Intensional FOL for reasoning about probabilities and probabilistic logic programming

Zoran Majkić

ISRST,
Roma, Italy
Email: majk.1234@yahoo.com

Bhanu Prasad*

Department of Computer and Information Sciences,
Florida A&M University,
Tallahassee, Florida 32307, USA
Email: bhanu.prasad@famu.edu
*Corresponding author

Abstract: It is important to have a logic, both for computation of probabilities and for reasoning about probabilities, with well-defined syntax and semantics. The current approaches, which are based on Nilsson's probability structures/logics as well as linear inequalities, to reason about probabilities, have some deficiencies. In this research, we have presented a complete revision of those approaches and have shown that the logic for reasoning about probabilities can be naturally embedded into a 2-valued intensional first-order logic (FOL) with intensional abstraction, by avoiding current ad-hoc system composed of two different 2-valued logics: one for the classical propositional logic at a lower-level and a new one, at a higher-level, for probabilistic constraints with probabilistic variables. The theoretical results that are obtained are applied to probabilistic logic programming.

Keywords: probabilities; 2-valued intensional first-order logic; Nilsson's probability structures; linear inequalities.

Reference to this paper should be made as follows: Majkić, Z. and Prasad, B. (2018) 'Intensional FOL for reasoning about probabilities and probabilistic logic programming', *Int. J. Intelligent Information and Database Systems*, Vol. 11, No 1, pp.79–96.

Biographical notes: Zoran Majkić received his Master of Telecommunication in Electromagnetic Engineering from ETF University of Belgrade, where he worked for two years as an Assistant Professor. After that he changed his work and started doing research in computer science and received his PhD in Computer Science from La Sapienza University, Roma, Italy. He is currently working as an Information Technology Advisor and also doing research in computer science and quantum physics. His research interests include knowledge-base and database systems, artificial intelligence and category theory and algebras.

Bhanu Prasad received his Master of Technology and PhD, both in Computer Science, from Andhra University and Indian Institute of Technology Madras, respectively. He is currently working as a Professor in the Department of Computer and Information Sciences at Florida A&M University in Tallahassee, Florida, USA. His current research interests include artificial intelligence and knowledge-based systems.

1 Introduction

In this research, we consider the probabilistic semantics for the propositional logic (which can be easily extended to predicate logics as well) (Nilsson, 1986; Fagin et al., 1990) with a fixed finite set $\Phi = \{p_1, ..., p_n\}$ of primitive propositions, which can be thought of as corresponding to basic probabilistic events. The set $\mathcal{L}(\Phi)$ of the propositional formulae is the closure of Φ under the Boolean operations for conjunction and negation, $\wedge$ and $\neg$, that is, $(\Phi, \{\wedge, \neg\})$ is the set of all formulae of the propositional logic.

In order to give the probabilistic semantics to such formulae, we first need to briefly review the probability theory [see, for example, Feller (1957) and Halmos (1950)]:

A probability space $(S, \mathcal{X}, \mu)$ consists of a set S, called the sample space, a σ-algebra $\mathcal{X}$ of subsets of S (i.e., a set of subsets of S containing S and closed under complementation and countable union, but not necessarily consisting of all subsets of S) whose elements are called measurable sets, and a probability measure $\mu : \mathcal{X} \to [0, 1]$ where $[0, 1]$ is the closed interval of reals from zero to one. This mapping satisfies Kolmogorov axioms (Kolmogorov, 1986):

A.1 $\mu \geq 0$ for all $X \in \mathcal{X}$.

A.2 $\mu(S) = 1$.

A.3 $\mu(\bigcup_{i\geq 1} X_i) = \sum_{i\geq 1} \mu(X_i)$, if X_i's are nonempty pairwise disjoint members of $\mathcal{X}$.

The $\mu(\{s\})$ is the value of probability in a single point of space s.

The property A.3 is called *countable additivity* for the probabilities in a space S. If $\mathcal{X}$ is a finite set, then the above property A.3 can be simplified as

A.3' $\mu(X \bigcup Y) = \mu(X) + \mu(Y)$,

if X and Y are disjoint members of $\mathcal{X}$, or, equivalently, to the following axiom:

A.3" $\mu(X) = \mu(X \bigcap Y) + \mu(X \bigcap \overline{Y})$,

where $\overline{Y}$ is the compliment of Y in S, so that $\mu(\overline{X}) = 1 - \mu(X)$.

In this research, we consider only finite sample space S, so that $\mathcal{X} = \mathcal{P}(S)$ is the set of all subsets of S. Thus, in our case of a finite set S, we obtain, form A.1 and A.2, that for any $X \in \mathcal{P}(S)$, $\mu(X) = \sum_{s\in X} \mu(\{s\})$.

Based on the work of Nilsson (1986), we can define, for a given propositional logic with a finite set of primitive proposition Φ, the sample space $S = \mathbf{2}^\Phi$, where $\mathbf{2} = \{0, 1\} \subset [0, 1]$, so that the probability space is equal to the Nilsson structure $N = (\mathbf{2}^\Phi, \mathcal{P}(\mathbf{2}^\Phi), \mu)$.

In his work [page 72, line 4–6 in Nilsson (1986)], Nilsson considered a probabilistic logic "which the truth values of sentences can range between 0 and 1. The truth value of a sentence in *probabilistic logic* is taken to be the probability of that sentence in ordinary first-order logic". That is, he considered this logic as a kind of a *many-valued* logic, but not a compositional truth-valued logic. However in his paper, he did not define the formal syntax and semantics for such a probabilistic logic, but defined only the matrix equations where the probability of a sentence $\phi \in \mathcal{L}(\Phi)$ is the sum of the probabilities of the sets of possible worlds (equal to the set $S = \mathbf{2}^{\Phi}$) in which that sentence is *true*. Therefore he assigned *two* different logic values to each sentence ϕ: one is its probability value and another is a classic 2-valued truth value in a given possible world. It is formally contradicting with his intension paraphrased above. In fact, as we discussed in one of the following sections, the correct formalization of such a many-valued logic *with probabilistic semantics* is different, and more complex, from his intuitive initial idea.

The *logic* inadequacy of this seminal work of Nilsson (1986) was also considered in Fagin and Halpern (1989), by extending this Nilsson structure $N = (\mathbf{2}^{\Phi}, \mathcal{P}(\mathbf{2}^{\Phi}), \mu)$ into a more general *probability structure* $M = (\mathbf{2}^{\Phi}, \mathcal{P}(\mathbf{2}^{\Phi}), \mu, \pi)$, where π associates with each $s \in S = \mathbf{2}^{\Phi}$ the truth assignment $\pi(s) : \Phi \to \mathbf{2}$ and we say that $p \in \Phi$ is *true at* s if $\pi(s)(p) = 1$, *false at* s otherwise. This mapping $\pi(s)$ can be uniquely extended to the truth assignment on all formulae in $\mathcal{L}(\Phi)$, by taking the usual rules of propositional logic (the unique homomorphic extension to all formulae), and we can associate to each propositional formula $\phi \in \mathcal{L}(\Phi)$, the set ϕ^M consisting of all states $s \in S$, where ϕ is true (so that $\phi^M = \{s \in S \mid \pi(s)(\phi) = 1\}$). Fagin and Halpern (1989) has demonstrated that, for each Nilsson structure N, there is an equivalent measurable probability structure M, and vice versa.

However, unlike Nilsson, Fagin and Halpern did not define a many-valued propositional logic, but defined a kind of 2-valued logic based on probabilistic constraints. The *weight* or *probability* of ϕ in Nilsson structure N is denoted by $w_N(\phi)$ (corresponding to the value $\mu(\phi^M)$) so that the basic probabilistic 2-valued constraint can be defined by expressions $c_1 \leq w_N(\phi)$ and $w_N(\phi) \leq c_2$ for given constants $c_1, c_2 \in [0, 1]$. Fagin and Halpern expected their logic to be used for reasoning *about* probabilities.

However, again, from the logic point of view, Fagin and Halpern did not define a unique logic but defined *two different* logics: one for the classical propositional logic $\mathcal{L}(\phi)$ and a new one for 2-valued probabilistic constraints obtained from the basic probabilistic formulae above and Boolean operators $\wedge$ and $\neg$. Fagin and Halpern did not consider the introduced symbol w_N as a formal functional symbol for the mapping $w_N : \mathcal{L}(\Phi) \to [0, 1]$ such that for any propositional formula $\phi \in \mathcal{L}(\Phi)$, $w_N(\phi) = \mu(\{s \in S \mid \pi(s)(\phi) = 1\})$. Instead of this intuitive meaning for w_N, they considered each expression $w_N(\phi)$ as a particular probabilistic term; more precisely, as a structured probabilistic *variable* over the domain of values in $[0, 1]$. A multitude of different probabilistic programming languages exist today. Each of these languages employs its own probabilistic primitives, and comes with a particular syntax, semantics, and inference procedure (Baral et al., 2009; Bellodj and Riguzzi, 2013; Raedt and Kimmig, 2015). Due to this, it is hard to understand the underlying programming concepts and appreciate the differences between different languages.

It seams that such a dichotomy and difficulty to have *a unique* 2-valued probabilistic logic, both for the original propositional formulae in $\mathcal{L}(\Phi)$ and for the probabilistic constraints, is based on the fact that if we consider w_N as a function with one argument then it has to be formally represented as a binary predicate $w_N(\phi, a)$ (for the graph of this function) where the first argument is a formula and the second argument is its resulting probability value. Consequently, a constraint "the probability of ϕ is less or equal to c", has to be formally expressed by the logic formula $w_N(\phi, a) \wedge \leq (a, c)$ (here we use the symbol $\leq$ as a built-in rigid binary predicate where $\leq (a, c)$ is equivalent to $a \leq c$), which is a *second-order* syntax because ϕ is a logic *formula* in such a unified logic language. That is, the problem of obtaining the unique logical framework for probabilistic logic comes out with the necessity of a *reification* feature of this logic language, similar to the case of the intensional semantics for RDF data structures (Majkić, 2008).

Consequently, we need a logic which is able to deal directly with reification of logic formulae, and this is the starting point of this work. In fact, as we will see, such an unified logical framework for the probabilistic theory can be achieved by a kind of predicate intensional logics with intensional abstracts that transforms a propositional formulae $\phi \in \mathcal{L}(\Phi)$ into an abstracted *term*, denoted by $\lessdot\phi\gtrdot$. By this approach, the expression $w_N(\lessdot\phi\gtrdot, a) \wedge \leq (a, c)$ remains to be an ordinary first-order formula. In fact, if $\lessdot\phi\gtrdot$ is translated into non-sentence 'that ϕ' then the first-order formula above corresponds to the sentence "the probability *that* ϕ is true is less than or equal to c".

The main motivation for the introduction of the intensionality in the probabilistic-theory of the propositional logic is based on the desire to have the *full* logical embedding of the probability into the FOL, with a clear difference from the classic concept of truth of the logic formulae and the concept of their probabilities. In this way, we are able to replace the ad-hoc syntax and semantics, used in the current practice for probabilistic logic programs (Ng and Subrahmanian, 1992; Dekhtyar and and Dekhtyar, 2004; Udrea et al., 2006; Majkić, 2007) and probabilistic deduction (Majkić, 2009), by the standard syntax and semantics used for the FOL where the probabilistic-theory properties are expressed simply by the particular constraints on their interpretations and models.

The rest of the paper is organized as follows:

In Section 2, we introduce the intensional FOL and its intensional algebra. We define its two-step intensional semantics as a conservative extension of the Tarski's FOL semantics and its Kripke models. In Section 3, we define an embedding of the probability theory, both with reasoning about probabilities, into an intensional FOL with intensional abstraction. Then we show that the probabilities of propositional formulae correspond to the computation of their probabilities in Nilsson's structures, that is, this intensional FOL is sound and complete w.r.t. the measurable probability structures. Finally, in Section 4, we apply the theoretical results, obtained in previous two sections, to the probabilistic logic programming.

2 Intensional FOL language with intensional abstraction

Intensional entities are concepts such as propositions and properties. What make them 'intensional' is that they violate the principle of extensionality; the principle that extensional equivalence implies identity. All (or most) of these intensional entities have been classified at one time or another as kinds of Universals (Bealer, 1993).

We consider a non empty domain $\mathcal{D} = D_{-1} \bigcup D_I$, where a subdomain D_{-1} is made up of particulars (extensional entities), and the rest $D_I = D_0 \bigcup D_1 ... \bigcup D_n ...$ is made up of universals (D_0 for propositions for the 0-ary concepts and $D_n, n \geq 1$, for n-ary concepts).

The fundamental entities are *intensional abstracts* or so called, 'that'-clauses. We assume that they are singular terms; intensional expressions like 'believe', 'mean', 'assert', 'know', are standard two-place predicates that take 'that'-clauses as arguments. Expressions like 'is necessary', 'is true', and 'is possible' are one-place predicates that take 'that'-clauses as arguments. For example, in the intensional sentence "it is necessary that ϕ", where ϕ is a proposition, the 'that ϕ' is denoted by the $\lessdot\phi\gtrdot$, where $\lessdot\gtrdot$ is the intensional abstraction operator which transforms a logic formula into a *term*. Or, for example, 'x believes that ϕ' is given by formula $p_i^2(x, \lessdot\phi\gtrdot)$ (p_i^2 is binary 'believe' predicate).

In this research, we present an intensional FOL with slightly different intensional abstraction than was originally presented in Bealer (1979):

Definition 1: The syntax of the First-order Logic language with intensional abstraction $\lessdot\gtrdot$, denoted by $\mathcal{L}$, is as follows:
Logic operators $(\wedge, \neg, \exists)$; Predicate letters in P (functional letters are considered as particular case of predicate letters); Variables $x, y, z, ..$ in $\mathcal{V}$; Abstraction $\lessdot\_\gtrdot$, and punctuation symbols (comma, parenthesis). With the following simultaneous inductive definition of *term* and *formula*:

1. All variables and constants (0-ary functional letters in P) are terms.
2. If $t_1, ..., t_k$ are terms, then $p_i^k(t_1, ..., t_k)$ is a formula ($p_i^k \in P$ is a k-ary predicate letter).
3. If ϕ and ψ are formulae, then $(\phi \wedge \psi)$, $\neg\phi$, and $(\exists x)\phi$ are formulae.
4. If $\phi(\mathbf{x})$ is a formula (virtual predicate) with a list of free variables in $\mathbf{x} = (x_1, ..., x_n)$ (with ordering from-left-to-right of their appearance in ϕ), and α is its sublist of *distinct* variables, then $\lessdot\phi\gtrdot_\alpha^\beta$ is a term, where β is the remaining list of free variables preserving ordering in $\mathbf{x}$ as well.

The externally quantifiable variables are the *free* variables not in α. When $n = 0$, $\lessdot\phi\gtrdot$ is a term which denotes a proposition, for $n \geq 1$ it denotes a n-ary concept.

An occurrence of a variable x_i in a formula (or a term) is *bound* (*free*) iff it lies (does not lie) within a formula of the form $(\exists x_i)\phi$ (or a term of the form $\lessdot\phi\gtrdot_{\alpha}^{\beta}$ with $x_i \in \alpha$). A variable is free (bound) in a formula (or term) iff it has (does not have) a free occurrence in that formula (or term).

A *sentence* is a formula having no free variables. The binary predicate letter p_1^2 for identity is singled out as a distinguished logical predicate and formulae of the form $p_1^2(t_1, t_2)$ are to be rewritten in the form $t_1 \doteq t_2$. We denote the binary relation, obtained by standard Tarski's interpretation of this predicate p_1^2, by $R_=$. The logic operators $\forall, \vee, \Rightarrow$ are defined in terms of $(\wedge, \neg, \exists)$ in the usual way.

The *intensional interpretation* of this intensional FOL is a mapping between the set $\mathcal{L}$ of formulae of the logic language and intensional entities in $\mathcal{D}$, $I : \mathcal{L} \to \mathcal{D}$ is a kind of 'conceptualization', such that an open-sentence (virtual predicate) $\phi(x_1, ..., x_k)$ with a tuple of all free variables $(x_1, ..., x_k)$ is mapped into a k-ary *concept*, that is, an intensional entity $u = I(\phi(x_1, ..., x_k)) \in D_k$, and (closed) sentence ψ into a proposition (i.e., *logic* concept) $v = I(\psi) \in D_0$ with $I(\top) = Truth \in D_0$ for the FOL tautology $\top$. If a language constant c is a proper name then it is mapped to a particular $a = I(c) \in D_{-1}$, otherwise it is mapped to a corresponding concept in $\mathcal{D}$.

An assignment $g : \mathcal{V} \to \mathcal{D}$ for variables in $\mathcal{V}$ is applied only to free variables in terms and formulae. Such an assignment $g \in \mathcal{D}^{\mathcal{V}}$ can be recursively and uniquely extended to the assignment $g^* : \mathcal{T} \to \mathcal{D}$, where $\mathcal{T}$ denotes the set of all terms (here I is an intensional interpretation of this FOL, as explained next), by:

1 $g^*(t) = g(x) \in \mathcal{D}$ if the term t is a variable $x \in \mathcal{V}$.

2 $g^*(t) = I(c) \in \mathcal{D}$ if the term t is a constant $c \in P$.

3 if t is an abstracted term $\lessdot\phi\gtrdot_{\alpha}^{\beta}$, then $g^*(\lessdot\phi\gtrdot_{\alpha}^{\beta}) = I(\phi[\beta/g(\beta)]) \in D_k, k = |\alpha|$ (i.e., the number of variables in α), where $g(\beta) = g(y_1, .., y_m) = (g(y_1), ..., g(y_m))$ and $[\beta/g(\beta)]$ is a uniform replacement of each i-th variable in the list β with the i-th constant in the list $g(\beta)$. Notice that α is the list of all free variables in the formula $\phi[\beta/g(\beta)]$.

We denote by t/g (or ϕ/g) the ground term (or formula) without free variables and obtained by assignment g from a term t (or a formula ϕ), and by $\phi[x/t]$ the formula obtained by uniformly replacing x by a term t in ϕ.

The distinction between intensions and extensions is important especially because we are now able to have an *equational theory* over intensional entities (as $\lessdot\phi\gtrdot$ which are the predicates or functions 'names'), that is separate from the extensional equality of relations and functions. An extensionalization function h assigns, to the intensional elements of $\mathcal{D}$, an appropriate extension as follows: for each proposition $u \in D_0$, $h(u) \in \{f, t\} \subseteq \mathcal{P}(D_{-1})$ is its extension (true or false value); for each n-ary concept $u \in D_n$, $h(u)$ is a subset of $\mathcal{D}^n$ (n-th Cartesian product of $\mathcal{D}$) and in the case of particulars $u \in D_{-1}$, $h(u) = u$. The sets f and t are empty set $\{\}$ and the set $\{<>\}$ respectively (with the empty tuple $<>\in D_{-1}$ i.e. the unique tuple of 0-ary relation) which may be thought of as falsity and truth respectively, as those used in the Codd's relational-database algebra Codd (1970), while $Truth \in D_0$ is the concept (intension) of the tautology.

We define that $\mathcal{D}^0 = \{<>\}$, hence $\{f, t\} = \mathcal{P}(\mathcal{D}^0)$. Thus we have:

$$h = h_{-1} + \sum_{i \geq 0} h_i : \sum_{i \geq -1} D_i \longrightarrow D_{-1} + \sum_{i \geq 0} \mathcal{P}(D^i)$$

where $h_{-1} = id : D_{-1} \to D_{-1}$ is identity, $h_0 : D_0 \to \{f, t\}$ assigns truth values in $\{f, t\}$ to all propositions, and $h_i : D_i \to \mathcal{P}(D^i)$, for $i \geq 1$, assigns extension to all concepts, where $\mathcal{P}$ is the powerset operator. Thus, intensions can be seen as *names* of abstract or concrete entities, while extensions correspond to various rules that these entities play in different worlds.
Remark: (Tarski's constraint) This semantics has to preserve Tarski's semantics of the FOL, that is, for any formula $\phi \in \mathcal{L}$ with the tuple of free variables $(x_1, ..., x_k)$, for any assignment $g \in \mathcal{D}^{\mathcal{V}}$, and for every $h \in \mathcal{E}$ the following should be satisfied:

$$\text{(T)} \quad h(I(\phi/g)) = t \quad \text{iff} \quad (g(x_1), ..., g(x_k)) \in h(I(\phi)). \qquad \square$$

Thus, intensional FOL has the simple Tarski first-order semantics, with a decidable unification problem, but we also need the actual world mapping which maps any intensional entity to its *actual world extension*. Next we identify a *possible world* by a particular mapping which assigns to intensional entities their extensions in such possible world. That is a direct bridge between intensional FOL and possible worlds representation (Lewis, 1986; Stalnaker, 1984; Montague, 1970, 1973, 1974; Majkić, 2011), where intension of a proposition is a *function* from possible worlds $\mathcal{W}$ to truth-values, and properties and functions from $\mathcal{W}$ to sets of possible (usually not-actual) objects. Here $\mathcal{E}$ denotes the set of possible *extensionalization functions* that satisfy the constraint (T); they can be considered as *possible worlds* [as in Montague's intensional semantics for natural language (Montague, 1970, 1974)], as demonstrated in Majkić (2009, 2008), given by the bijection is:

$$\mathcal{W} \simeq \mathcal{E}.$$

Now we are able to formally define this intensional semantics Majkić (2011):

Definition 2 Two-step intensional semantics: Let $\mathfrak{R} = \bigcup_{k \in \mathbb{N}} \mathcal{P}(\mathcal{D}^k) = \sum_{k \in \mathbb{N}} \mathcal{P}(D^k)$ be the set of all k-ary relations, where $k \in \mathbb{N} = \{0, 1, 2, ...\}$. Notice that $\{f, t\} = \mathcal{P}(\mathcal{D}^0) \in \mathfrak{R}$, that is, the truth values are extensions in $\mathfrak{R}$. The intensional semantics of the logic language with the set of formulae $\mathcal{L}$ can be represented by the mapping

$$\mathcal{L} \ \longrightarrow_I \ \mathcal{D} \ \Longrightarrow_{w \in \mathcal{W}} \ \mathfrak{R}$$

where $\longrightarrow_I$ is a *fixed intensional* interpretation $I : \mathcal{L} \to \mathcal{D}$ and $\Longrightarrow_{w \in \mathcal{W}}$ is *the set* of all extensionalization functions $h = is(w) : \mathcal{D} \to \mathfrak{R}$ in $\mathcal{E}$, where $is : \mathcal{W} \to \mathcal{E}$ is the mapping from the set of possible worlds to the set of extensionalization functions.
We define the mapping $I_n : \mathcal{L}_{op} \to \mathfrak{R}^{\mathcal{W}}$, where $\mathcal{L}_{op}$ is the subset of formulae with free variables (virtual predicates), such that for any virtual predicate $\phi(x_1, ..., x_k) \in \mathcal{L}_{op}$ the mapping $I_n(\phi(x_1, ..., x_k)) : \mathcal{W} \to \mathfrak{R}$ is the Montague's meaning (i.e., *intension*) of this virtual predicate (Lewis, 1986; Stalnaker, 1984; Montague, 1970, 1973, 1974), that is, the mapping which returns with the extension of this (virtual) predicate in every possible world in $\mathcal{W}$.

We adopted this two-step intensional semantics, instead of well known Montague's semantics (which lies in the construction of a compositional and recursive semantics that covers both intension and extension), because it has several weaknesses.

Example 1: Let us consider the following two past participles: 'bought' and 'sold'(with unary predicates $p_1^1(x)$, 'x has been bought', and $p_2^1(x)$, 'x has been sold'). These two different concepts in the Montague's semantics would have not only the same extension but also have the same intension, based on the fact that their extensions are identical in every possible world. With the two-steps formalism, we can avoid this problem by assigning two different concepts (meanings) $u = I(p_1^1(x))$ and $v = I(p_2^1(x))$ in $\in D_1$. Notice that the same problem we have in the Montague's semantics for two sentences with different meanings, which bear the same truth value across all possible worlds: in the Montague's semantics, they will be forced to the *same* meaning.□

Another relevant question w.r.t. this two-step interpretations of an intensional semantics is how the extensional identity relation $\doteq$ (binary predicate of the identity) of the FOL is managed in it. Here this extensional identity relation is mapped into the binary concept $Id = I(\doteq (x,y)) \in D_2$, such that $(\forall w \in \mathcal{W})(is(w)(Id) = R_=)$, where $\doteq (x,y)$ (i.e., $p_1^2(x,y)$) denotes an atom of the FOL of the binary predicate for identity in FOL, usually written by FOL formula $x \doteq y$ (here we prefer to distinguish this *formal symbol* $\doteq \in P$ of the built-in identity binary predicate letter in the FOL from the standard mathematical symbol '$=$' used in all mathematical definitions in this paper).

Next, we use the function $f_{<>} : \mathfrak{R} \rightarrow \mathfrak{R}$ such that for any $R \in \mathfrak{R}$, $f_{<>}(R) = \{<>\}$ if $R \neq \emptyset$; $\emptyset$ otherwise. Let us define the following set of algebraic operators for relations in $\mathfrak{R}$:

1. Binary operator $\bowtie_S : \mathfrak{R} \times \mathfrak{R} \rightarrow \mathfrak{R}$ such that for any two relations $R_1, R_2 \in \mathfrak{R}$, the $R_1 \bowtie_S R_2$ is equal to the relation obtained by natural join of these two relations `if` S is a non empty set of pairs of joined columns of respective relations (where the first argument in the pair is the column index of the relation R_1 while the second argument is the column index of the joined column of the relation R_2); 'otherwise' it is equal to the cartesian product $R_1 \times R_2$. For example, the logic formula $\phi(x_i, x_j, x_k, x_l, x_m) \wedge \psi(x_l, y_i, x_j, y_j)$ will be traduced by the algebraic expression $R_1 \bowtie_S R_2$ where $R_1 \in \mathcal{P}(\mathcal{D}^5)$ and $R_2 \in \mathcal{P}(\mathcal{D}^4)$ are the extensions for a given Tarski's interpretation of the virtual predicates ϕ and ψ relatively, so that $S = \{(4,1),(2,3)\}$ and the resulting relation will have the following ordering of attributes: $(x_i, x_j, x_k, x_l, x_m, y_i, y_j)$.
2. Unary operator $\sim : \mathfrak{R} \rightarrow \mathfrak{R}$ such that for any k-ary (with $k \geq 0$) relation $R \in \mathcal{P}(\mathcal{D}^k) \subset \mathfrak{R}$ we have that $\sim (R) = \mathcal{D}^k \backslash R \in \mathcal{D}^k$, where '$\backslash$' is the subtraction of relations. For example, the logic formula $\neg\phi(x_i, x_j, x_k, x_l, x_m)$ will be traduced by the algebraic expression $\mathcal{D}^5 \backslash R$ where R is the extensions for a given Tarski's interpretation of the virtual predicate ϕ.
3. Unary operator $\pi_{-m} : \mathfrak{R} \rightarrow \mathfrak{R}$ such that for any k-ary (with $k \geq 0$) relation $R \in \mathcal{P}(\mathcal{D}^k) \subset \mathfrak{R}$ we have that $\pi_{-m}(R)$ is equal to the relation obtained by elimination of the m-th column of the relation R `if` $1 \leq m \leq k$ and $k \geq 2$; equal to $f_{<>}(R)$ `if` $m = k = 1$; `otherwise` it is equal to R. For example, the logic formula $(\exists x_k)\phi(x_i, x_j, x_k, x_l, x_m)$ will be traduced by the algebraic expression $\pi_{-3}(R)$ where R is the extensions for a given Tarski's interpretation of the virtual

predicate ϕ and the resulting relation will have the following ordering of attributes: (x_i, x_j, x_l, x_m).

Notice that the ordering of attributes of resulting relations corresponds to the method used for generating the ordering of variables in the tuples of free variables adopted for virtual predicates. Analogous to Boolean algebras, which are extensional models of propositional logic, we introduce an intensional algebra for this intensional FOL:

Definition 3 : Intensional algebra for the intensional FOL in Definition 2 is a structure $Alg_{int} = (\mathcal{D}, f, t, Id, Truth, \{conj_S\}_{S\in\mathcal{P}(\mathbb{N}^2)}, neg, \{exists_n\}_{n\in\mathbb{N}})$, with binary operations $conj_S : D_I \times D_I \to D_I$, unary operation $neg : D_I \to D_I$, unary operations $exists_n : D_I \to D_I$ such that for any extensionalization function $h \in \mathcal{E}$, $u \in D_k, v \in D_j$, $k, j \geq 0$

1 $h(Id) = R_=$ and $h(Truth) = \{<>\}$.

2 $h(conj_S(u, v)) = h(u) \bowtie_S h(v)$, where $\bowtie_S$ is the natural join operation defined above and $conj_S(u, v) \in D_m$ where $m = k + j - |S|$ if for every pair $(i_1, i_2) \in S$ it holds that $1 \leq i_1 \leq k$, $1 \leq i_2 \leq j$ (otherwise $conj_S(u, v) \in D_{k+j}$).

3 $h(neg(u)) = \sim (h(u)) = \mathcal{D}^k \backslash (h(u))$, where $\sim$ is the operation defined above and $neg(u) \in D_k$.

4 $h(exists_n(u)) = \pi_{-n}(h(u))$, where π_{-n} is the operation defined above and $exists_n(u) \in D_{k-1}$ if $1 \leq n \leq k$ (otherwise $exists_n$ is the identity function).

Notice that for $u \in D_0$, $h(neg(u)) = \sim (h(u)) = \mathcal{D}^0 \backslash (h(u)) = \{<>\} \backslash (h(u)) \in \{f, t\}$.

We define a derived operation $union : (\mathcal{P}(D_i)\backslash\emptyset) \to D_i$, $i \geq 0$ such that, for any $B = \{u_1, ..., u_n\} \in \mathcal{P}(D_i)$, we have that $union(\{u_1, ..., u_n\}) =_{def} u_1$ if $n = 1$; $neg(conj_S(neg(u_1), conj_S(..., neg(u_n))...)$, where $S = \{(l, l) \mid 1 \leq l \leq i\}$, otherwise. Then we obtain that for $n \geq 2$:

$$\begin{aligned} h(union(B) &= h(neg(conj_S(neg(u_1), conj_S(..., neg(u_n))...) \\ &= \mathcal{D}^i \backslash ((\mathcal{D}^i \backslash h(u_1) \bowtie_S ... \bowtie_S (\mathcal{D}^i \backslash h(u_n)) \\ &= \mathcal{D}^i \backslash ((\mathcal{D}^i \backslash h(u_1) \bigcap ... \bigcap (\mathcal{D}^i \backslash h(u_n)) \\ &= \bigcup \{h(u_j) \mid 1 \leq j \leq n\} = \bigcup \{h(u) \mid u \in B\}. \end{aligned}$$

Once a method for specifying the interpretations of singular terms of $\mathcal{L}$ has been found (by taking the particularity of abstracted terms into consideration), the Tarski-style definitions of truth and validity for $\mathcal{L}$ may be given in the customary way. What is being sought specifically is a method for characterizing the intensional interpretations of singular terms of $\mathcal{L}$ in such a way that a given singular abstracted term $\lessdot\phi\gtrdot_\alpha^\beta$ will denote an appropriate property, relation, or proposition, depending on the value of $m = |\alpha|$. Thus, we define the mapping of intensional abstracts (terms) into $\mathcal{D}$ differently from one that was given in the version of Bealer (1982), as follows:

Definition 4: An intensional interpretation I can be extended to abstracted terms as follows: for any abstracted term $\lessdot\phi\gtrdot_\alpha^\beta$ we define that

$$I(\lessdot\phi\gtrdot_\alpha^\beta) = union(\{I(\phi[\beta/g(\beta)]) \mid g \in \mathcal{D}^{\overline{\beta}}\})$$

where $\overline{\beta}$ denotes the 'set' of elements in the list β, and the assignments in $\mathcal{D}^{\overline{\beta}}$ are limited only to the variables in $\overline{\beta}$.

Remark: Can we extend the interpretation also to (abstracted) terms, because in Tarski's interpretation of FOL we do not have any interpretation for terms, but only the assignments for terms as we defined previously by the mapping $g^* : \mathcal{T} \to \mathcal{D}$? Yes because the abstraction symbol $\lessdot\_\gtrdot_\alpha^\beta$ can be considered as a kind of the unary built-in functional symbol of intensional FOL so that we can apply the Tarskian interpretation to this functional symbol into the fixed mapping $I(\lessdot\_\gtrdot_\alpha^\beta) : \mathcal{L} \to \mathcal{D}$ so that for any $\phi \in \mathcal{L}$ we have that $I(\lessdot\phi\gtrdot_\alpha^\beta)$ is equal to the application of this function to the value ϕ, that is, to $I(\lessdot\_\gtrdot_\alpha^\beta)(\phi)$. In such an approach, we would also introduce the typed variable X for the formulae in $\mathcal{L}$ so that the Tarskian assignment for this functional symbol with variable X, with $g(X) = \phi \in \mathcal{L}$, can be given by:

$$\begin{aligned} g^*(\lessdot\_\gtrdot_\alpha^\beta (X)) &= I(\lessdot\_\gtrdot_\alpha^\beta)(g(X)) = I(\lessdot\_\gtrdot_\alpha^\beta)(\phi) \\ &= <>\in D_{-1} \end{aligned}$$

if $\overline{\alpha} \bigcup \overline{\beta}$ is not equal to the set of free variables in ϕ:

$$= union(\{I(\phi[\beta/g'(\beta)]) \mid g' \in \mathcal{D}^{\overline{\beta}}\}) \in D_{|\overline{\alpha}|}, \text{otherwise.} \qquad \square$$

Notice that if $\beta = \emptyset$ is the empty list then $I(\lessdot\phi\gtrdot_\alpha^\beta) = I(\phi)$. Consequently, the denotation of $\lessdot\phi\gtrdot$ is equal to the meaning of a proposition ϕ, that is, $I(\lessdot\phi\gtrdot) = I(\phi) \in D_0$. In the case when ϕ is an atom $p_i^m(x_1,..,x_m)$ then $I(\lessdot p_i^m(x_1,..,x_m)\gtrdot_{x_1,..,x_m}) = I(p_i^m(x_1,..,x_m)) \in D_m$, while $I(\lessdot p_i^m(x_1,..,x_m)\gtrdot^{x_1,..,x_m}) = union(\{I(p_i^m(g(x_1),...,g(x_m))) \mid g \in \mathcal{D}^{\{x_1,..,x_m\}}\}) \in D_0$, with $h(I(\lessdot p_i^m(x_1,..,x_m)\gtrdot^{x_1,..,x_m})) = h(I((\exists x_1)...(\exists x_m)p_i^m(x_1,..,x_m))) \in \{f,t\}$. For example, $h(I(\lessdot p_i^1(x_1) \wedge \neg p_i^1(x_1)\gtrdot^{x_1})) = h(I((\exists x_1)(\lessdot p_i^1(x_1) \wedge \neg p_i^1(x_1)\gtrdot^{x_1}))) = f$.

The interpretation of a more complex abstract $\lessdot\phi\gtrdot_\alpha^\beta$ is defined in terms of the interpretations of the relevant syntactically simpler expressions, because the interpretation of more complex formulae is defined in terms of the interpretation of the relevant syntactically simpler formulae, based on the intensional algebra above. For example, $I(p_i^1(x) \wedge p_k^1(x)) = conj_{\{(1,1)\}}(I(p_i^1(x)), I(p_k^1(x)))$, $I(\neg\phi) = neg(I(\phi))$, $I(\exists x_i)\phi(x_i, x_j, x_i, x_k) = exists_3(I(\phi))$.

Consequently, based on the intensional algebra in Definition 2 and on intensional interpretations of abstracted term in Definition 2, it holds that the interpretation of any formula in $\mathcal{L}$ (and any abstracted term) will be reduced to an algebraic expression over interpretation of primitive atoms in $\mathcal{L}$. This obtained expression is finite for any finite formula (or abstracted term), and represents the *meaning* of such finite formula (or abstracted term).

The *extension* of abstracted terms satisfies the following property: For any abstracted term $\lessdot\phi\gtrdot_\alpha^\beta$ with $|\alpha| \geq 1$ we have that $h(I(\lessdot\phi\gtrdot_\alpha^\beta)) = \pi_{-\beta}(h(I(\phi)))$, where

$\pi_{-(y_1,...,y_k)} = \pi_{-y_1} \circ ... \circ \pi_{-y_1}$, $\circ$ is the sequential composition of functions), and $\pi_{-\emptyset}$ is an identity.

We can connect $\mathcal{E}$ with a possible-world semantics. Such correspondence is a natural identification of intensional logics with modal Kripke based logics.

Definition 5 (Model): A model for intensional FOL with fixed intensional interpretation I, which expresses the two-step intensional semantics in Definition 2, is the Kripke structure $\mathcal{M}_{int} = (\mathcal{W}, \mathcal{D}, V)$, where $\mathcal{W} = \{is^{-1}(h) \mid h \in \mathcal{E}\}$, a mapping $V : \mathcal{W} \times P \to \bigcup_{n<\omega}\{t, f\}^{\mathcal{D}^n}$, with P a set of predicate symbols of the language, such that for any world $w = is^{-1}(h) \in \mathcal{W}, p_i^n \in P$, and $(u_1, ..., u_n) \in \mathcal{D}^n$, it holds that $V(w, p_i^n)(u_1, ..., u_n) = h(I(p_i^n(u_1, ..., u_n)))$. The satisfaction relation $\models_{w,g}$ for a given $w \in \mathcal{W}$ and assignment $g \in \mathcal{D}\mathcal{V}$ is defined as follows:

1 $\mathcal{M} \models_{w,g} p_i^k(x_1, ..., x_k)$ iff $V(w, p_i^k)(g(x_1), ..., g(x_k)) = t$,

2 $\mathcal{M} \models_{w,g} \varphi \wedge \phi$ iff $\mathcal{M} \models_{w,g} \varphi$ and $\mathcal{M} \models_{w,g} \phi$,

3 $\mathcal{M} \models_{w,g} \neg\varphi$ iff not $\mathcal{M} \models_{w,g} \varphi$,

4 $\mathcal{M} \models_{w,g} (\exists x)\phi$ iff

 4.1 $\mathcal{M} \models_{w,g} \phi$, if x is not a free variable in ϕ;

 4.2 exists $u \in \mathcal{D}$ such that $\mathcal{M} \models_{w,g} \phi[x/u]$, if x is a free variable in ϕ.

It is easy to show that the satisfaction relation $\models$ for this Kripke semantics in a world $w = is^{-1}(h)$ is defined by, $\mathcal{M} \models_{w,g} \phi$ iff $h(I(\phi/g)) = t$.

We can enrich this intensional FOL by another modal operator, as for example the 'necessity' universal operator $\Box$ with an accessibility relation $\mathcal{R} = \mathcal{W} \times \mathcal{W}$, obtaining the S5 Kripke structure $\mathcal{M}_{int} = (\mathcal{W}, \mathcal{R}, \mathcal{D}, V)$, in order to be able to define the equivalences Majkić (2012) between the abstracted terms without free variables.

3 Embedding of probabilistic logic into intensional FOL

In order to reason about probabilities of the propositional formulae, we need a kind of 2-valued meta-logic with *reification* features, thus, a kind of intensional FOL with *intensional abstraction* presented previously. Consequently, the sentence "the probability that A is less than or equal to c" is expressed by the first-order logic formula $w_N(\lessdot\phi\gtrdot, a) \wedge \leq (a, c)$, where $\leq$ is the binary built-in predicate 'is less than or equal', where the usual notation '$a \leq b$' is rewritten in this standard predicate-based way by '$\leq (a, b)$', while 'the probability that ϕ is equal to a' is denoted by the ground atom $w_N(\lessdot\phi\gtrdot, a)$ with the binary 'functional' predicate symbol w_N (in intensional logic, any n-ary function is represented by the $n+1$-ary predicate symbol with the first n attributes used as arguments of this function and the last $(n+1)$-th attribute for the resulting function's value, analogously as in FOL with identity).

The basic intensional logic language $\mathcal{L}_{PR} \subseteq \mathcal{L}$ for probabilistic theory is composed of propositions in $\mathcal{L}(\Phi)$, with propositional symbols (0-ary predicate symbols) $p_i^0 = p_i \in \Phi$ (with $I(p_i) \in D_0$), with the binary predicate p_3^2 for the weight or probabilistic function w_N, with the binary built-in (with constant fixed extension in any "world' $h \in \mathcal{E}$) predicate p_2^2 for $\leq$ (the binary predicate $=$ for identity is defined by $a = b$ iff

$a \leq b$ and $b \leq a$), and with two built-in ternary predicates p_1^3 and p_2^3 and denoted by $\oplus$ and $\odot$ for addition and multiplication operations $+$ and $\cdot$ respectively as required for a logic for reasoning about probabilities (Fagin et al., 1990). The 0-ary functional symbols $a, b, c, ..$ in this logic language will be used as numeric constants for denotation of probabilities in $[0,1]$, i.e., with $I(a) = \overline{a} \in [0,1] \subset D_{-1}$. Consequently, the 'worlds' (i.e., the extensionalization functions) will be reduced to the mappings

$$h = h_{-1} + h_0 + h_2 + h_3.$$

Note that in intensional FOL, each n-ary functional symbol is represented by the (n+1)-ary predicate letter, where the last attribute (of this predicate) is introduced for the resulting values of such a function. For example, the first attribute of the predicate letter w_N will contain the intensional abstract of a propositional formula in $\mathcal{L}(\Phi)$, while the second place will contain the probabilistic value in the interval of reals $[0,1] \subset D_{-1}$, so that the ground atom $w_N(\lessdot\phi\gtrdot, a)$ in $\mathcal{L}_{PR}$ will have the interpretation $I(w_N(\lessdot\phi\gtrdot, a)) \in D_0$. The atom $w_N(x, y)$, with variables x and y, will satisfy the functional requirements, that is $I(w_N(x, y)) \in D_2$ with a binary relation $R = h(I(w_N(x, y))) \in \mathcal{P}(D_0 \times [0,1]) \subseteq \mathcal{P}(\mathcal{D}^2)$, such that for any $(u, v) \in R$ there is no $v_1 \neq v$ such that $(u, v_1) \in R$. Obviously, for this intensional logic, we have that $h(I(w_N(\lessdot\phi\gtrdot, a))) = t$ iff $(I(\lessdot\phi\gtrdot), I(a)) = (I(\phi), \overline{a}) \in R$.

Analogously, for the ground atom $\oplus(a, b, c)$, with $\overline{a} = I(a), \overline{b} = I(b), \overline{c} = I(c) \in D_{-1}$ real numbers, we have that $I(\oplus(a, b, c)) \in D_0$ such that for any 'world' $h \in \mathcal{E}$ we have that $h(I(\oplus(a, b, c))) = t$ iff $\overline{a} + \overline{b} = \overline{c}$ (remember that for elements $\overline{a} = I(a) \in D_{-1}$ we have that $h(I(a)) = \overline{a}$). For addition of more than two elements in this intensional logic, we will use intensional abstract, for example for the sum of three elements we can use a ground formula $\oplus(a, b, d) \wedge \oplus(d, c, e)$, such that it holds that $h(I(\oplus(a, b, d) \wedge \oplus(d, b, c))) = conj(I(\oplus(a, b, d)), I(\oplus(d, c, e))) = t$ iff $\overline{a} + \overline{b} = \overline{d}$ and $\overline{d} + \overline{c} = \overline{e}$, that is, iff $\overline{a} + \overline{b} + \overline{c} = \overline{e}$. The fixed extensions of the two built-in ternary predicates $\oplus(x, y, z)$ and $\odot(x, y, z)$ are equal to:

$$R_{\oplus} = h(I(\oplus(x, y, z))) = \{(u_1, u_2, u_1 + u_2) \mid u_1, u_2 \in D_{-1} \text{ are real numbers}\},$$

$$R_{\odot} = h(I(\odot(x, y, z))) = \{(u_1, u_2, u_1 \cdot u_2) \mid u_1, u_2 \in D_{-1} \text{ are real numbers}\}.$$

The built-in binary predicate $\leq$ satisfies the following requirements for its intensional interpretation: $I(\leq (x, y)) \in D_2$ such that for every $h \in \mathcal{E}$ it holds that its fixed extension is a binary relation $R_{\leq} = h(I(\leq (x, y))) = \{(u, v) \mid u, v \in D_{-1}$ are real numbers and $u \leq v\}$, with the property that $h(I(\leq (a_1, a_2))) = t$ iff $(I(a_1), I(a_2)) = (\overline{a}_1, \overline{a}_2) \in R_{\leq}$.

Definition 6 : Intensional FOL $\mathcal{L}_{PR}$ is a probabilistic logic with a *probability structure* $M = (\mathbf{2}^{\Phi}, \mathcal{P}(\mathbf{2}^{\Phi}), \mu, \pi)$ if its intentional interpretations satisfy the following property for any propositional formula $\phi \in \mathcal{L}(\Phi) \subseteq \mathcal{L}_{PR}$:

$$h(I(w_N(\lessdot\phi\gtrdot, a))) = t \quad \text{iff} \quad I(a) = \sum_{s \in \mathbf{2}^{\Phi} \ \& \ \pi(s)(\phi) = 1} \mu(\{s\}).$$

Let us show that the binary predicate w_N is a functional built-in predicate, whose extension is equal in every possible 'world' $h \in \mathcal{E}$, and that the probability structure can use $\mathcal{E}$ as the set of possible worlds in the place of Nilsson's set $\mathbf{2}^{\Phi}$. That is, we can replace Nilsson's structure with the intensional probability structure $M_I = (\mathcal{E}, \mathcal{P}(\mathcal{E}), \mu, \pi)$.

Proposition 1: Intensional FOL $\mathcal{L}_{PR}$ is a probabilistic logic with a *probability structure* $M = (\mathbf{2}^{\Phi}, \mathcal{P}(\mathbf{2}^{\Phi}), \mu, \pi)$ if w_N is a built-in functional symbol such that its fixed extension is equal to $R_{w_N} = h(I(w_N(x,y))) = \{(I(\phi), I(a)) \mid \phi \in \mathcal{L}(\Phi)$ and $I(a) = \sum_{h_1 \in \mathcal{E} \ \& \ h_1(I(w_N(\lessdot\phi\gtrdot, a)))=t} \ \mu(\{is^{-1}(h_1)\})\}$ where the mapping $is : \mathbf{2}^{\Phi} \to \mathcal{E}$ is a bijection, and is^{-1} its inverse.

Proof: Let us show that there is a bijection $is : \mathbf{2}^{\Phi} \to \mathcal{E}$ between the sets $\mathbf{2}^{\Phi}$ and $\mathcal{E}$. In fact, let $v \in \mathbf{2}^{\Phi}$ be extended (in the unique standard homomorphic way) to all propositional formulae by $\overline{v} : \mathcal{L}(\Phi) \to \mathbf{2}$. This propositional valuation corresponds to the intensional interpretation (I, h) obtained, for any sentence $\phi \in \mathcal{L}(\Phi)$, by $h(I(\phi)) = is_{\mathbf{2}}(\overline{v}(\phi))$, where $is_{\mathbf{2}} : \mathbf{2} \to \{f, t\}$ is a bijection of these two lattices such that $is_{\mathbf{2}}(0) = f$, $is_{\mathbf{2}}(1) = t$. We have seen that all predicate symbols with arity greater than 0 of our intensional probabilistic logic $\mathcal{L}_{PR}$ are *built-in predicates* (that do not depend on $h \in \mathcal{E}$) so that for a fixed intensional interpretation I, any two extensionalization functions h and h' differ only on propositions in D_0, so that we obtain the bijective mapping $is : v \mapsto h$ such that $v = is_{\mathbf{2}}^{-1} \circ h \circ I$, where $\circ$ denotes the composition of functions. From Tarski's constraint (T) of intensional algebra, we have that for any ground atom $w_N(\lessdot\phi\gtrdot, a)$ it holds that $I(w_N(\lessdot\phi\gtrdot, a)) = t$ iff $(I(\phi), I(a)) \in h(I(w_N(x,y)))$. Thus, form Definition 3, we obtain that $(I(\phi), I(a)) \in h(I(w_N(x,y)))$ iff $I(a) = \sum_{s \in \mathbf{2}^{\Phi} \ \& \ \pi(s)(\phi)=1} \ \mu(\{s\})$. Consequently, $R_{w_N} = h(I(w_N(x,y))) = \{(I(\phi), I(a)) \mid \phi \in \mathcal{L}(\Phi)$ and $I(a) = \sum_{s \in \mathbf{2}^{\Phi} \ \& \ \pi(s)(\phi)=1} \ \mu(\{s\})\}$, where $I(w_N(x,y)) \in D_2$, $I(\phi) \in D_0$ and $I(a) \in [0,1] \subseteq D_{-1}$. But from the bijection is, instead of $s \in \mathbf{2}^{\Phi}$, we can take $h_1 = is(s) \in \mathcal{E}$. The condition $\pi(s)(\phi) = 1$, which means that "ϕ is true in the state s", can be equivalently replaced by "ϕ is true in the world $h_1 = is(s)$", and hence by condition $h_1(I(w_N(\lessdot\phi\gtrdot, a))) = t$. So, the definition for the extension of the binary relation R_{w_N} for Nilsson's probabilities of propositional formulae, given in this proposition, is correct. This extension is constant in any 'possible world' in $\mathcal{E}$ hence the binary functional-predicate w_N is a built-in predicate in this intensional FOL $\mathcal{L}_{PR}$. □

Consequently, the sentence "the probability that ϕ is equal to a", expressed by the ground atom $w_N(\lessdot\phi\gtrdot, a)$, is true iff $h(I(w_N(\lessdot\phi\gtrdot, a))) = t$ iff $(I(\phi), \overline{a}) \in h(I(w_n(x,y)) = R_{w_N}$.

Thus, for the most simple linear inequality, "the probability that ϕ is less than or equal to c", expressed by the formula $\exists x(w_N(\lessdot\phi\gtrdot, x) \wedge \leq (x, c))$, is true iff $h(I(\exists x(w_N(\lessdot\phi\gtrdot, x) \wedge \leq (x, c)))) = t$ iff $(u, I(c)) \in R_{\leq}$, where $u \in [0,1] \subset D_{-1}$ is determined by $(v, u) \in R_{w_N}$ where $v = I(\phi) \in D_0$.

Analogous to the results obtained for a logic for reasoning about probabilities in Fagin et al. (1990), we obtain the following property:

Theorem 1 : The intensional FOL $\mathcal{L}_{PR}$, with built-in binary predicate w_N defined in Proposition 3, built-in binary predicate $\leq$ and ternary built-in predicates $\oplus$ and $\odot$, is sound and complete with respect to the measurable probability structures.

Proof: We will follow the demonstration analogous to the demonstration of Theorem 2.2 in Fagin et al. (1990) for the sound and complete axiomatization of the axiomatic system AX_{MEAS} for logic reasoning about probabilities. It is divided into three parts, which deal respectively with propositional reasoning, reasoning about linear inequalities, and reasoning about probabilities:

1 Propositional reasoning: set of all instances of propositional tautologies, with unique inference rule Modus Ponens.

2 Reasoning about linear inequalities: set of all instances of valid formulae about linear inequalities of the form $a_1 \cdot x_1 + ... + a_k \cdot x_k \leq c$, where $a1, ..., a_k$ and c are integers with $k \geq 1$, while $x1, ..., x_k$ are probabilistic variables.

3 Reasoning about probability function:

   3.1 $w(\phi) \geq 0$ (nonnegativity)

   3.2 $w(true) = 1$ (the probability of the event true is 1)

   3.3 $w(\phi \wedge \psi) + w(\phi \wedge \neg\psi) = w(\phi)$ (additivity)

   3.4 $w(\phi) = w(\psi)$ if $\phi \equiv \psi$ (distributivity).

It is easy to verify that, for any propositional axiom ϕ, we have that for all 'worlds' $h \in \mathcal{E}$ it holds that $h(I(\phi)) = t$, so that it is true in the S5 Kripke model of the intensional FOL given in Definition 2, because all algebraic operations in Alg_{int} in Definition 2 are defined in order to satisfy standard propositional logic. Moreover, the Modus Ponens rule is satisfied in every 'world' $h \in \mathcal{E}$. Thus, the point 1 above is satisfied by intensional logic $\mathcal{L}_{PR}$.
The definition of built-in predicates $\odot, \oplus$ and $\leq$ satisfy all linear inequalities, thus the point 2 above.

The definition of binary predicate $w_N(x, y)$ is given in order to satisfy Nilsson's probability structure, thus all properties of probability funcion in Point 3 above are satisfied by $w_N(x, y)$ built-in predicate in every 'world' $h \in \mathcal{E}$. Consequently, the soundness and completeness of the intensional logic $\mathcal{L}_{PR}$ with respect to measurable probability structures, based on the Theorem 2.2 in Fagin et al. (1990), is satisfied. □

For example, the satisfaction of the linear inequality $a_1 \cdot x_1 + a_2 \cdot x_2 \leq c$, where x_1 and x_2 are the probabilities of the propositional formulae ϕ_1 and ϕ_2 respectively (here the list of quantifications $(\exists x_1)...(\exists x_k)$ is abbreviated by $(\exists x_1, ..., x_k)$), expressed by the following intensional formula

$$
\begin{aligned}
&(\exists x_1, x_2, y_1, y_2, y_3)(w_N(\lessdot\phi_1\gtrdot, x_1) \wedge w_N(\lessdot\phi_2\gtrdot, x_2) \\
&\wedge \odot\, (a_1, x_1, y_1) \\
&\wedge \odot\, (a_2, x_2, y_2) \wedge \oplus(y_1, y_2, y_3) \\
&\wedge \leq (y_3, c)), \text{is true iff} \\
&(I(\phi_1), u_1), (I(\phi_2), u_2) \in R_{w_N}, (I(a_1), u_1, v_1), \\
&(I(a_2), u_2, v_2) \in R_{\odot}, (v_1, v_2, v_3)) \in R_{\oplus} and (v_3, I(c)) \in R_{\leq}
\end{aligned}
$$

where $u_1, u_2, v_1, .., v_3 \in D_{-1}$ are real numbers.

Analogously, the satisfaction of any linear inequality $a_1 \cdot x_1 + ... + a_k \cdot x_k \leq c$, where x_i are the probabilities of the propositional formulae ϕ_i for $i = 1, ..., k,\ k \geq 2$, can be expressed by the logic formula

$$(\exists x_1, ..., x_k, y_1, ..., y_k, z_1, ..., z_k)(w_N(\lessdot\phi_1\gtrdot, x_1) \wedge ...$$
$$\wedge w_N(\lessdot\phi_k\gtrdot, x_k) \wedge \odot(a_1, x_1, y_1) \wedge ... \wedge \odot(a_k, x_k, y_k)$$

$$\wedge \oplus (0, y_1, z_1) \wedge \oplus(z_1, y_2, z_2) \wedge ... \wedge \oplus(z_{k-1}, y_k, z_k)$$
$$\wedge \leq (z_k, c)).$$

Based on the Theorem 2.2 in Fagin et al. (1990) we can conclude that the problem of deciding whether such an intensional formula in $\mathcal{L}_{PR}$ is satisfiable in a measurable probability structure of Nilsson is NP-complete.

4 Application to probabilistic logic programs

The semantics of the interval-based probabilistic logic programs, based on possible worlds with the fixpoint semantics for such programs (Ng and Subrahmanian, 1992), has been considered valid for more than 13 years. However, some years ago, when one of the authors (Majkic) worked with Prof.V.S.Subrahmanian, director of the UMIACS institute, he had the opportunity to consider the general problems of (temporal) probabilistic databases (Majkić et al., 2007) and to analyse the semantics of Subrahmanian's interval-based probabilistic logic programs. Majkic then realised that, unfortunately, the semantics was not correctly defined.

Because of that, Majkić (2005) formally developed the reduction of (temporal) probabilistic databases into constraint logic programs (CLP). Consequently, it was possible to apply the interval probabilistic satisfiability (interval PSAT) in order to find the models of such interval-based probabilistic programs, as presented and compared with other approaches in Majkić (2007). Moreover, in the complete revision presented in this paper, it was demonstrated that the temporal-probabilistic logic programs can be reduced to a particular case of the ordinary probabilistic logic programs, hence we can apply intensional semantics only to this last general case of logic programs.

Next, we introduce the syntax of probabilistic logic programs. More about it can be found in the original work in Ng and Subrahmanian (1992) and in its revision in Majkić (2007). Let $ground(P)$ denotes the set of all ground instances of rules of a Probabilistic Logic Program P with a given domain for object variables, and let H denotes the Herbrand base of this program P. Then, each ground instance of rules in $ground(P)$ has the following syntax:

$$A : \mu_0 \leftarrow \phi_1 : \mu_1 \wedge ... \wedge \phi_m : \mu_m \tag{1}$$

where $A \in H$ is a ground atom in a Herebrand base H; $\phi_i, i \geq 1$ are logic formulae composed by ground atoms and standard logic connectives $\wedge$ and $\neg$, while $\mu_i = (b_i, c_i), i \geq 0$, where $b_i, c_i \in [0, 1]$, are the lower and upper probability boundaries.

The expression $\phi_i : \mu_i$ is a probabilistic-annotated (p-annotated) basic formula, which is true if the probability x_i of the ground formula ϕ_i is between b_i and c_i; false otherwise. Thus, this basic p-annotated formula is the particular case of the 2-valued probabilistic formula:

$$(1 \cdot x_i \geq a_i) \wedge (1 \cdot x_i \leq b_i) \tag{2}$$

composed by two linear inequalities.

Consequently, the standard logic embedding of annotated interval-based logic programs can be easily obtained by the intensional logic $\mathcal{L}_{PR}$ described in Section 3 where Φ is equal to the Herbrand base H of the annotated interval-based probabilistic logic program P.

Thus, based on the translation (2), the logic formula in intensional logic $\mathcal{L}_{PR}$ corresponds to basic annotated formula $\phi_i : \mu_i$ of the annotated logic program $ground(P)$ is equal to the following first-order closed formulae with a variable x_i:

$$\exists x_i (w_N(\lessdot \phi_i \gtrdot, x_i) \wedge \leq (b_i, x_i) \wedge \leq (x_i, c_i)).$$

Based on this translation, the rule (1) of the annotated logic program $ground(P)$ can be replaced by the following rule of an intensional probabilistic logic program:

$$\begin{aligned}&\exists x_0 (w_N(\lessdot A \gtrdot, x_0) \wedge \leq (b_0, x_0) \wedge \leq (x_0, c_0)) \quad \leftarrow \quad \exists x_1 (w_N(\lessdot \phi_1 \gtrdot, x_1)\\ &\wedge \leq (b_1, x_1) \wedge \leq (x_1, c_1)) \wedge \ldots \wedge \ \exists x_m (w_N(\lessdot \phi_m \gtrdot, x_m) \wedge \leq (b_m, x_m)\\ &\wedge \leq (x_m, c_m))\end{aligned}$$

with the variables $x_0, x_1, \ldots, x_m$.

In this way, we obtain a grounded intensional probabilistic logic program P_{PR}, which has both the syntax and semantics different from the original annotated probabilistic logic program $ground(P)$.

As an alternative to this full intensional embedding of the annotated logic programs into the first-order intensional logic, we can use a partial embedding by preserving the old ad hoc annotated syntax of the probabilistic program $ground(P)$, by extending the standard predicate-based syntax of the intensional FOL logic with annotated formulae, and by defining only the new intensional interpretation I for these annotated formulae as follows:

$$\begin{aligned}I(\phi_i : \mu_i) &= I((\exists x)(w_N(\lessdot \phi_i \gtrdot, x)) \wedge (\leq (b_i, x)) \wedge \leq (x, c_i)))\\ &= exist(conj_{\{(1,1)\}}(I(w_N(\lessdot \phi_i \gtrdot, x)), conj_{\{(1,1)\}}(I(\leq (b_i, x)),\\ &\quad I(\leq (x, c_i)))))\end{aligned}$$

where $I(w_N(\lessdot \phi_i \gtrdot, x)), I(\leq (b_i, x)), I(\leq (x, c_i)) \in D_1$. So that $h(I(\phi_i : \mu_i)) = t$ iff $h(exist(u_1)) = t$,

where $u_1 = conj_{\{(1,1)\}}(I(w_N(\lessdot \phi_i \gtrdot, x)), conj_{\{(1,1)\}}(I(\leq (b_i, x)), I(\leq (x, c_i)))) \in D_1$, iff $\exists u (u \in h(u_1))$ iff $(I(b_i), u), (u, I(c_i)) \in R_{\leq}$, where $u \in [0, 1] \subset D_{-1}$ is a particular assignment for a variable x determined by $(v, u) \in R_{w_N}$ where $v = I(\phi_i)$.

Notice that, from the fact that $u_1 \in D_1$

$$\begin{aligned}&h(exist(u_1))\\ &\quad = f_{<>}(h(I(w_N(\lessdot \phi_i \gtrdot, x)) \bowtie_{\{(1,1)\}} (h(I(\leq (b_i, x))) \bowtie_{\{(1,1)\}} h(I(\leq (x, c_i)))))\\ &\quad = f_{<>}(h(I(w_N(\lessdot \phi_i \gtrdot, x)) \bigcap h(I(\leq (b_i, x))) \bigcap h(I(\leq (x, c_i)))\end{aligned}$$

where $f_{<>}(R) = f$ if $R = \emptyset$; t otherwise.

The advantage of this second partial embedding is that we can preserve the old syntax for (temporal) probabilistic logic programs (Ng and Subrahmanian, 1992; Majkić, 2007) and are also providing to them the standard intensional FOL semantics instead of having the current ad-hoc semantics for such kind of logic programs.

5 Conclusions

The logic for reasoning *about* probabilities can be embedded into an intensional FOL that remains to be 2-valued logic, both for propositional formulae in $\mathcal{L}(\Phi)$ and predicate formulae for probability constraints, based on the binary built-in predicate $\leq$ and binary predicate w_N used for the probability function, where the basic propositional letters in Φ are formally considered as nullary predicate symbols.

The intensional FOL for reasoning about probabilities is obtained by a particular fusion of the intensional algebra (analogous to Bealer's approach) and Montague's possible-worlds modal logic for the semantics of the natural language. In this paper, we enriched such a logic framework by a number of built-in binary and ternary predicates, which can be used to define the basic set of probability inequalities and to render the probability weight function w_N an explicit object in this logic language. We conclude that this intensional FOL logic with intensional abstraction is a good candidate language for the specification of probabilistic logic programs, and we applied two different approaches for this: the first one is obtained by the translation of the annotated syntax of current logic programs into this intensional FOL; the second one, instead, modifies only the semantics of these logic programs by preserving their current ad-hoc annotated syntax.

References

Baral, C., Gelfond, M. and Rushton, J.N. (2009) 'Probabilistic reasoning with answer sets', *Theory and Practice of Logic Programming*, Vol. 9, No. 1, pp.57–144.

Bealer, G. (1979) 'Theories of properties, relations, and propositions', *The Journal of Philosophy*, Vol. 76, No. 11, pp.634–648.

Bealer, G. (1982) *Quality and Concept*, Oxford University Press, USA.

Bealer, G. (1993) 'Universals', *The Journal of Philosophy*, Vol. 90, No. 1, pp.5–32.

Bellodj, E. and Riguzzi, F. (2013) 'Expectation maximization over bynary decision dyagrams for probabilistic logic programs', *Intell. Data Anal.*, Vol. 17, No. 2, pp.343–363.

Codd, E.F. (1970) 'A relational model of data for large shared data banks', *Communications of the ACM (Association for Computing Machinery)*, Vol. 13, No. 6, pp.377–387.

Dekhtyar, A. and Dekhtyar, M.I. (2004) 'Possible worlds semantics for probabilistic logic programs', *ICLP 2004*, 2004, pp.137–148.

Fagin, R. and Halpern, J.Y. (1989) 'Uncertainty, belief, and probability', *IJCAI*, Vol. 89, pp.1161–1167.

Fagin, R., Halpern, J. and Megiddo, N. (1990) 'A logic for reasoning about probabilities', *Information and Computation*, Vol. 87, Nos. 1–2, pp.78–128.

Feller, W. (1957) *An Introduction to* Probability Theory *and its Applications*, Vol. 1, 2nd ed., Wiley, New York.

Halmos, P. (1950) 'Measure theory', *Van Nostrand*.

Kolmogorov, A.N. (1986) *Selected Works of* A.N.K*olmogorov: Vol.2* P*robability Theory and Mathematical Statistics*, A.N. Shiryayev (Ed.), Nauka, Moscow.

Lewis, D.K. (1986) *On the Plurality of Worlds*, Blackwell, Oxford.

Majkić, Z. (2005) 'Constraint logic programming and logic modality for event's valid-time approximation', *2nd Indian International Conference on Artificial Intelligence (IICAI-05)*, Pune, India, 20–22 December.

Majkić, Z. (2007) 'Temporal Probabilistic logic programs: State and revision', *International Conference in Artificial Intelligence and Pattern Recognition (AIPR-07)*, Orlando, FL, USA, 9–12 July.

Majkić, Z. (2008) 'Intensional semantics for RDF data structures', *12th International Database Engineering* & *Applications Systems (IDEAS08)*, Coimbra, Portugal, 10–13 September.

Majkić, Z. (2009) 'Intensional first-order logic for P2P database systems', *Journal of Data Semantics (JoDS XII)*, LNCS 5480, Springer-Verlag Berlin Heidelberg, pp.131–152.

Majkić, Z. (2009) 'Probabilistic deduction and pattern recognition of complex events', *International Conference on Information Security and Privacy (ISP-09)*, Orlando FL, USA, 13–16 July.

Majkić, Z. (2011) *First-order Logic: Modality and Intensionality*, arXiv: 1103.0680v1 [cs.LO], 3 March, pp.1–33.

Majkić, Z. (2012) 'Conservative intensional extension of Tarski's semantics', *Advances in Artificial Intelligence*, Hindawi Publishing Corporation, ISSN: 16870-7470, 23 October, pp.1–17.

Majkić, Z., Udrea, O. and Subrahmanian, V.S. (2007) 'Aggregates in generalized temporally indeterminate databases', *Int. Conference on Scalable Uncertainty Management (SUM 2007), in LNCS 4772*, Washington DC, USA, 10–12 October, pp.171–186.

Montague, R. (1970) 'Universal grammar', *Theoria*, Vol. 36, pp.373–398.

Montague, R. (1973) 'The proper treatment of quantification in ordinary English', in Hintikka, J. et al. (Eds.):*Approaches to Natural Language*, pp.221–242, Reidel, Dordrecht.

Montague, R. (1974) *Formal Philosophy*, selected papers of Richard Montague, in Thomason, R. (Ed.), pp.108–221, Yale University Press, New Haven, London.

Nilsson, N.J. (1986) 'Probabilistic logic', *Artif. Intelligence*, Vol. 28, No. 1, pp.71–87.

Ng, R.T. and Subrahmanian, V.S. (1992) 'Probabilistic logic programming', *Information and Computation*, Vol. 101, No. 2, pp.150–201.

Raedt, L. and Kimmig, A. (2015) 'Probabilistic (logic) programming concepts', in *Machine Learning*, Vol. 100, No. 1, pp.5–47.

Stalnaker, R. (1984) *Inquiry*, MIT Press, Cambridge, MA.

Udrea, O., Subrahmanian, V.S. and Majkić, Z. (2006) 'Probabilistic RDF', *IEEE Conference on Information Reuse and Integration (IEEE IRI 2006)*, 16–18 September, Waikoloa, Hawaii, USA.