditional event such that $L \cup \{\alpha \succcurlyeq \lambda\}$ is satisfiable. Let $s \in \{z_\lambda, lex_\lambda\}$, and let $\mathcal{D}^s_\alpha(KB)$ be the set of all s-minimal elements in $\{H \subseteq P \mid L \cup H \cup \{\alpha \succcurlyeq \lambda\}$ is satisfiable$\}$. Then, l (resp., u) such that $KB \mathrel{|\!\!\sim}^{s}_{tight} (\beta|\alpha)[l, u]$ is given by $l = \min c$ (resp., $u = \max d$) subject to $L \cup H \cup \{\alpha \succcurlyeq \lambda\} \Vert\!\!\models_{tight} (\beta|\alpha)[c, d]$ and $H \in \mathcal{D}^s_\alpha(KB)$.*

Proof (sketch). The statement of the theorem follows from the observation that a probabilistic interpretation Pr is an s-minimal model of $L \cup \{\alpha \succcurlyeq \lambda\}$ iff (i) Pr is a model of $L \cup \{\alpha \succcurlyeq \lambda\}$ and (ii) $\{F \in P \mid Pr \models F\}$ is an s-minimal element in the set of all $H \subseteq P$ such that $L \cup H \cup \{\alpha \succcurlyeq \lambda\}$ is satisfiable. The latter is in turn equivalent to Pr being a model of $L \cup H \cup \{\alpha \succcurlyeq \lambda\}$ for some $H \in \mathcal{D}^s_\alpha(KB)$. $\Box$

Algorithms consistency, tight-z-consequence, and tight-lex-consequence are based on reductions to the following decision and optimization problems: (i) given a probabilistic knowledge base $KB = (L, P)$ and an event α, decide whether KB has a model Pr such that $Pr(\alpha) > 0$; and (ii) given $KB = (L, P)$ and a conditional event $\beta|\alpha$, compute the tight interval for $\beta|\alpha$ under logical entailment from KB. Some upper bounds for the number of tasks (i) and (ii) to be solved in Algorithms consistency, tight-z-consequence, and tight-lex-consequence are given by $O(|P|^2)$, $O(\ln(|P|))$, and $O(2^{|P|})$, respectively. The task (i) can be reduced to deciding whether a system of linear constraints is solvable, while (ii) can be reduced to computing the optimal values of two linear programs. These two well-known results (see especially [27, 20, 2]) are summarized in the following theorem.

Algorithm tight-z-consequence

Input: λ-consistent probabilistic knowledge base $KB = (L, P)$, conditional event $\beta|\alpha$, and strength $\lambda \in [0,1]$. The z_λ-partition of KB is denoted by $(P_0, \ldots, P_k)$, $k \geqslant -1$.

Output: interval $[l, u] \subseteq [0,1]$ such that $KB \,\|\!\sim^{z_\lambda}_{tight}\, (\beta|\alpha)[l,u]$.

if $L \cup \{\alpha \succcurlyeq \lambda\}$ is unsatisfiable **then return** $[1, 0]$;
if $L \cup P \cup \{\alpha \succcurlyeq \lambda\}$ is satisfiable **then** $(m, n) := (k, k)$ **else** $(m, n) := (-1, k-1)$;
while $m < n$ **do begin**
  $l := \lceil (m+n)\,/\,2 \rceil$;
  if $L \cup P_0 \cup \cdots \cup P_l \cup \{\alpha \succcurlyeq \lambda\}$ is satisfiable **then** $m := l$ **else** $n := l-1$
end;
compute $l, u \in [0,1]$ such that $L \cup P_0 \cup \cdots \cup P_m \cup \{\alpha \succcurlyeq \lambda\} \,\|\!\!=_{tight} (\beta|\alpha)[l,u]$;
return $[l, u]$.

Figure 3: Algorithm tight-z-consequence.

Algorithm tight-lex-consequence

Input: λ-consistent probabilistic knowledge base $KB = (L, P)$, conditional event $\beta|\alpha$, and strength $\lambda \in [0,1]$. The z_λ-partition of KB is denoted by $(P_0, \ldots, P_k)$, $k \geqslant -1$.

Output: interval $[l, u] \subseteq [0,1]$ such that $KB \,\|\!\sim^{lex_\lambda}_{tight}\, (\beta|\alpha)[l,u]$.

if $L \cup \{\alpha \succcurlyeq \lambda\}$ is unsatisfiable **then return** $[1, 0]$;
$\mathcal{H} := \{\emptyset\}$;
for $j := k$ **downto** 0 **do begin**
  $\mathcal{H}' := \{P_j \cup H \mid H \in \mathcal{H},\ L \cup P_j \cup H \cup \{\alpha \succcurlyeq \lambda\}$ is satisfiable$\}$;
  if $\mathcal{H}' \neq \emptyset$ **then** $\mathcal{H} := \mathcal{H}'$ **else begin**
    $(m, n) := (0, |P_j| - 1)$;
    while $m < n$ **do begin**
      $l := \lceil (m+n)\,/\,2 \rceil$;
      $\mathcal{H}' := \{G \cup H \mid G \subseteq P_j,\ |G| = l,\ H \in \mathcal{H},\ L \cup G \cup H \cup \{\alpha \succcurlyeq \lambda\}$ is satisfiable$\}$;
      if $\mathcal{H}' \neq \emptyset$ **then** $m := l$ **else** $n := l-1$
    end;
    $\mathcal{H} := \{G \cup H \mid G \subseteq P_j,\ |G| = m,\ H \in \mathcal{H},\ L \cup G \cup H \cup \{\alpha \succcurlyeq \lambda\}$ is satisfiable$\}$
  end
end;
$(l, u) := (1, 0)$;
for each $H \in \mathcal{H}$ **do begin**
  compute $c, d \in [0,1]$ such that $L \cup H \cup \{\alpha \succcurlyeq \lambda\} \,\|\!\!=_{tight} (\beta|\alpha)[c,d]$;
  $(l, u) := (\min(l, c), \max(u, d))$
end;
return $[l, u]$.

Figure 4: Algorithm tight-lex-consequence.

Theorem 5.2 *Let $KB = (L, P)$ be a probabilistic knowledge base, and let α, β be events. Let $R = \{I \in \mathcal{I}_\Phi \mid I \models L\}$. Let LC denote the system of linear constraints in Fig. 5 over the variables y_r $(r \in R)$. Then, (a) $L \cup P$ has a model Pr such that $Pr(\alpha) > 0$ iff LC is solvable. (b) If $L \cup P$ has a model Pr such that $Pr(\alpha) > 0$, then l (resp., u) such that $L \cup P \Vdash_{tight} (\beta|\alpha)[l, u]$ is given by the optimal value of the following linear program over the variables y_r $(r \in R)$:*

minimize (resp., maximize) $\sum_{r \in R,\, r \models \beta \wedge \alpha} y_r$ *subject to LC.*

$$
\begin{array}{rl}
\sum_{r \in R,\, r \models \neg\psi \wedge \phi} -l\, y_r + \sum_{r \in R,\, r \models \psi \wedge \phi} (1{-}l)\, y_r \geqslant 0 & \text{(for all } (\psi|\phi)[l,u] \in P,\ l > 0) \\
\sum_{r \in R,\, r \models \neg\psi \wedge \phi} u\, y_r + \sum_{r \in R,\, r \models \psi \wedge \phi} (u{-}1)\, y_r \geqslant 0 & \text{(for all } (\psi|\phi)[l,u] \in P,\ u < 1) \\
\sum_{r \in R,\, r \models \alpha} y_r = 1 & \\
y_r \geqslant 0 & \text{(for all } r \in R)
\end{array}
$$

Figure 5: System of linear constraints *LC*.

5.3 Complexity

We now analyze the complexity of the above decision and optimization problems. We first briefly recall the complexity classes that occur in our results. We assume some basic knowledge about the complexity classes P and NP; see especially [22, 30, 45] for further background. The class P^{NP} contains all decision problems that can be solved in deterministic polynomial time with an oracle for NP. The relationship between these complexity classes is described by the following inclusion hierarchy (note that all inclusions are currently believed to be strict):

$$\mathrm{P} \subseteq \mathrm{NP} \subseteq \mathrm{P}^{\mathrm{NP}}.$$

In order to classify problems that compute an output value, rather than a Yes / No-answer, function classes have been introduced. In particular, FP and $\mathrm{FP}^{\mathrm{NP}}$ are the functional analogs of P and P^{NP}, respectively.

The following result shows that CONSISTENCY is NP-complete.

Theorem 5.3 CONSISTENCY *is* NP-*complete.*

Proof (sketch). Hardness for NP follows from the fact that the special case of deciding whether KB is 0-consistent is NP-complete [43]. Membership in NP can be proved by showing that guessing and verifying a conditional constraint ranking σ on KB that is λ-admissible with KB can be done in nondeterministic polynomial time. The line of argumentation for this is similar to the proof of NP-membership of deciding whether KB is 0-consistent [43]. □

The next result shows that TIGHT z- and lex-CONSEQUENCE are $\mathrm{FP}^{\mathrm{NP}}$-complete.

Theorem 5.4 TIGHT s-CONSEQUENCE, $s \in \{z, lex\}$, *is* $\mathrm{FP}^{\mathrm{NP}}$-*complete.*

Proof (sketch). Hardness for $\mathrm{FP}^{\mathrm{NP}}$ follows from the fact that the special case of computing the tight interval for a given $\beta|\alpha$ under z_0- and lex_0-entailment from a given KB is $\mathrm{FP}^{\mathrm{NP}}$-complete [43]. As for Membership in $\mathrm{FP}^{\mathrm{NP}}$, computing the tight interval for $\beta|\alpha$ under z_λ- and lex_λ-entailment from $KB=(L,P)$ can be done in $\mathrm{FP}^{\mathrm{NP}}$ by a variant of Algorithm tigh-entailment-opt in [40]. Rather than checking the existence of some model Pr of $L\cup P$ with $Pr(\alpha)>0$, we check the existence of some $P'\in\mathcal{D}^s_\alpha(KB)$ (see Section 5.2) and some model Pr of $L\cup P'$ with $Pr\models\alpha\not\geq\lambda$. The proof of this is similar to the proof of $\mathrm{FP}^{\mathrm{NP}}$-membership of computing the tight interval for $\beta|\alpha$ under z_0- and lex_0-entailment from KB [43]. $\Box$

6 The System NMPROBLOG

The main components of the system NMPROBLOG are the main window, as well as one window each for (i) checking satisfiability, (ii) checking λ-consistency, (iii) computing the z_λ-partition (see Fig. 6), and (iv) computing tight entailed intervals for any conditional event under λ-logical, z_λ-, lex_λ-, and p_λ-entailment (see Fig. 7), for any probabilistic knowledge base $KB=(L,P)$ and any strength $\lambda\in\{i/100\,|\,i\in\{0,\ldots,100\}\}$. Here, p_λ-*entailment* is a probabilistic generalization of entailment in System P of strength $\lambda\in[0,1]$, which coincides with *g-coherent entailment* (see, e.g., [12]) for $\lambda=0$. The above restriction of the available strengths λ allows for a more comfortable use of NMPROBLOG via its graphical user interface (GUI). Note that the entailment relation and the strength λ that are actually used in a concrete application naturally depend on the desired entailment behavior. They may be chosen after some testing with NMPROBLOG. The system NMPROBLOG is written in C, and uses the linear programming solver "lp_solve 5.1" [9] for deciding the solvability of systems of linear constraints and for computing the optimal values of linear programs. Its GUI has been built using "glade 2.6".

NMPROBLOG loads from a file with suffix ".tax" a set of statements of one of the following forms: (i) $p=1$, where p is a nonempty string, which declares p as $\top$, (ii) $p=0$, where p is a nonempty string, which declares p as $\bot$, (iii) $p<1$, where p is a nonempty string, which declares p as a basic event, and (iv) $\psi>\phi$, where ψ and ϕ are events (in which "^", "&", and "#" encode $\neg$, $\wedge$, and $\vee$, respectively), which encodes that ϕ implies ψ. Furthermore, it then loads from a file with suffix ".prb" a set of statements of the form "$\psi\ \phi\ l\ u$", where ψ and ϕ are events as above, and l and u are real numbers. Such a statement encodes the conditional constraint $(\psi|\phi)[l,u]$. Note that every basic event in the ".prb"-file and in queries (window for computing tight entailed intervals; see Fig. 7) must be declared in the ".tax"-file.

Example 6.1 Consider again the probabilistic knowledge base $KB=(L,P)$ of Example 2.1. The ".tax"-file contains the statements $1>bird$, $1>penguin$, $1>fly$, $1>legs$, $1>see$, $1>yellow$, and $bird>penguin$, which declare the basic events in KB and encode the logical constraints in L. The ".prb"-file contains the statements $legs\ bird$ 1.0 1.0, $see\ yellow$ 0.8 0.9, $fly\ bird$ 0.9 1.0, and $fly\ penguin$ 0.0 0.1, which encode the conditional constraints in P. After reading the ".tax"- and the ".prb"-file, NMPROBLOG allows the user to open the window for computing tight consequences in Fig. 7 and to compute $[l,u]$ such that, e.g., $KB\,\|\!\!\sim^{lex_\lambda}_{tight}(see|yellow\wedge bird)[l,u]$, where $\lambda=0.5$, which is given by $[l,u]=[0.6,1]$ (see Fig. 7).

Example 6.2 Fig. 8 shows the time used by NMPROBLOG on a chain of n correlated basic events (which produces linear optimization problems that consist of 2^n variables and $4\,n-3$ constraints) for checking satisfiability and λ-consistency, as well as computing the z_λ-partition and tight entailed intervals under λ-logical, z_λ-, lex_λ-, and p_λ-entailment. Note especially that all the above reasoning tasks can be solved in few minutes, even when large linear optimization problems are generated (up to 16384 variables and 53

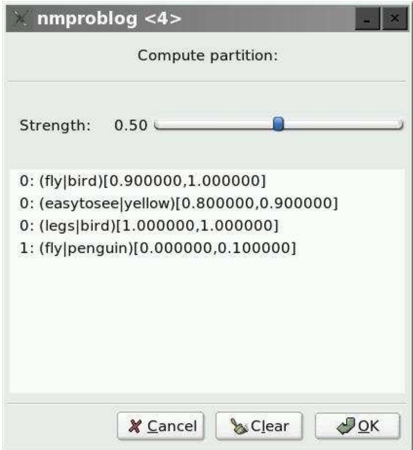

Figure 6: Window for computing the z_λ-partition.

Figure 7: Window for computing tight entailed intervals.

linear constraints). Note also that computing tight intervals under λ-logical, z_λ-, lex_λ-, and p_λ-entailment from KB always includes computing the z_λ-partition of KB as a first computation step.

7 Conclusion

I have recalled nonmonotonic probabilistic logics under variable-strength inheritance with overriding, namely, the notions of z_λ- and lex_λ-entailment, along with their semantic properties and some new examples. I have given a precise picture of the complexity of deciding λ-consistency and of computing tight entailed intervals under z_λ- and lex_λ-entailment. I have also provided algorithms for these tasks, which are based on reductions to the problems of deciding satisfiability and of computing tight logically entailed intervals in model-theoretic probabilistic logic. Hence, efficient linear optimization techniques for reasoning in model-theoretic probabilistic logic (such as e.g. the very powerful column generation techniques [29, 28]) can immediately be applied for reasoning in the presented nonmonotonic probabilistic logics under variable-strength inheritance with overriding.

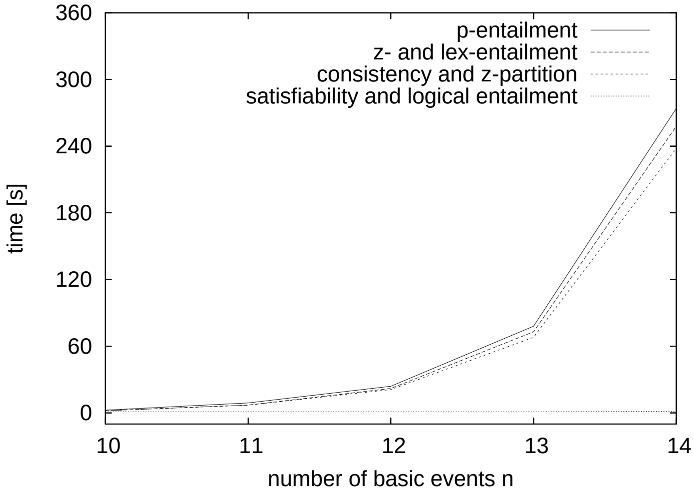

Figure 8: Time used by NMPROBLOG on a chain of n basic events (2^n variables).

I have then presented the system NMPROBLOG (available at `http://www.kr.tuwien.ac.at/staff/lukasiew/nmproblog.tar.gz`), which comprises a prototype implementation of the above algorithms. The system allows for (i) checking the satisfiability of probabilistic knowledge bases KB, (ii) checking the λ-consistency of KB, (iii) computing the z_λ-partition of KB, and (iv) computing tight entailed intervals from KB under any among λ-logical, lex_λ-, z_λ-, and p_λ-entailment, for any *strength* $\lambda \in \{i/100 \mid i \in \{0, \ldots, 100\}\}$. In particular, it thus also allows for probabilistic and default reasoning in all the special cases of λ-logical, lex_λ-, z_λ-, and p_λ-entailment (summarized in Sections 3.4 and 6).

An interesting topic of future work is to include an implementation of more efficient linear optimization techniques (such as e.g. the very powerful column generation techniques [29, 28]) into NMPROBLOG, and to explore whether there are other techniques for more efficient or even tractable inference in nonmonotonic probabilistic logics under variable-strength inheritance with overriding (e.g., using preprocessing steps along the lines of [11] and [16]) and to include them into NMPROBLOG. Another topic of future research is to explore whether similar forms of nonmonotonic probabilistic logics under variable-strength inheritance with overriding can be defined for other default reasoning formalisms (such as e.g. the approach in [6], which allows for dealing with explicit independence assumptions).

References

[1] E. W. Adams. *The Logic of Conditionals*, volume 86 of *Synthese Library*. D. Reidel, Dordrecht, Netherlands, 1975.

[2] S. Amarger, D. Dubois, and H. Prade. Constraint propagation with imprecise conditional probabilities. In *Proceedings UAI-1991*, pp. 26–34.

[3] F. Bacchus, A. Grove, J. Y. Halpern, and D. Koller. From statistical knowledge bases to degrees of belief. *Artif. Intell.*, 87(1–2):75–143, 1996.

[4] S. Benferhat, C. Cayrol, D. Dubois, J. Lang, and H. Prade. Inconsistency management and prioritized syntax-based entailment. In *Proceedings IJCAI-1993*, pp. 640–645.

[5] S. Benferhat, D. Dubois, and H. Prade. Representing default rules in possibilistic logic. In *Proceedings KR-1992*, pp. 673–684.

[6] S. Benferhat, D. Dubois, and H. Prade. Coping with the limitations of rational inference in the framework of possibility theory. In *Proceedings UAI-1996*, pp. 90–97.

[7] S. Benferhat, D. Dubois, and H. Prade. Nonmonotonic reasoning, conditional objects and possibility theory. *Artif. Intell.*, 92(1–2):259–276, 1997.

[8] S. Benferhat, S. Kaci, D. Le Berre, and M.-A. Williams. Weakening conflicting information for iterated revision and knowledge integration. In *Proceedings IJCAI-2001*, pp. 109–118.

[9] M. Berkelaar, J. Dirks, K. Eikland, and P. Notebaert. Reference guide to lp_solve (version 5.1). Available at `http://lpsolve.sourceforge.net/5.1/`.

[10] V. Biazzo and A. Gilio. A generalization of the fundamental theorem of de Finetti for imprecise conditional probability assessments. *Int. J. Approx. Reasoning*, 24(2–3):251–272, 2000.

[11] V. Biazzo, A. Gilio, T. Lukasiewicz, and G. Sanfilippo. Probabilistic logic under coherence: Complexity and algorithms. *Ann. Math. Artif. Intell.*, 45(1–2):35–81, 2005.

[12] V. Biazzo, A. Gilio, T. Lukasiewicz, and G. Sanfilippo. Probabilistic logic under coherence, model-theoretic probabilistic logic, and default reasoning in System P. *Journal of Applied Non-Classical Logics*, 12(2):189–213, 2002.

[13] G. Boole. *An Investigation of the Laws of Thought, on which are Founded the Mathematical Theories of Logic and Probabilities*. Walton and Maberley, London, 1854. (Reprint: Dover Publications, New York, 1958).

[14] D. Dubois and H. Prade. On fuzzy syllogisms. *Computational Intelligence*, 4(2):171–179, 1988.

[15] D. Dubois, H. Prade, and J.-M. Touscas. Inference with imprecise numerical quantifiers. In Z. W. Ras and M. Zemankova, editors, *Intelligent Systems*, chapter 3, pp. 53–72. Ellis Horwood, 1990.

[16] T. Eiter and T. Lukasiewicz. Default reasoning from conditional knowledge bases: Complexity and tractable cases. *Artif. Intell.*, 124(2):169–241, 2000.

[17] R. Fagin, J. Y. Halpern, and N. Megiddo. A logic for reasoning about probabilities. *Inf. Comput.*, 87:78–128, 1990.

[18] N. Friedman and J. Y. Halpern. Plausibility measures and default reasoning. *J. ACM*, 48(4):648–685, 2001.

[19] A. M. Frisch and P. Haddawy. Anytime deduction for probabilistic logic. *Artif. Intell.*, 69(1–2):93–122, 1994.

[20] L. van der Gaag. Computing probability intervals under independency constraints. In *Uncertainty in Artificial Intelligence 6*, pp. 457–466. North-Holland, 1991.

[21] D. M. Gabbay and P. Smets, editors. *Handbook on Defeasible Reasoning and Uncertainty Management Systems*. Kluwer Academic, Dordrecht, Netherlands, 1998.

[22] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, New York, 1979.

[23] A. Gilio. Probabilistic consistency of conditional probability bounds. In *Advances in Intelligent Computing, LNCS* 945, pp. 200–209. Springer, 1995.

[24] R. Giugno, T. Lukasiewicz. P-$\mathcal{SHOQ}(\mathbf{D})$: A probabilistic extension of $\mathcal{SHOQ}(\mathbf{D})$ for probabilistic ontologies in the Semantic Web. In *Proceedings JELIA-2002, LNCS* 2424, pp. 86–97.

[25] M. Goldszmidt and J. Pearl. On the consistency of defeasible databases. *Artif. Intell.*, 52(2):121–149, 1991.

[26] M. Goldszmidt and J. Pearl. Qualitative probabilities for default reasoning, belief revision, and causal modeling. *Artif. Intell.*, 84(1–2):57–112, 1996.

[27] T. Hailperin. *Boole's Logic and Probability*, 2nd ed., North-Holland, 1986.

[28] P. Hansen, B. Jaumard, G.-B. D. Nguetsé, and M. P. de Aragão. Models and algorithms for probabilistic and Bayesian logic. In *Proc. IJCAI-1995*, pp. 1862–1868, 1995.

[29] B. Jaumard, P. Hansen, and M. P. de Aragão. Column generation methods for probabilistic logic. *ORSA J. Comput.*, 3:135–147, 1991.

[30] D. S. Johnson. A catalog of complexity classes. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume A, chapter 2, pp. 67–161. MIT Press, Cambridge, MA, 1990.

[31] G. Kern-Isberner and T. Lukasiewicz. Combining probabilistic logic programming with the power of maximum entropy. *Artif. Intell.*, 157(1–2):139–202, 2004.

[32] S. Kraus, D. Lehmann, and M. Magidor. Nonmonotonic reasoning, preferential models and cumulative logics. *Artif. Intell.*, 14(1):167–207, 1990.

[33] H. E. Kyburg, Jr. *The Logical Foundations of Statistical Inference*. D. Reidel, Dordrecht, Netherlands, 1974.

[34] H. E. Kyburg, Jr. and C.-M. Teng. Evaluating defaults. In *Proceedings NMR-2002*, pp. 257–264.

[35] D. Lehmann and M. Magidor. What does a conditional knowledge base entail? *Artif. Intell.*, 55(1):1–60, 1992.

[36] D. Lehmann. Another perspective on default reasoning. *Ann. Math. Artif. Intell.*, 15(1):61–82, 1995.

[37] T. Lukasiewicz. Local probabilistic deduction from taxonomic and probabilistic know- ledge-bases over conjunctive events. *Int. J. Approx. Reasoning*, 21(1):23–61, 1999.

[38] T. Lukasiewicz. Probabilistic deduction with conditional constraints over basic events. *J. Artif. Intell. Res.*, 10:199–241, 1999.

[39] T. Lukasiewicz. Probabilistic logic programming under inheritance with overriding. In *Proceedings UAI-2001*, pp. 329–336.

[40] T. Lukasiewicz. Probabilistic logic programming with conditional constraints. *ACM Transactions on Computational Logic (TOCL)*, 2(3):289–339, 2001.

[41] T. Lukasiewicz. Probabilistic default reasoning with conditional constraints. *Ann. Math. Artif. Intell.*, 34:35–88, 2002.

[42] T. Lukasiewicz. Nonmonotonic probabilistic reasoning under variable-strength inheri- tance with overriding. *Synthese*, 146(1–2):153–169, 2005.

[43] T. Lukasiewicz. Weak nonmonotonic probabilistic logics. *Artif. Intell.*, 168(1–2):119–161, 2005.

[44] N. J. Nilsson. Probabilistic logic. *Artif. Intell.*, 28(1):71–88, 1986.

[45] C. H. Papadimitriou. *Computational Complexity*. Addison-Wesley, Reading, 1994.

[46] J. Pearl. Probabilistic semantics for nonmonotonic reasoning: A survey. In *Procee- dings KR-1989*, pp. 505–516.

[47] J. Pearl. System Z: A natural ordering of defaults with tractable applications to default reasoning. In *Proceedings TARK-1990*, pp. 121–135.

[48] J. L. Pollock. *Nomic Probabilities and the Foundations of Induction*. Oxford Uni- versity Press, Oxford, 1990.

[49] G. R. Wheeler. A resource bounded default logic. In *Proc. NMR-2004*, pp. 416–422.