The Distribution Semantics Is Well-Defined for All Normal Programs

Fabrizio Riguzzi

Dipartimento di Matematica e Informatica, Università di Ferrara
Via Saragat 1, I-44122, Ferrara, Italy
fabrizio.riguzzi@unife.it

Abstract. The distribution semantics is an approach for integrating logic programming and probability theory that underlies many languages and has been successfully applied in many domains. When the program has function symbols, the semantics was defined for special cases: either the program has to be definite or the queries must have a finite number of finite explanations. In this paper we show that it is possible to define the semantics for all programs.

Keywords: Distribution Semantics, Function Symbols, ProbLog, Probabilistic Logic Programming

1 Introduction

The distribution semantics [1, 2] was successfully applied in many domains and underlies many languages that combine logic programming with probability theory such as Probabilistic Horn Abduction, Independent Choice Logic, PRISM, Logic Programs with Annotated Disjunctions and ProbLog.

The definition of the distribution semantics can be given quite simply in the case of no function symbols in the program: a probabilistic logic program under the distribution semantics defines a probability distribution over normal logic programs called *worlds* and the probability of a ground query can be obtained by marginalizing the joint distribution of the worlds and the query. In the case the program has function symbols, however, this simple definition does not work as the probability of individual worlds is zero.

A definition of the distribution semantics for programs with function symbols was proposed in [1, 3] but restricted to definite programs. The case of normal programs was taken into account in [4] where the semantics required that the programs are acyclic. A looser condition was proposed in [5] but still required each goal to have a finite set of finite explanations.

In this paper we show that the distribution semantics can be defined for all programs, thus also for programs that have goals with an infinite number of possibly infinite explanations. We do so by adapting the definition of the well-founded semantics in terms of iterated fixpoints of [6] to the case of ProbLog, similarly to the way in which the T_P operator has been adapted in [7] to the case of stratified ProbLog programs using parameterized interpretations. In the case of infinite number of infinite explanations, we show that the probability of queries is defined in the limit and the limit always exists.

We consider the case of ProbLog but the results are equally applicable to all other languages under the distribution semantics, as there are linear transformations from one language to another that preserve the semantics.

The paper is organized as follows. Section 2 presents preliminary material on fixpoints and the well-founded semantics. Section 3 introduces the distribution semantics for programs without function symbols. Section 4 discusses the definition of the distribution semantics with function symbols in the case of finite set of finite explanations. Section 5 represents the main contribution of this paper and discusses the case of infinite set of infinite explanations. Finally, Section 6 concludes the paper. The proofs of the main results are reported in the Appendix.

2 Preliminaries

A relation on a set S is a *partial order* if it is reflexive, antisymmetric and transitive. In the following, let S be a set with a partial order $\leq$. $a \in S$ is an *upper bound* of a subset X of S if $x \leq a$ for all $x \in X$. Similarly, $b \in S$ is a *lower bound* of X if $b \leq x$ for all $x \in X$.

$a \in S$ is the *least upper bound* of a subset X of X if a is an upper bound of X and, for all upper bounds a' of X, we have $a \leq a'$. Similarly, $b \in S$ is the *greatest lower bound* of a subset X of S if b is a lower bound of X and, for all lower bounds b' of X, we have $b' \leq b$. The least upper bound of X is unique, if it exists, and is denoted by $lub(X)$. Similarly, the greatest lower bound of X is unique, if it exists, and is denoted by $glb(X)$.

A partially ordered set L is a *complete lattice* if $lub(X)$ and $glb(X)$ exist for every subset X of L. We let $\top$ denote the top element $lub(L)$ and $\bot$ denote the bottom element $glb(L)$ of the complete lattice L.

Let L be a complete lattice and $T : L \to L$ be a mapping. We say T is *monotonic* if $T(x) \leq T(y)$, whenever $x \leq y$. We say $a \in L$ is the *least fixpoint* of T if a is a fixpoint (that is, $T(a) = a$) and for all fixpoints b of T we have $a \leq b$. Similarly, we define *greatest fixpoint*.

Let L be a complete lattice and $T : L \to L$ be monotonic. Then we define $T \uparrow 0 = \bot$; $T \uparrow \alpha = T(T \uparrow (\alpha - 1))$, if α is a successor ordinal; $T \uparrow \alpha = lub(\{T \uparrow \beta | \beta < \alpha\})$, if α is a limit ordinal; $T \downarrow 0 = \top$; $T \downarrow \alpha = T(T \downarrow (\alpha - 1))$, if α is a successor ordinal; $T \downarrow \alpha = glb(\{T \downarrow \beta | \beta < \alpha\})$, if α is a limit ordinal.

Proposition 1. *Let L be a complete lattice and $T : L \to L$ be monotonic. Then T has a lest fixpoint, $lfp(T)$ and a greatest fixpoint $gfp(T)$.*

A normal program P is a set of normal rules. A normal rule has the form

$$r = h \leftarrow b_1, \ldots, b_n, not\, c_1, \ldots, not\, c_m \tag{1}$$

where $h, b_1, \ldots, b_n, c_1, \ldots, c_m$ are atoms.

The set of ground atoms that can be built with the symbols of a program P is called the Herbrand base and is denoted as $\mathcal{B}_P$.

A *two-valued interpretation* I is a subset of $\mathcal{B}_P$. I is the set of true atoms. The set $Int2$ of two-valued interpretations for a program P forms a complete lattice where the partial order $\leq$ is given by the subset relation $\subseteq$. The least upper bound and greatest lower bound are defined as $lub(X) = \bigcup_{I \in X} I$ and $glb(X) = \bigcap_{I \in X} I$. The bottom and top element are respectively $\emptyset$ and $\mathcal{B}_P$.

A *three-valued interpretation* $\mathcal{I}$ is a pair $\langle I_T; I_F \rangle$ where I_T and I_F are subsets of $\mathcal{B}_P$ and represent respectively the set of true and false atoms. The union of two three-valued interpretations $\langle I_T, I_F \rangle$ and $\langle J_T, J_F \rangle$ is defined as $\langle I_T, I_F \rangle \cup \langle J_T, J_F \rangle = \langle I_T \cup J_T, I_F \cup J_F \rangle$. The intersection of two three-valued interpretations $\langle I_T, I_F \rangle$ and $\langle J_T, J_F \rangle$ is defined as $\langle I_T, I_F \rangle \cap \langle J_T, J_F \rangle = \langle I_T \cap J_T, I_F \cap J_F \rangle$.

The set $Int3$ of three-valued interpretations for a program P forms a complete lattice where the partial order $\leq$ is defined as $\langle I_T, I_F \rangle \leq \langle J_T, J_F \rangle$ if $I_T \subseteq J_T$ and $I_F \subseteq J_F$. The least upper bound and greatest lower bound are defined as $lub(X) = \bigcup_{I \in X} I$ and $glb(X) = \bigcap_{I \in X} I$. The bottom and top element are respectively $\langle \emptyset, \emptyset \rangle$ and $\langle \mathcal{B}_P, \mathcal{B}_P \rangle$.

The well-founded semantics (WFS) assigns a three-valued model to a program, i.e., it identifies a three-valued interpretation as the meaning of the program. The WFS was given in [8] in terms of the least fixpoint of an operator that is composed by two sub-operators, one computing consequences and the other computing unfounded sets. We give here the alternative definition of the WFS of [6] that is based on a different iterated fixpoint.

Definition 1. *For a normal program P, sets Tr and Fa of ground atoms, and a 3-valued interpretation $\mathcal{I}$ we define*

$OpTrue^P_{\mathcal{I}}(Tr) = \{a | a$ *is not true in* $\mathcal{I}$*; and there is a clause* $b \leftarrow l_1, ..., l_n$ *in* P*, a grounding substitution* θ *such that* $a = b\theta$ *and for every* $1 \leq i \leq n$ *either* $l_i\theta$ *is true in* $\mathcal{I}$*, or* $l_i\theta \in Tr\}$*;*

$OpFalse^P_{\mathcal{I}}(Fa) = \{a | a$ *is not false in* $\mathcal{I}$*; and for every clause* $b \leftarrow l_1, ..., l_n$ *in* P *and grounding substitution* θ *such that* $a = b\theta$ *there is some* i $(1 \leq i \leq n)$ *such that* $l_i\theta$ *is false in* $\mathcal{I}$ *or* $l_i\theta \in Fa\}$.

In words, the operator $OpTrue^P_{\mathcal{I}}$ extends the interpretation $\mathcal{I}$ to add the new atomic facts that can be derived from P knowing $\mathcal{I}$, while $OpFalse^P_{\mathcal{I}}$ adds the new negations of atomic facts that can be shown false in P by knowing $\mathcal{I}$. $OpTrue^P_{\mathcal{I}}$ and $OpFalse^P_{\mathcal{I}}$ are both monotonic [6], so they both have a least and greatest fixpoints. An iterated fixpoint operator builds up *dynamic strata* by constructing successive three-valued interpretations as follows.

Definition 2 (Iterated Fixed Point). *For a normal program P, let $IFP^P : Int3 \rightarrow Int3$ be defined as $IFP^P(\mathcal{I}) = \mathcal{I} \cup \langle lfp(OpTrue^P_{\mathcal{I}}), gfp(OpFalse^P_{\mathcal{I}}) \rangle$.*

IFP^P is monotonic [6] and thus as a least fixed point $lfp(IFP^P)$. Moreover, the well-founded model $WFM(P)$ of P is in fact $lfp(IFP^P)$. Let δ be the smallest ordinal such that $WFM(P) = IFP^P \uparrow \delta$. We refer to δ as the *depth* of P. The *stratum* of atom a is the least ordinal β such that $a \in IFP^P \uparrow \beta$ (where a may be either in the true or false component of $IFP^P \uparrow \beta$). Undefined atoms of the well-founded model do not belong to any stratum – i.e. they are not added to $IFP^P \uparrow \delta$ for any ordinal δ.

3 The Distribution Semantics for Programs without Function Symbols

We present the distribution semantics for the case of ProbLog [9] as it is the language with the simplest syntax. A ProbLog program $\mathcal{P}$ is composed by a set of normal rules $\mathcal{R}$ and a set $\mathcal{F}$ of probabilistic facts. Each *probabilistic fact* is of the form $p_i :: a_i$ where $p_i \in [0, 1]$ and a_i is an atom, meaning that each ground instantiation $a_i\theta$ of a_i is true with probability p_i and false with probability $1 - p_i$. Each world is obtained by selecting or rejecting each grounding of all probabilistic facts.

An *atomic choice* is the selection or not of grounding $F\theta$ of a probabilistic fact F. It is represented with the triple (F, θ, i) where $i \in \{0, 1\}$. A set κ of atomic choices is *consistent* if it does not contain two atomic choices (F, θ, i) and (F, θ, j) with $i \neq j$ (only one alternative is selected for a ground probabilistic fact). The function $consistent(\kappa)$ returns true if κ is consistent. A *composite choice* κ is a consistent set of atomic choices. The probability of composite choice κ is $P(\kappa) = \prod_{(F_i,\theta,1)\in\kappa} p_i \prod_{(F_i,\theta,0)\in\kappa} 1 - p_i$ where p_i is the probability of the i-th probabilistic fact F_i. A *selection* σ is a total composite choice, i.e., it contains one atomic choice for every grounding of each probabilistic fact. A *world* w_σ is a logic program that is identified by a selection σ. The world w_σ is formed by including the atom corresponding to each atomic choice $(F, \theta, 1)$ of σ.

The probability of a world w_σ is $P(w_\sigma) = P(\sigma)$. Since in this section we are assuming programs without function symbols, the set of groundings of each probabilistic fact is finite, and so is the set of worlds $W_\mathcal{P}$. Accordingly, for a ProbLog program $\mathcal{P}$, $W_\mathcal{P} = \{w_1, \ldots, w_m\}$. Moreover, $P(w)$ is a distribution over worlds: $\sum_{w\in W_\mathcal{P}} P(w) = 1$. We call *sound* a program for which every world has a two-valued well-founded model. We consider only sound programs, as the uncertainty should be handled by the choices rather than by the semantics of negation.

Let q be a query in the form of a ground atom. We define the conditional probability of q given a world w as: $P(q|w) = 1$ if q is true in w and 0 otherwise. Since the program is sound, q can be only true or false in a world. The probability of q can thus be computed by summing out the worlds from the joint distribution of the query and the worlds: $P(q) = \sum_w P(q, w) = \sum_w P(q|w)P(w) = \sum_{w\models q} P(w)$.

4 The Distribution Semantics for Programs with Function Symbols

When a program contains functions symbols there is the possibility that its grounding may be infinite. If so, the number of atomic choices in a selection that defines a world is countably infinite and there is an uncountably infinite number of worlds. In this case, the probability of each individual world is zero since it is the product of infinite numbers all smaller than one. So the semantics of Section 3 is not well-defined.

Example 1. Consider the program

$$\begin{array}{lll} p(0) \leftarrow u(0). & t \leftarrow \neg s. & F_1 = a :: u(X). \\ p(s(X)) \leftarrow p(X), u(X). & s \leftarrow r, \neg q. & F_2 = b :: r. \\ & q \leftarrow u(X). & \end{array}$$

The set of worlds is infinite and uncountable. In fact, each world can be put in a one to one relation with a selection and a selection can be represented as a countable sequence of atomic choices of which the first involves fact F_2, the second $F_1/\{X/0\}$, the third $F_1/\{X/s(0)\}$ and so on. The set of selections can be shown uncountable by Cantor's diagonal argument. Suppose the set of selections is countable. Then the selections could be listed in order, suppose from top to bottom. Suppose the atomic choices of each selection are listed from left to right. We can pick a composite choice that differs from the first selection in the first atomic choice (if $(F_2, \emptyset, k)$ is the first atomic choice of the first selection, pick $(F_2, \emptyset, 1-k)$), from the second selection in the second atomic choice (similar to the case of the first atomic choice) and so on. In this way we have obtained a selection that is not present in the list because it differs from each selection in the list for at least an atomic choice. So it is not possible to list the selections in order.

We now present the definition of the distribution semantics for programs with function symbols following [4]. The semantics for a probabilistic logic program $\mathcal{P}$ with function symbols is given by defining a *probability measure* μ over the set of worlds $W_\mathcal{P}$. Informally, μ assigns a probability to a set of *subsets* of $W_\mathcal{P}$, rather than to every element of (the infinite set) $W_\mathcal{P}$. The approach dates back to [10] who defined a probability measure μ as a real-valued function whose domain is a σ*-algebra* Ω on a set $\mathcal{W}$ called the *sample space*. Together $\langle \mathcal{W}, \Omega, \mu \rangle$ is called a *probability space*.

Definition 3. *[11, Section 3.1] The set Ω of subsets of $\mathcal{W}$ is a σ-*algebra *on the set $\mathcal{W}$ iff (σ-1) $\mathcal{W} \in \Omega$; (σ-2) Ω is closed under complementation, i.e., $\omega \in \Omega \to (\mathcal{W} \setminus \omega) \in \Omega$; and ($\sigma$-3) Ω is closed under countable union, i.e., if $\omega_i \in \Omega$ for $i = 1, 2, \ldots$ then $\bigcup_i \omega_i \in \Omega$.*

The elements of Ω are called *measurable sets*. Importantly, for defining the distribution semantics for programs with function symbols, not every subset of $\mathcal{W}$ need be present in Ω.

Definition 4. *[10] Given a sample space $\mathcal{W}$ and a σ-algebra Ω of subsets of $\mathcal{W}$, a* probability measure *is a function $\mu : \Omega \to \mathbb{R}$ that satisfies the following axioms: (μ-1) $\mu(\omega) \geq 0$ for all $\omega \in \Omega$; (μ-2) $\mu(\mathcal{W}) = 1$; (μ-3) μ is countably additive, i.e., if $O = \{\omega_1, \omega_2, \ldots\} \subseteq \Omega$ is a countable collection of pairwise disjoint sets, then $\mu(\bigcup_{\omega \in O}) = \sum_i \mu(\omega_i)$.*

We first consider the finite additivity version of probability spaces. In this stronger version, the σ-algebra is replaced by an algebra.

Definition 5. *[11, Section 3.1] The set Ω of subsets of $\mathcal{W}$ is an* algebra *on the set $\mathcal{W}$ iff it respects conditions (σ-1), (σ-2) and condition (a-3): Ω is closed under finite union, i.e., $\omega_1 \in \Omega, \omega_2 \in \Omega \rightarrow (\omega_1 \cup \omega_2) \in \Omega$*

The probability measure is replaced by a finitely additive probability measure.

Definition 6. *Given a sample space $\mathcal{W}$ and an algebra Ω of subsets of $\mathcal{W}$, a* finitely additive probability measure *is a function $\mu : \Omega \rightarrow \mathbb{R}$ that satisfies axioms (μ-1) and (μ-2) of Definition 4 and axiom (m-3): μ is finitely additive, i.e., $\omega_1 \cap \omega_2 = \emptyset \rightarrow \mu(\omega_1 \cup \omega_2) = \mu(\omega_1) + \mu(\omega_2)$ for all $\omega_1, \omega_2 \in \Omega$.*

Towards defining a suitable algebra given a probabilistic logic program $\mathcal{P}$, we define the *set of worlds ω_κ compatible with* a composite choice κ as $\omega_\kappa = \{w_\sigma \in W_\mathcal{P} | \kappa \subseteq \sigma\}$. Thus a composite choice identifies a set of worlds. For programs without function symbols $P(\kappa) = \sum_{w \in \omega_\kappa} P(w)$.

Given a *set* of composite choices K, the *set of worlds ω_K compatible with K* is $\omega_\mathcal{K} = \bigcup_{\kappa \in K} \omega_\kappa$. Two composite choices κ_1 and κ_2 are *incompatible* if their union is not consistent. A set K of composite choices is *pairwise incompatible* if for all $\kappa_1 \in K, \kappa_2 \in K$, $\kappa_1 \neq \kappa_2$ implies that κ_1 and κ_2 are incompatible.

Regardless of whether a probabilistic logic program has a finite number of worlds or not, obtaining pairwise incompatible sets of composite choices is an important problem. This is because the *probability of a pairwise incompatible set K of composite choices* is defined as $P(K) = \sum_{\kappa \in K} P(\kappa)$ which is easily computed. Two sets K_1 and K_2 of finite composite choices are *equivalent* if they correspond to the same set of worlds: $\omega_{K_1} = \omega_{K_2}$.

One way to assign probabilities to a set K of composite choices is to construct an equivalent set that is pairwise incompatible; such a set can be constructed through the technique of *splitting*. More specifically, if $F\theta$ is an instantiated fact and κ is a composite choice that does not contain an atomic choice (F, θ, k) for any k, the *split* of κ on $F\theta$ is the set of composite choices $S_{\kappa,F\theta} = \{\kappa \cup \{(F, \theta, 0)\}, \kappa \cup \{(F, \theta, 1)\}\}$. It is easy to see that κ and $S_{\kappa,F\theta}$ identify the same set of possible worlds, i.e., that $\omega_\kappa = \omega_{S_{\kappa,F\theta}}$, and that $S_{\kappa,F\theta}$ is pairwise incompatible. The technique of splitting composite choices on formulas is used for the following result [12].

Theorem 1 *(Existence of a pairwise incompatible set of composite choices [12]) Given a finite set K of composite choices, there exists a finite set K' of pairwise incompatible composite choices such that K and K' are equivalent.*

Proof: Given a finite set of composite choices K, there are two possibilities to form a new set K' of composite choices so that K and K' are equivalent:

1. **removing dominated elements**: if $\kappa_1, \kappa_2 \in K$ and $\kappa_1 \subset \kappa_2$, let $K' = K \setminus \{\kappa_2\}$.
2. **splitting elements**: if $\kappa_1, \kappa_2 \in K$ are compatible (and neither is a superset of the other), there is a $(F, \theta, k) \in \kappa_1 \setminus \kappa_2$. We replace κ_2 by the split of κ_2 on $F\theta$. Let $K' = K \setminus \{\kappa_2\} \cup S_{\kappa_2,F\theta}$.

In both cases $\omega_K = \omega_{K'}$. If we repeat this two operations until neither is applicable we obtain a splitting algorithm that terminates because K is a finite set of composite choices. The resulting set K' is pairwise incompatible and is equivalent to the original set. $\diamondsuit$

Theorem 2 *(Equivalence of the probability of two equivalent pairwise incompatible finite set of finite composite choices [13]) If K_1 and K_2 are both pairwise incompatible finite sets of finite composite choices such that they are equivalent then $P(K_1) = P(K_2)$.*

For a probabilistic logic program $\mathcal{P}$, we can thus define a unique probability measure $\mu : \Omega_\mathcal{P} \rightarrow [0, 1]$ where $\Omega_\mathcal{P}$ is defined as the set of sets of worlds identified by finite sets of finite composite choices: $\Omega_\mathcal{P} = \{\omega_K | K$ is a finite set of finite composite choices $\}$. $\Omega_\mathcal{P}$ is an algebra over $W_\mathcal{P}$ since $W_\mathcal{P} = \omega_K$ with $K = \{\emptyset\}$. Moreover, the complement ω^c_K of ω_K where K is a finite set of finite composite choice is $\omega_{\overline{K}}$ where $\overline{K}$ is a finite set of finite composite choices. In fact, $\overline{K}$ can obtained with the function $duals(K)$ of [12] that performs Reiter's hitting set algorithm over K, generating an element κ of $\overline{K}$ by picking an atomic choice (F, θ, k) from each element of K and inserting in κ the atomic choice $(F, \theta, 1-k)$. After this process is performed in all possible ways, inconsistent sets of atom choices are removed obtaining $\overline{K}$. Since the possible choices of the atomic choices are finite, so is $\overline{K}$. Finally, condition (a-3) holds since the union of ω_{K_1} with ω_{K_2} is equal to $\omega_{K_1 \cup K_2}$ for the definition of ω_K.

The corresponding measure μ is defined by $\mu(\omega_K) = P(K')$ where K' is a pairwise incompatible set of composite choices equivalent to K. $\langle W_\mathcal{P}, \Omega_\mathcal{P}, \mu\rangle$ is a finitely additive probability space according to Definition 6 because $\mu(\omega_{\{\emptyset\}}) = 1$, $\mu(\omega_K) \geq 0$ for all K and if $\omega_{K_1} \cap \omega_{K_2} = \emptyset$ and K'_1 (K'_2) is pairwise incompatible and equivalent to K_1 (K_2), then $K'_1 \cup K'_2$ is pairwise incompatible and

$$\mu(\omega_{K_1} \cup \omega_{K_2}) = \sum_{\kappa \in K'_1 \cup K'_2} P(\kappa) = \sum_{\kappa_1 \in K'_1} P(\kappa_1) + \sum_{\kappa_2 \in K'_2} P(\kappa_2) = \mu(\omega_{K_1}) + \mu(\omega_{K_2}).$$

Given a query q, a composite choice κ is an *explanation* for q if $\forall w \in \omega_\kappa : \quad w \models q$. A set K of composite choices is *covering* wrt q if every world in which q is true belongs to ω_K

Definition 7. *For a probabilistic logic program $\mathcal{P}$, the probability of a ground atom q is given by $P(q) = \mu(\{w | w \in W_\mathcal{P}, w \models q\})$.*

If q has a finite set K of finite explanations such that K is covering then $\{w | w \in W_\mathcal{P} \wedge w \models q\} = \omega_K \in \Omega_T$ and we say that $P(q)$ is *finitely well-defined* for the distribution semantics. A program $\mathcal{P}$ is *finitely well-defined* if the probability of all ground atoms in the grounding of $\mathcal{P}$ is finitely well-defined.

Example 2. Consider the program of Example 1. The set $K = \{\kappa\}$ with $\kappa = \{(F_1, \{X/0\}, 1), (F_1, \{X/s(0)\}, 1)\}$ is a pairwise incompatible finite set of finite explanations that are covering for the query $p(s(0))$. Definition 7 therefore applies, and $P(p(s(0))) = P(\kappa) = a^2$

5 Infinite Covering Set of Explanations

In this section we go beyond [4] and we remove the requirement of the finiteness of the covering set of explanations and of each explanation for a query q.

Example 3. In Example 1, the query s has the pairwise incompatible covering set of explanations $K^s = \{\kappa_0^s, \kappa_1^s, \ldots\}$ with

$$\kappa_i^s = \{(F_2, \emptyset, 1), (F_1, \{X/0\}, 1), \ldots, (F_1, \{X/s^{i-1}(0)\}, 1), (F_1, \{X/s^i(0)\}, 0)\}$$

where $s^i(0)$ is the term where the functor s is applied i times to 0. So K^s is countable and infinite. A covering set of explanation for t is $K^t = \{\{(F_2, \emptyset, 0)\}, \kappa^t\}$ where κ^t is the infinite composite choice

$$\kappa^t = \{(F_2, \emptyset, 1), (F_1, \{X/0\}, 1), (F_1, \{X/s(0)\}, 1), \ldots\}$$

For a probabilistic logic program $\mathcal{P}$, we can define the probability measure $\mu : \Omega_\mathcal{P} \rightarrow [0, 1]$ where $\Omega_\mathcal{P}$ is defined as the set of sets of worlds identified by countable sets of countable composite choices: $\Omega_\mathcal{P} = \{\omega_K | K$ is a countable set of countable composite choices $\}$.

Lemma 3 *$\Omega_\mathcal{P}$ is a an σ-algebra over $W_\mathcal{P}$.*

Proof: (σ-1) is true as in the algebra case. To see that the complement ω_K^c of ω_K is in $\Omega_\mathcal{P}$, let us prove by induction that the dual $\overline{K}$ of K is a countable set of countable composite choices and then that $\omega_K^c = \omega_{\overline{K}}$. In the base case, if $K_1 = \{\kappa_1\}$, then we can obtain $\overline{K_1}$ by picking each atomic choice (F, θ, k) of κ_1 and inserting in $\overline{K_1}$ the composite choice $\{(F, \theta, 1 - k)\}$. As there is a finite or countable number of atomic choices in κ_1, $\overline{K_1}$ is a finite or countable set of composite choices each with one atomic choice.

In the inductive case, assume that $K_{n-1} = \{\kappa_1, \ldots, \kappa_{n-1}\}$ and that $\overline{K_{n-1}}$ is a finite or countable set of composite choices. Let $K_n = K_{n-1} \cup \{\kappa_n\}$ and $\overline{K_{n-1}} = \{\kappa'_1, \kappa'_2, \ldots\}$. We can obtain $\overline{K_n}$ by picking each κ'_i and each atomic choice (F, θ, k) of κ_n. If $(F, \theta, k) \in \kappa'_i$, then discard κ'_i, else if $(F, \theta, k') \in \kappa'_i$ with $k' \neq k$, insert κ'_i in $\overline{K_n}$. Otherwise generate the composite choice κ''_i where $\kappa''_i = \kappa'_i \cup \{(F, \theta, 1 - k)\}$ and insert it in $\overline{K_n}$. Doing this for all atomic choices (F, θ, k) in κ_n generates a finite set of composite choices if κ_n is finite and a countable number if κ_n is countable. Doing this for all κ'_i we obtain that $\overline{K_n}$ is a countable union of countable sets which is a countable set [14, page 3]. $\omega_K^c = \omega_{\overline{K}}$ because all composite choices of $\overline{K}$ are incompatible with each world of ω_K, as they are incompatible with each composite choice of K. So $\omega_K^c \in \Omega_\mathcal{P}$. ($\sigma$-3) is true as in the algebra case. $\diamondsuit$

We can see K as $\lim_{n\rightarrow\infty} K_n$ where $K_n = \{\kappa_1, \ldots, \kappa_n\}$. Each K_n is a finite set of composite choices and we can compute an equivalent finite pairwise incompatible set of composite choices K'_n. For each K'_n we can compute the probability $P(K'_n)$, noting that the probability of infinite composite choices is 0.

Now consider $\lim_{n\rightarrow\infty} P(K'_n)$. We can see the $P(K'_n)$s as the partial sums of a series. Moreover, it can be shown that $P(K'_{n-1}) \leq P(K'_n)$ so the series has non-negative terms. Such a series converges if the sequence of partial sums is bounded from above [15, page 92]. Since $P(K'_n)$ is bounded by 1, the limit $\lim_{n\to\infty} P(K'_n)$ exists. So we can define measure μ as $\mu(\omega_K) = \lim_{n\to\infty} P(K'_n)$.

Theorem 4 *$\langle W_{\mathcal{P}}, \Omega_{\mathcal{P}}, \mu\rangle$ is a probability space according to Definition 4.*

Proof: (μ-1) and (μ-2) hold as for the finite case and for (μ-3) let $O = \{\omega_{L_1}, \omega_{L_2}, \ldots\}$ be a countable set of subsets of $\Omega_{\mathcal{P}}$ such that the ω_{L_i}s are pairwise disjoint. Let L'_i be the pairwise incompatible set equivalent to L_i and let $\mathcal{L}$ be $\bigcup_{i=1}^{\infty} L'_i$. Since the ω_{L_i}s are pairwise disjoint, then $\mathcal{L}$ is pairwise incompatible. $\Omega_{\mathcal{P}}$ is a σ-algebra, so $\mathcal{L}$ is countable. Let $\mathcal{L}$ be $\{\kappa_1, \kappa_2, \ldots\}$ and let K'_n be $\{\kappa_1, \ldots, \kappa_n\}$. Then $\mu(O) = \lim_{n\to\infty} P(K'_n) = \lim_{n\to\infty} \sum_{\kappa\in K'_n} P(\kappa) = \sum_{\kappa\in\mathcal{L}} P(\kappa)$. Since $\mathcal{L} = \bigcup_{i=1}^{\infty} L'_i$, by rearranging the terms in the last summation we get $\mu(O) = \sum_{\kappa\in\mathcal{L}} P(\kappa) = \sum_{n=1}^{\infty} P(L'_n) = \sum_{n=1}^{\infty} \mu(\omega_{L_n})$. $\diamondsuit$

For a probabilistic logic program $\mathcal{P}$, the probability of a ground atom q is again given by $P(q) = \mu(\{w|w \in W_{\mathcal{P}}, w \models q\})$. If q has a countable set K of explanations such that K is covering then $\{w|w \in W_{\mathcal{P}} \wedge w \models q\} = \omega_K \in \Omega_{\mathcal{P}}$ and we say that $P(q)$ is *well-defined* for the distribution semantics. A program $\mathcal{P}$ is well-defined if the probability of all ground atoms in the grounding of $\mathcal{P}$ is well-defined.

Example 4. Consider Example 3. Since the explanations in K^s are pairwise incompatible the probability of s can be computed as

$$P(s) = b(1-a) + ba(1-a) + ba^2(1-a) + \ldots = \frac{b(1-a)}{1-a} = b.$$

since the sum is a geometric series. K^t is also pairwise incompatible and $P(\kappa^t) = 0$ so $P(t) = 1 - b + 0 = 1 - b$ which is what we intuitively expect.

We now want to show that every program has countable set of countable explanations that is covering for each query. In the following, we consider only ground programs that however may be countably infinite, thus they can be the result of grounding a program with function symbols.

Given two sets of composite choices K_1 and K_2, define the conjunction $K_1 \otimes K_2$ of K_1 and K_2 as $K_1 \otimes K_2 = \{\kappa_1 \cup \kappa_2 | \kappa_1 \in K_1, \kappa_2 \in K_2, consistent(\kappa_1 \cup \kappa_2)\}$

Similarly to [7], we define parametrized interpretations and a $IFPC_{\mathcal{P}}$ operator. Differently from [7], here parametrized interpretations associate a set of composite choices instead of a Boolean formula to each atom.

Definition 8 (Parameterized two-valued interpretations). *A parameterized positive two-valued interpretation Tr of a ground probabilistic logic program $\mathcal{P}$ with and atoms $\mathcal{B}_{\mathcal{P}}$ is a set of pairs (a, K_a) with $a \in$ atoms and K_a a set of composite choices. A parameterized negative two-valued interpretation Fa of a ground probabilistic logic program P with atoms $\mathcal{B}_{\mathcal{P}}$ is a set of pairs $(a, K_{\neg a})$ with $a \in \mathcal{B}_{\mathcal{P}}$ and $K_{\neg a}$ a set of composite choices.*

Parametrized two-valued interpretations form a complete lattice where the partial order is defined as $I \leq J$ if $\forall (a, K_a) \in I, (a, L_a) \in J : \omega_{K_a} \subseteq \omega_{L_a}$. The least upper bound and greatest lower bound always exist and are $lub(X) = \{(a, \bigcup_{(a,K_a)\in I, I\in X} K_a)|a \in \mathcal{B}_\mathcal{P}\}$ and $glb(X) = \{(a, \bigotimes_{(a,K_a)\in I, I\in X} K_a)|a \in \mathcal{B}_\mathcal{P}\}$. The top element $\top$ is $\{(a, \{\emptyset\})|a \in \mathcal{B}_\mathcal{P}\}$ and the bottom element $\bot$ is $\{(a, \emptyset)|a \in \mathcal{B}_\mathcal{P}\}$.

Definition 9 (Parameterized three-valued interpretation). *A parameterized three-valued interpretation $\mathcal{I}$ of a ground probabilistic logic program $\mathcal{P}$ with atoms $\mathcal{B}_\mathcal{P}$ is a set of triples $(a, K_a, K_{\neg a})$ with $a \in \mathcal{B}_\mathcal{P}$ and K_a and $K_{\neg a}$ sets of composite choices.*

Parametrized three-valued interpretations form a complete lattice where the partial order is defined as $I \leq J$ if $\forall (a, K_a, K_{\neg a}) \in I, (a, L_a, L_{\neg a}) \in J : \omega_{K_a} \subseteq \omega_{L_a}$ and $\omega_{K_{\neg a}} \subseteq \omega_{L_{\neg a}}$. The least upper bound and greatest lower bound always exist and are $lub(X) = \{(a, \bigcup_{(a,K_a,K_{\neg a})\in I, I\in X} K_a, \bigcup_{(a,K_a,K_{\neg a})\in I, I\in X} K_{\neg a})|a \in \mathcal{B}_\mathcal{P}\}$ and $glb(X) = \{(a, \bigotimes_{(a,K_a,K_{\neg a})\in I, I\in X} K_a, \bigotimes_{(a,K_a,K_{\neg a})\in I, I\in X} K_{\neg a})|a \in \mathcal{B}_\mathcal{P}\}$. The top element $\top$ is $\{(a, \{\emptyset\}, \{\emptyset\})|a \in \mathcal{B}_\mathcal{P}\}$, the bottom element $\bot$ is $\{(a, \emptyset, \emptyset)|a \in \mathcal{B}_\mathcal{P}\}$.

Definition 10. *For a ground program $\mathcal{P}$, a two-valued parametrized positive interpretation Tr with pairs (a, L_a), a two-valued parametrized negative interpretation Fa with pairs $(a, M_{\neg a})$ and a three-valued parametrized interpretation $\mathcal{I}$ with triples $(a, K_a, K_{\neg a})$, we define $OpTrueC^\mathcal{P}_\mathcal{I}(Tr) = \{(a, L'_a)|a \in \mathcal{B}_\mathcal{P}\}$ where*

$$L'_a = \begin{cases} \{\{(a, \emptyset, 1)\}\} & \text{if } a \in \mathcal{F} \\ \bigcup_{a\leftarrow b_1,\dots,b_n,\neg c_1,\dots,c_m\in\mathcal{R}}((L_{b_1} \cup K_{b_1}) \otimes \dots & \\ \otimes (L_{b_n} \cup K_{b_n}) \otimes K_{\neg c_1} \otimes \dots \otimes K_{\neg c_m}) & \text{if } a \in \mathcal{B}_\mathcal{P} \setminus \mathcal{F} \end{cases}$$

$OpFalseC^\mathcal{P}_\mathcal{I}(Fa) = \{(a, M'_a)|a \in \mathcal{B}_\mathcal{P}\}$ where

$$M'_{\neg a} = \begin{cases} \{\{(a, \emptyset, 0)\}\} & \text{if } a \in \mathcal{F} \\ \bigotimes_{a\leftarrow b_1,\dots,b_n,\neg c_1,\dots,c_m\in\mathcal{R}}((M_{\neg b_1} \otimes K_{\neg b_1}) \cup \dots & \\ \cup (M_{\neg b_n} \otimes K_{\neg b_n}) \cup K_{c_1} \cup \dots \cup K_{c_m}) & \text{if } a \in \mathcal{B}_\mathcal{P} \setminus \mathcal{F} \end{cases}$$

Proposition 5 *$OpTrueC^\mathcal{P}_\mathcal{I}$ and $OpFalseC^\mathcal{P}_\mathcal{I}$ are monotonic.*

Since $OpTrueC^\mathcal{P}_\mathcal{I}$ and $OpFalseC^\mathcal{P}_\mathcal{I}$ are monotonic, they have a least fixpoint and a greatest fixpoint.

Definition 11 (Iterated Fixed Point). *For a ground program $\mathcal{P}$, let $IFPC^\mathcal{P}$ be defined as $IFPC^\mathcal{P}(\mathcal{I}) = \{(a, K_a, K_{\neg a})|(a, K_a) \in lfp(OpTrueC^\mathcal{P}_\mathcal{I}), (a, K_{\neg a}) \in lfp(OpFalseC^\mathcal{P}_\mathcal{I})\}$.*

Proposition 6 *$IFPC^\mathcal{P}$ is monotonic.*

So $IFPC^\mathcal{P}$ has a least fixpoint. Let $WFMC(\mathcal{P})$ denote $lfp(IFPC^\mathcal{P})$, and let δ the smallest ordinal such that $IFPC^\mathcal{P} \uparrow \delta = WFMC(\mathcal{P})$. We refer to δ as the *depth* of $\mathcal{P}$.

Theorem 7 *For a ground probabilistic logic program $\mathcal{P}$ with atoms $\mathcal{B}_\mathcal{P}$, let K_a^α and $K_{\neg a}^\alpha$ be the formulas associated with atom a in $IFPC^\mathcal{P} \uparrow \alpha$. For every atom a and total choice σ, there is an iteration α_0 such that for all $\alpha > \alpha_0$ we have:*

$$w_\sigma \in \omega_{K_a^\alpha} \leftrightarrow WFM(w_\sigma) \models a \quad w_\sigma \in \omega_{K_{\neg a}^\alpha} \leftrightarrow WFM(w_\sigma) \models \neg a$$

Theorem 8 *For a ground probabilistic logic program $\mathcal{P}$, let K_a^α and $K_{\neg a}^\alpha$ be the formulas associated with atom a in $IFPC^\mathcal{P} \uparrow \alpha$. For every atom a and every iteration α, K_a^α and $K_{\neg a}^\alpha$ are countable sets of countable composite choices.*

So every query for every program has a countable set of countable explanations that is covering and the probability measure is well defined. Moreover, since the program is sound, for all atoms a, $\omega_{K_a^\delta} = \omega^c_{K_{\neg a}^\delta}$ where δ is the depth of the program, as in each world a is either true or false.

5.1 Comparison with Sato and Kameya's Definition

Sato and Kameya [3] define the distribution semantics for definite programs. They build a probability measure on the sample space $W_\mathcal{P}$ from a collection of finite distributions. Let $\mathcal{F}$ be $\{F_1, F_2, \ldots\}$ and let X_i be a random variable associated to F_i whose domain is $\{0, 1\}$.

The finite distributions $P_\mathcal{P}^{(n)}(X_1 = k_1, \ldots, X_n = k_n)$ for $n \geq 1$ must be such that

$$\begin{cases} 0 \leq P_\mathcal{P}^{(n)}(X_1 = k_1, \ldots, X_n = k_n) \leq 1 \\ \sum_{k_1,\ldots,k_n} P_\mathcal{P}^{(n)}(X_1 = k_1, \ldots, X_n = k_n) = 1 \\ \sum_{k_{n+1}} P_\mathcal{P}^{(n+1)}(X_1 = k_1, \ldots, X_{n+1} = k_{n+1}) = P_\mathcal{P}^{(n)}(X_1 = k_1, \ldots, X_n = k_n) \end{cases} \tag{2}$$

The last equation is called the compatibility condition. It can be proved [16] from the compatibility condition that there exists a probability space $(W_\mathcal{P}, \Psi_\mathcal{P}, \eta)$ where η is a probability measure on $\Psi_\mathcal{P}$, the minimal σ-algebra containing open sets of $W_\mathcal{P}$ such that for any n,

$$\eta(X_1 = k_1, \ldots, X_n = k_n) = P_T^{(n)}(X_1 = k_1, \ldots, X_n = k_n). \tag{3}$$

$P_\mathcal{P}^{(n)}(X_1 = k_1, \ldots, X_n = k_n)$ is defined as $P_\mathcal{P}^{(n)}(X_1 = k_1, \ldots, X_n = k_n) = p_1 \ldots p_n$ where p_i is the annotation of alternative k_i in fact F_i. This definition clearly satisfies the properties in (2). $P_\mathcal{P}^{(n)}(X_1 = k_1, \ldots, X_n = k_n)$ is then extended to a probability measure over $\mathcal{B}_\mathcal{P}$.

We conjecture that this definition of the distribution semantics with function symbols coincides for definite programs with the one given above.

To show that the two definition coincide, we conjecture that $\Psi_\mathcal{P} = \Omega_T$. Moreover, $X_1 = k_1, \ldots, X_n = k_n$ is equivalent to the set of composite choices $K = \{\{(F_1, \emptyset, k_1), \ldots, (F_n, \emptyset, k_n)\}\}$ and $\mu(\omega_K)$ gives $p_1 \ldots p_n$ which satisfies equation (3).

6 Conclusions

We have presented a definition of the distribution semantics in terms of an iterated fixpoint operator that allowed us to prove that the semantics is well defined for all programs. The operator we have presented is also interesting from an inference point of view, as it can be used for forward inference similarly to [7].

References

1. Poole, D.: Logic programming, abduction and probability - a top-down anytime algorithm for estimating prior and posterior probabilities. New Gen. Comp. **11** (1993) 377–400
2. Sato, T.: A statistical learning method for logic programs with distribution semantics. In: International Conference on Logic Programming. (1995) 715–729
3. Sato, T., Kameya, Y.: Parameter learning of logic programs for symbolic-statistical modeling. J. Artif. Intell. Res. **15** (2001) 391–454
4. Poole, D.: The Independent Choice Logic for modelling multiple agents under uncertainty. Artif. Intell. **94** (1997) 7–56
5. Riguzzi, F., Swift, T.: Terminating evaluation of logic programs with finite three-valued models. ACM T. Comput. Log. **15** (2014) 32:1–32:38
6. Przymusinski, T.: Every logic program has a natural stratification and an iterated least fixed point model. In: ACM Conference on Principles of Database Systems. (1989) 11–21
7. Vlasselaer, J., Van den Broeck, G., Kimmig, A., Meert, W., De Raedt, L.: Anytime inference in probabilistic logic programs with Tp-compilation. In: Internation Joint Conference on Artificial Intelligence. (2015) 1852–1858
8. Van Gelder, A., Ross, K.A., Schlipf, J.S.: The well-founded semantics for general logic programs. J. ACM **38** (1991) 620–650
9. De Raedt, L., Kimmig, A., Toivonen, H.: ProbLog: A probabilistic Prolog and its application in link discovery. In: Internation Joint Conference on Artificial Intelligence. (2007) 2462–2467
10. Kolmogorov, A.N.: Foundations of the Theory of Probability. Chelsea Publishing Company, New York (1950)
11. Srivastava, S.: A Course on Borel Sets. Graduate Texts in Mathematics. Springer (2013)
12. Poole, D.: Abducing through negation as failure: stable models within the independent choice logic. J. Logic Program. **44** (2000) 5–35
13. Poole, D.: Probabilistic Horn abduction and Bayesian networks. Artif. Intell. **64** (1993)
14. Cohn, P.: Basic Algebra: Groups, Rings, and Fields. Springer (2003)
15. Brannan, D.: A First Course in Mathematical Analysis. Cambridge University Press (2006)
16. Chow, Y., Teicher, H.: Probability Theory: Independence, Interchangeability, Martingales. Springer Texts in Statistics. Springer (2012)

A Proofs of Theorems

Proposition 5 *$OpTrueC_{\mathcal{I}}^{\mathcal{P}}$ and $OpFalseC_{\mathcal{I}}^{\mathcal{P}}$ are monotonic.*

Proof: Let us consider $OpTrueC_{\mathcal{I}}^{\mathcal{P}}$. We have to prove that if $Tr_1 \leq Tr_2$ then $OpTrueC_{\mathcal{I}}^{\mathcal{P}}(Tr_1) \leq OpTrueC_{\mathcal{I}}^{\mathcal{P}}(Tr_2)$. $Tr_1 \leq Tr_2$ means that $\forall (a, L_a) \in Tr_1, (a, M_a) \in Tr_2 : L_a \subseteq M_a$. Let (a, L'_a) be the elements of $OpTrueC_{\mathcal{I}}^{\mathcal{P}}(Tr_1)$ and (a, M'_a) the elements of $OpTrueC_{\mathcal{I}}^{\mathcal{P}}(Tr_2)$. We have to prove that $L'_a \subseteq M'_a$

If $a \in \mathcal{F}$ then $L'_a = M'_a = \{\{(a, \theta, 1)\}\}$. If $a \in \mathcal{B}_{\mathcal{P}} \setminus \mathcal{F}$, then L'_a and M'_a have the same structure. Since $\forall b \in \mathcal{B}_{\mathcal{P}} : L_b \subseteq M_b$, then $L'_a \subseteq M'_a$

We can prove similarly that $OpFalseC_{\mathcal{I}}^{\mathcal{P}}$ is monotonic. $\diamondsuit$

Proposition 6 *$IFPC^{\mathcal{P}}$ is monotonic.*

Proof: We have to prove that if $\mathcal{I}_1 \leq \mathcal{I}_2$ then $IFPC^{\mathcal{P}}(\mathcal{I}_1) \leq IFPC^{\mathcal{P}}(\mathcal{I}_2)$. $\mathcal{I}_1 \leq \mathcal{I}_2$ means that $\forall (a, L_a, L_{\neg a}) \in \mathcal{I}_1, (a, M_a, M_{\neg a}) \in \mathcal{I}_2 : L_a \subseteq M_a, L_{\neg a} \subseteq M_{\neg a}$. Let $(a, L'_a, L'_{\neg a})$ be the elements of $IFPC^{\mathcal{P}}(\mathcal{I}_1)$ and $(a, M'_a, M'_{\neg a})$ the elements of $IFPC^{\mathcal{P}}(\mathcal{I}_2)$. We have to prove that $L'_a \subseteq M'_a$ and $L'_{\neg a} \subseteq M'_{\neg a}$. This follows from the montonicity of $OpTrueC_{\mathcal{I}_1}^{\mathcal{P}}$ and $OpFalseC_{\mathcal{I}_2}^{\mathcal{P}}$ in $\mathcal{I}_1$ and $\mathcal{I}_2$ respectively, which can be proved as in Proposition 5. $\diamondsuit$

Lemma 9 *For a ground probabilistic logic program $\mathcal{P}$ with probabilistic facts $\mathcal{F}$, rules $\mathcal{R}$ and atoms $\mathcal{B}_{\mathcal{P}}$, let L_a^α be the formula associated with atom a in $OpTrueC_{\mathcal{I}}^{\mathcal{P}} \uparrow \alpha$. For every atom a, total choice σ and iteration α, we have:*

$$w_\sigma \in \omega_{L_a^\alpha} \rightarrow WFM(w_\sigma|\mathcal{I}) \models a$$

where $w_\sigma|\mathcal{I}$ is obtained by adding to w_σ the atoms a for which $(a, K_a, K_{\neg a}) \in \mathcal{I}$ and $w_\sigma \in K_a$ as facts and by removing all the rules with a in the head for which $(a, K_a, K_{\neg a}) \in \mathcal{I}$ and $w_\sigma \in K_{\neg a}$.

Proof: Let us prove the lemma by transfinite induction: let as assume the thesis for all $\beta < \alpha$ and let us prove it for α. If α is a successor ordinal, then it is easily verified for $a \in \mathcal{F}$. Otherwise assume $w_\sigma \in \omega_{L_a^\alpha}$ where

$$L_a^\alpha = \bigcup_{a \leftarrow b_1, \ldots, b_n, \neg c_1, \ldots, c_m \in \mathcal{R}} ((L_{b_1}^{\alpha-1} \cup K_{b_1}) \otimes \ldots \otimes (L_{b_n}^{\alpha-1} \cup K_{b_n}) \otimes K_{\neg c_1} \otimes \ldots \otimes K_{\neg c_m})$$

This means that there is rule $a \leftarrow b_1, \ldots, b_n, \neg c_1, \ldots, c_m \in \mathcal{R}$ such that $w_\sigma \in \omega_{L_{b_i}^{\alpha-1} \cup K_{b_i}}$ for $i = 1, \ldots, n$ and $w_\sigma \in \omega_{K_{\neg c_j}}$ for $j = 1 \ldots, m$. By the inductive assumption and because of how $w_\sigma|\mathcal{I}$ is built then $WFM(w_\sigma|\mathcal{I}) \models b_i$ and $WFM(w_\sigma|\mathcal{I}) \models \neg c_j$ so $WFM(w_\sigma|\mathcal{I}) \models a$.

If α is a limit ordinal, then

$$L_a^\alpha = lub(\{L_a^\beta | \beta < \alpha\}) = \bigcup_{\beta < \alpha} L_a^\beta$$

If $w_\sigma \in \omega_{L_a^\alpha}$ then there must exist a $\beta < \alpha$ such that $w_\sigma \in \omega_{L_a^\beta}$. By the inductive assumption the hypothesis holds. $\diamondsuit$

Lemma 10 *For a ground probabilistic logic program $\mathcal{P}$ with probabilistic facts $\mathcal{F}$, rules $\mathcal{R}$ and atoms $\mathcal{B}_\mathcal{P}$, let $M^\alpha_{\neg a}$ be the set of composite choices associated with atom a in $OpFalseC^\mathcal{P}_\mathcal{I} \downarrow \alpha$. For every atom a, total choice σ and iteration α, we have:*

$$w_\sigma \in \omega_{M_{\neg a}} \rightarrow WFM(w_\sigma|\mathcal{I}) \models \neg a$$

where $w_\sigma|\mathcal{I}$ is built as in Lemma 9.

Proof: Similar to the proof of Theorem Lemma 9. $\diamondsuit$

Lemma 11 *For a ground probabilistic logic program $\mathcal{P}$ with probabilistic facts $\mathcal{F}$, rules $\mathcal{R}$ and atoms $\mathcal{B}_\mathcal{P}$, let K^α_a and $K^\alpha_{\neg a}$ be the formulas associated with atom a in $IFPC^\mathcal{P} \uparrow \alpha$. For every atom a, total choice σ and iteration α, we have:*

$$w_\sigma \in \omega_{K^\alpha_a} \rightarrow WFM(w_\sigma) \models a \quad (4)$$

$$w_\sigma \in \omega_{K^\alpha_{\neg a}} \rightarrow WFM(w_\sigma) \models \neg a \quad (5)$$

Proof: Let us first prove that for all α, $WFM(w_\sigma) = WFM(w_\sigma|IFPC^\mathcal{P} \uparrow \alpha)$. We can prove it by transfinite induction. Consider the case of α a successor ordinal. Consider an atom b. If $w_\sigma \notin \omega_{K^\alpha_b}$ and $w_\sigma \notin \omega_{K^\alpha_{\neg b}}$ then the rules for b in w_σ and $w_\sigma|IFPC^\mathcal{P} \uparrow \alpha$ are the same. If $w_\sigma \in \omega_{K^\alpha_b}$ then b is a fact in $w_\sigma|IFPC^\mathcal{P} \uparrow \alpha$ but, according to Lemma 9, $WFM(w_\sigma|IFPC^\mathcal{P} \uparrow (\alpha-1)) \models b$. For the inductive hypothesis $WFM(w_\sigma) \models b$ so b has the same truth value in $WFM(w_\sigma)$ and $WFM(w_\sigma|IFPC^\mathcal{P} \uparrow \alpha)$. Similarly, if $w_\sigma \in \omega_{K^\alpha_{\neg b}}$, then $WFM(w_\sigma) \models \neg b$ and b has the same truth value in $WFM(w_\sigma)$ and $WFM(w_\sigma|IFPC^\mathcal{P} \uparrow \alpha)$. So overall $WFM(w_\sigma) = WFM(w_\sigma|IFPC^\mathcal{P} \uparrow \alpha)$.

If α is a limit ordinal, then $K^\alpha_b = \bigcup_{\beta<\alpha} K^\beta_b$ and $K^\alpha_{\neg b} = \bigcup_{\beta<\alpha} K^\beta_b$. So if $w_\sigma \in \omega_{K^\alpha_b}$ there is a β such $w_\sigma \in \omega_{K^\beta_b}$ and for the inductive hypothesis $WFM(w_\sigma) \models b$ so b has the same truth value in $WFM(w_\sigma)$ and $WFM(w_\sigma|IFPC^\mathcal{P} \uparrow \alpha)$. Similarly if $w_\sigma \in \omega_{K^\alpha_{\neg b}}$.

We can now prove the lemma by transfinite induction. Consider the case of α a successor ordinal. Since $(a, K^\alpha_a) \in \mathit{lfp}(OpTrueC^\mathcal{P}_{IFPC\uparrow(\alpha-1)})$, by Lemma 9

$$w_\sigma \in \omega_{K^\alpha_a} \rightarrow WFM(w_\sigma|IFPC^\mathcal{P} \uparrow (\alpha-1)) \models a$$

Since $WFM(w_\sigma|IFPC^\mathcal{P} \uparrow (\alpha-1)) = WFM(w_\sigma)$, (4) is proved.

Since $(a, K^\alpha_{\neg a}) \in \mathit{gfp}(OpFalseC^\mathcal{P}_{IFPC^\mathcal{P}\uparrow(\alpha-1)})$, by Lemma 10

$$w_\sigma \in \omega_{K^\alpha_{\neg a}} \rightarrow WFM(w_\sigma|IFPC^\mathcal{P} \uparrow (\alpha-1)) \models \neg a$$

Since $WFM(w_\sigma|IFPC^\mathcal{P} \uparrow (\alpha-1)) = WFM(w_\sigma)$, (5) is proved.

If α is a limit ordinal, $K^\alpha_a = \bigcup_{\beta<\alpha} K^\beta_a$ and $K^\alpha_{\neg a} = \bigcup_{\beta<\alpha} K^\beta_a$. If $w_\sigma \in \omega_{K^\alpha_a}$ there is a β such that $w_\sigma \in \omega_{K^\alpha_b}$ and by the inductive hypothesis (4) is proved. Similarly for (5). $\diamondsuit$

Lemma 12 *For a ground probabilistic logic program $\mathcal{P}$ with probabilistic facts $\mathcal{F}$, rules $\mathcal{R}$ and atoms $\mathcal{B}_{\mathcal{P}}$, let K_a^α and $K_{\neg a}^\alpha$ be the formulas associated with atom a in $IFPC^{\mathcal{P}} \uparrow \alpha$. For every atom a, total choice σ and iteration α, we have:*

$$a \in IFP^{w_\sigma} \uparrow \alpha \rightarrow w_\sigma \in K_a^\alpha$$

$$\neg a \in IFP^{w_\sigma} \uparrow \alpha \rightarrow w_\sigma \in K_{\neg a}^\alpha$$

Proof: Let us prove it by double transfinite induction. If α is a successor ordinal, assume that

$$a \in IFP^{w_\sigma} \uparrow (\alpha-1) \rightarrow w_\sigma \in K_a^{\alpha-1}$$

$$\neg a \in IFP^{w_\sigma} \uparrow (\alpha-1) \rightarrow w_\sigma \in K_{\neg a}^{\alpha-1}$$

Let us perform transfinite induction on the iterations of $OpTrue^{\mathcal{P}}_{IFPC^{\mathcal{P}}\uparrow(\alpha-1)}$. Let us consider a successor ordinal δ: assume that

$$a \in OpTrue^{w_\sigma}_{IFP^{w_\sigma}\uparrow(\alpha-1)} \uparrow (\delta-1) \rightarrow w_\sigma \in L_a^{\delta-1}$$

$$\neg a \in OpFalse^{w_\sigma}_{IFP^{w_\sigma}\uparrow(\alpha-1)} \downarrow (\delta-1) \rightarrow w_\sigma \in M_{\neg a}^{\delta-1}$$

and prove that

$$a \in OpTrue^{w_\sigma}_{IFP^{w_\sigma}\uparrow(\alpha-1)} \uparrow \delta \rightarrow w_\sigma \in L_a^{\delta}$$

$$\neg a \in OpFalse^{w_\sigma}_{IFP^{w_\sigma}\uparrow(\alpha-1)} \downarrow \delta \rightarrow w_\sigma \in M_{\neg a}^{\delta}$$

Consider a. If $a \in \mathcal{F}$ then it is easily proved.

For other atoms a, $a \in OpTrue^{w_\sigma}_{IFP^{w_\sigma}\uparrow(\alpha-1)} \uparrow \delta$ means that there is a rule $a \leftarrow b_1, \ldots, b_n, \neg c_1, \ldots, c_m$ such that for all $i = 1, \ldots, n$ $b_i \in OpTrue^{w_\sigma}_{IFP^{w_\sigma}\uparrow(\alpha-1)} \uparrow (\delta-1)$ and for all $j = 1, \ldots, m$ $\neg c_j \in IFP^{w_\sigma} \uparrow (\alpha-1)$. For the inductive hypothesis $\forall i : w_\sigma \in L_{b_i}^{\delta-1} \vee w_\sigma \in K_{b_i}^{\alpha-1}$ and $\forall j : w_\sigma \in K_{\neg c_j}^{\alpha-1}$ so, for the definition of $OpTrue^{w_\sigma}_{IFP^{w_\sigma}\uparrow(\alpha-1)}$, $w_\sigma \in L_a^\delta$. Analogously for $\neg a$.

If δ is a limit ordinal, then $L_a^\delta = \bigcup_{\mu<\delta} L_a^\mu$ and $M_{\neg a}^\delta = \bigotimes_{\mu<\delta} M_{\neg a}^\mu$. For the inductive hypothesis for all $\mu < \delta$

$$a \in OpTrue^{w_\sigma}_{IFP^{w_\sigma}\uparrow(\alpha-1)} \uparrow \mu \rightarrow w_\sigma \in L_a^{\mu}$$

$$\neg a \in OpFalse^{w_\sigma}_{IFP^{w_\sigma}\uparrow(\alpha-1)} \downarrow \mu \rightarrow w_\sigma \in M_{\neg a}^{\mu}$$

If $a \in OpTrue^{w_\sigma}_{IFP^{w_\sigma}\uparrow(\alpha-1)} \uparrow \delta$, then there exists a $\mu < \delta$ such that $a \in OpTrue^{w_\sigma}_{IFP^{w_\sigma}\uparrow(\alpha-1)} \uparrow \mu$. For the inductive hypothesis, $w_\sigma \in L_a^\delta$.

If $\neg a \in OpFalse^{w_\sigma}_{IFP^{w_\sigma}\uparrow(\alpha-1)} \downarrow \delta$, then, for all $\mu < \delta$, $\neg a \in OpFalse^{w_\sigma}_{IFP^{w_\sigma}\uparrow(\alpha-1)} \downarrow \mu$. For the inductive hypothesis, $w_\sigma \in M_a^\delta$.

Consider a limit α. Then $K_a^\alpha = \bigcup_{\beta<\alpha} K_a^\beta$ and $K_{\neg a}^\alpha = \bigcup_{\beta<\alpha} K_{\neg a}^\beta$. The inductive hypothesis is

$$a \in IFP^{w_\sigma} \uparrow \beta \rightarrow w_\sigma \in K_a^\beta$$

$$\neg a \in IFP^{w_\sigma} \uparrow \beta \rightarrow w_\sigma \in K_{\neg a}^\beta$$

If $a \in IFP^{w_\sigma} \uparrow \alpha$, then there exists a $\beta < \alpha$ such that $a \in IFP^{w_\sigma} \uparrow \beta$. For the inductive hypothesis $w_\sigma \in K_a^\beta$ so $w_\sigma \in K_a^\alpha$. Similarly for $\neg a$. $\diamondsuit$

Theorem 7 *For a ground probabilistic logic program $\mathcal{P}$ with atoms $\mathcal{B}_\mathcal{P}$, let K_a^α and $K_{\neg a}^\alpha$ be the formulas associated with atom a in $IFPC^\mathcal{P} \uparrow \alpha$. For every atom a and total choice σ, there is an iteration α_0 such that for all $\alpha > \alpha_0$ we have:*

$$w_\sigma \in \omega_{K_a^\alpha} \leftrightarrow WFM(w_\sigma) \models a \quad w_\sigma \in \omega_{K_{\neg a}^\alpha} \leftrightarrow WFM(w_\sigma) \models \neg a$$

Proof: The $\rightarrow$ direction is Lemma 11. In the other direction, $WFM(w_\sigma) \models a$ implies $\exists \alpha_0 \forall \alpha \geq \alpha_0 : IFP_{w_\sigma} \uparrow \alpha \models a$. For Lemma 12, $w_\sigma \in \omega_{K_a^\alpha}$. $WFM(w_\sigma) \models \neg a$ implies $\exists \alpha_0 \forall \alpha \geq \alpha_0 : IFP^{w_\sigma} \uparrow \alpha \models \neg a$. For Lemma 12, $w_\sigma \in \omega_{K_{\neg a}^\alpha}$. $\diamondsuit$

Theorem 8 *For a ground probabilistic logic program $\mathcal{P}$, let K_a^α and $K_{\neg a}^\alpha$ be the formulas associated with atom a in $IFPC^\mathcal{P} \uparrow \alpha$. For every atom a and every iteration α, K_a^α and $K_{\neg a}^\alpha$ are countable sets of countable composite choices.*

Proof: It can be proved by observing that each iteration of $OpTrueC^\mathcal{P}_{IFPC^\mathcal{P}\uparrow\beta}$ and $OpFalseC^\mathcal{P}_{IFPC^\mathcal{P}\uparrow\beta}$ generates countable sets of countable explanations since the set of rules is countable. $\diamondsuit$