INVESTIGATING AND EXTENDING P-LOG

by

Evgenii Balai, Dipl.-Ing

A Dissertation
in
COMPUTER SCIENCE

Submitted to the Graduate Faculty
of Texas Tech University in
Partial Fulfillment of
the Requirements for
the Degree of

DOCTOR OF PHILOSOPHY

Approved

Michael Gelfond
Chair of Committee

Yuanlin Zhang
Co-Chair of Committee

Nelson Rushton

Richard Watson

Mark Sheridan
Dean of the Graduate School

December 2017

ACKNOWLEDGMENTS

I would like to express my sincere appreciation and acknowledgment to my advisor, Dr. Michael Gelfond, for all the help, the encouragement and the patience shown by him at all the stages of working on this dissertation. His deep knowledge, experience and professionalism helped me greatly to continue and complete this work. I also thank the members of Michael Gelfond's family: Larisa Gelfond, Greg Gelfond, Patrick Kahl, and Yulia Kahl for their attention, support and active participation in my life here in Lubbock.

I am grateful to all the members of my committee for teaching me very important classes, collaborating with me on various research projects and contributing towards my development as a scientist, an engineer and a person.

I would especially like to thank the co-chair of this dissertation committee, Dr. Yuanlin Zhang. His hard work, persistence and dedication helped greatly in moving this work forward and were constant sources of motivation.

I am glad I had an opportunity to work with Dr. Nelson Rushton and Dr. Richard Watson, who made valuable suggestions to improve this dissertation and helped me to know and to better understand the life and the culture of such a great and an interesting country as the United States of America.

Finally, I am thankful to all the students I had a chance to work with and to teach classes for. It has been a very valuable experience.

Evgenii Balai

October 25, 2017

Department of Computer Science
Texas Tech University
Lubbock, TX, USA

TABLE OF CONTENTS

ABSTRACT

This dissertation focuses on the investigation and improvement of knowledge representation language P-log that allows for both logical and probabilistic reasoning. In particular, we extend P-log with new constructs to increase its expressive power and usability, clarify its semantics, define a new class of coherent (i.e., logically and probabilistically consistent) P-log programs and develop an inference algorithm for the programs from the new class. We also present the performance results of the preliminary implementation of the new algorithm. The results demonstrate that the new algorithm can substantially increase the performance of P-log inference on a number of important examples.

LIST OF TABLES

LIST OF FIGURES

CHAPTER I
INTRODUCTION

The language P-log, introduced in [Baral et al., 2004, Baral et al., 2009], is capable of combining non-monotonic logical reasoning about agents' beliefs in the style of Answer Set Prolog (ASP) [Gelfond & Lifschitz, 1991a] and probabilistic reasoning with Causal Bayesian Networks [Pearl, 2009]. The main goal of this dissertation is to improve P-log in the following directions.

1. Improve and expand the syntax and semantics of P-log by:
   - introducing additional means for describing sorts of a P-log program,
   - clarifying the original semantics of partial functions and activity records, and
   - allowing statements used to record activities of a reasoner to occur in the bodies of rules.
2. Define a new class of coherent (i.e, probabilistically and logically consistent) P-log programs which includes a number of classical examples not included in the classes defined in [Baral et al., 2009] and [Zhu, 2012].
3. Design and implement a reasonably efficient inference algorithm for the programs from the new class.

Sorts: In the original P-log sorts are described by statements of the form $s = \{t_1, \ldots, t_n\}$, where s is a sort name and $t_1, \ldots, t_n$ are ground terms[1], or by a program which has an unique answer set. This is inconvenient to describe large hierarchies of sorts which may be defined using string concatenations, conditions, set operators, etc. We extend the syntax for defining sorts by using the framework from [Balai et al., 2013]. Sorts there are defined by statements of the form:

[1]For the precise definition of ground term see Section 2.1.1

$$sort\_name = sort\_expression$$

where *sort_name* is a unique identifier preceded by the symbol # and *sort_expression* can be in one of the following forms:[2]

- $\{t_1, ..., t_n\}$
- $f(sort\_name_1, \ldots, sort\_name_n)$
- $sort\_name_1 \oplus \cdots \oplus sort\_name_n$

where $n > 0$, each t_i is a ground term, f is a function symbol, and $\oplus$ denotes a set operator (union, intersection, or difference) where the set operations result in a non-empty set. More details and examples can be found in [Balai et al., 2013].

Elimination of negative literals in the heads: One of the oversights made in the design of the original P-log was allowing negative literals of the form $f(\bar{x}) \neq y$ in a head of a rule. If f is a total function then the decision does not cause any problems. If, however, f is partial, it leads to a discrepancy between intuitive meaning of the program and its formal semantics. To see that let us consider a program P consisting of rules:

$f \ : \ boolean$

$f \neq false$

Intuitively, $f \neq false$ means that f is defined, i.e. has a value and this value is different from $false$. The intuition agrees with some other extensions of ASP with functional symbols, e.g. [Balduccini, 2012]. Together with declaration f : *boolean* this should imply that the value of f is $true$. However, the program P has one possible world consisting of a literal $f \neq false$ and hence P does not entail $f = true$. To remedy the problem one probably should define f as random which will force f to take a value. The new program will clearly entail $f = true$. But, if f is random, then $f \neq false$ should be replaced by a more appropriate $obs(f \neq false)$ which would

[2] the actual syntax is slightly different, we simplify it here to shorten the description

allow us to produce the same result. This observation suggests that allowing negative literals in the head is redundant. Moreover, disallowing this syntactic feature leads to a substantial simplification of the formal semantics of P-log. Instead of defining possible worlds as sets of literals we can view them simply as (partial) interpretations of the attribute terms from the program's signature (in other words, a collection of atoms). So far we were not able to find any adverse effect of our restriction on the original syntax.

P-log observations: Another problem with the original P-log is related to the intuitive meaning of P-log observations. According to [Baral et al., 2009] such observations are used "to record the outcomes of random events, i.e., random attributes, and attributes dependent on them". However, axioms 10 – 13 from the original paper do not faithfully reflect this intuition. Axiom (12), for instance, does not prohibit observations of non-random events. Instead it simply views $obs(f(\bar{x}) = y)$ as a shorthand for the constraint

$$\leftarrow \ \ not\ f(\bar{x}) = y$$

where f is an arbitrary attribute. The new observation simply eliminates some of the possible worlds of the program, which reflects understanding of observations in classical probability theory. This view is also compatible with the treatment of observations in action languages. So if we limit ourselves to the syntax of traditional P-log there are no adverse consequences of expanding the observability of attribute values to a non-random case. Later we will discuss some of its benefits.

P-log intervening actions: Let us now attempt to clarify the P-log meaning of the *do* statement. The original paper states: "the statement $do(f(\bar{x}) = y)$ indicates that $f(\bar{x}) = y$ is made true as a result of a deliberate (non-random) action". Note, that here, $f(\bar{x})$ is not required to be declared as random, i.e. its value does not have to be normally defined by a random experiment. This is not wrong. Even though the original intervening action *do* of Pearl only applies to random attributes (no other types are available in Bayesian Nets) nothing prohibits us from expanding the domain of *do* to non-random ones. After all this is exactly what we did with observations. But in the case of intervening actions, such an extension seems to be unwarranted. It is easy to see that for non-random $f(\bar{x})$, $do(f(\bar{x}) = y)$ is (modulo *do*) equivalent to $f(\bar{x}) = y$, which undermines the utility of such statements. In addition, it violates an important principle of language design frequently advocated by N. Wirth and others: *Whenever possible, make sure that each important type of informal statements you want expressible in your formal language corresponds to one language construct*. Moreover, applying *do* to interfere with a random experiment with a dynamic domain p causes an ambiguity of an interpretation: should the deliberately assigned value belong to the dynamic domain or arbitrary value of a proper sort must be allowed? The formal semantics from [Baral et al., 2009] corresponds to the second option, but, according to the best recollection of the authors of [Baral et al., 2009], this is accidental. The decision was not deliberate.

We may avoid this redundancy and ambiguity by slightly modifying the definition of the set R of general axioms of program Π with signature Σ.

Firstly, we keep the rules of the form

$$random(f(\bar{x}) : \{X : p(X)\}) \leftarrow body.$$

unchanged. Since the axioms need to satisfy ASP syntax, we identify the P-log atom $random(f(\bar{x}) : \{X : p(X)\})$ with a simpler atom $random(f(\bar{x}), p)$. After this new interpretation, $random(f(\bar{x}) : \{X : p(X)\})$ says that $f(\bar{x})$ may take the value from $\{X : p(X)\}$ as the result of a random experiment, whose outcome could possibly be manipulated. To separate a deliberate interference from a truly random assignment we introduce a new special attribute term $truly\_random$.

For every attribute term $random(f(\bar{x}), \{X : p(X)\})$ from Σ such that $range(f(\bar{x}))$ $= \{y_1, \dots, y_k\}$, R contains rules

$$f(\bar{x}) = y_1 \textbf{ or } \dots \textbf{ or } f(\bar{x}) = y_k \leftarrow random(f(\bar{x}) : \{X : p(X)\}) \tag{1.1}$$

$$\begin{aligned} truly\_random(f(\bar{x})) \quad \leftarrow \quad & random(f(\bar{x}) : \{X : p(X)\}), \\ & not\ do(f(\bar{x}), y_1), \ldots, not\ do(f(\bar{x}), y_k) \end{aligned} \tag{1.2}$$

$$\leftarrow f(\bar{x}) = Y,\ \textbf{not}\ p(Y), random(f(\bar{x}) : \{X : p(X)\}). \tag{1.3}$$

$$\begin{aligned} \leftarrow \quad & not\ f(\bar{X}) = Y, \\ & do(f(\bar{X}), Y). \end{aligned} \tag{1.4}$$

$$\begin{aligned} \leftarrow \quad & not\ random(f(\bar{X}) : \{X : p(X)\}), \\ & do(f(\bar{X}), Y). \end{aligned} \tag{1.5}$$

Intuitively, axioms (1.1) and (1.3) guarantee that if $random(f(\bar{x}) : \{X : p(X)\})$ is true, $f(\bar{x})$ is assigned a value satisfying condition p, axiom (1.2) guarantees that $truly\_random(f(\bar{x}))$ is true iff the value of $f(\bar{x})$ is assigned as the result of a genuine random experiment (that is, an experiment which was not interfered with), (1.4) guarantees that the atoms made true by interventions are indeed true, and (1.5) makes sure that an attempt to apply *do* to a non-random $f(\bar{X})$ leads to inconsistency.

Actions and observations in the bodies of rules: Another important modification of the language is allowing literals formed by *do* and *obs* (i.e, actions and observations) to occur in the bodies of P-log rules. We have already mentioned that the addition of an observation to a program in the original P-log language may only eliminate some of its possible worlds but cannot create a new one. Allowing observations to occur in the bodies of P-log rules changes the situation. Addition of an observation $obs(a, true)$ to a program

$$Q = \begin{cases} \neg a & \leftarrow \quad not\ a. \\ a & \leftarrow \quad obs(a, true). \end{cases}$$

creates a possible world which did not exist according to the original program. This extension of the language does not significantly complicate the mathematical semantics of the language but seems to add substantially to its expressive power.

To further illustrate this phenomena let us assume that we would like to use P-log to formalize knowledge relevant to the following problem.

Suppose that an experienced diagnostician was able to determine that a certain patient's symptom s has two possible causes c_1 and c_2. The purely qualitative information available to the diagnostician can be expressed in P-log by a program P_1:

$$P_1 = \begin{cases} \neg c_1 & \leftarrow & not\ obs(s) \\ \neg c_2 & \leftarrow & not\ obs(s) \\ s & \leftarrow & c_1 \\ s & \leftarrow & c_2 \\ \neg s & \leftarrow & not\ s \end{cases}$$

The first two rules say that the causes are not true in case of the absence of an observation of the symptom – a natural default we use in our actions before becoming aware of a problem by experiencing its symptoms. The next three rules give the complete list of possible causes for s. According to this program the probabilities of s, c_1 and c_2 are 0. It is important to note that an update of P_1 by observation of the truth of any attribute of P_1 leads to inconsistency. This is not necessarily an unwelcome outcome for the observations of causes – after all causes are normally not directly observable and need to be derived from the observations of symptoms and the background knowledge. This shall not however happen for the observation of the symptom s. The following informal argument is possible in this case: *Since we are given a complete list of possible causes of s and s is observed to be true we cannot continue to use closed world assumptions for causes. Instead we should think of them as random attributes which may or may not be true.* Accordingly, the program describing the agent's knowledge should have possible worlds $W_1 = \{c_1, \neg c_2, s\}$, $W_2 = \{\neg c_1, c_2, s\}$, and $W_3 = \{c_1, c_2, s\}$.

The missing knowledge used by the reasoner to go from observations of a symptom to its causes can be represented by expanding P_1 by the rules:

$$R = \begin{cases} random(c_1) \leftarrow obs(s) \\ random(c_2) \leftarrow obs(s) \end{cases}$$

which have observations in their bodies. It is easy to check that program

$$P_2 = P_1 \cup R \cup \{obs(s)\}$$

is consistent and has three possible worlds W_1, W_2, and W_3 described above. The program assigns probabilities 2/3 to c_1 and c_2 and probability 1 to s.

It may be tempting to replace P_2 by program P_2', obtained from P_2 by replacing R with collection of rules:

$$R' = \begin{cases} random(c_1) \leftarrow s \\ random(c_2) \leftarrow s \end{cases}$$

This, however, will not work, since the resulting program will be inconsistent. This is not surprising, since there is a substantial difference between s and $obs(s)$. The first is a fact and can be used by a reasoner to justify his belief in s. The second is a constraint which cannot be used for this purpose. As the result, allowing observations in the bodies of rules is essential for the type of reasoning discussed in this example.

Let us now assume that, by checking some available statistics, the diagnostician acquire knowledge about probabilities of c_1 and c_2. These probabilities can be added to the program by the set PA of causal probability atoms:

$$PA = \begin{cases} pr(c_1) = 0.05 \\ pr(c_2) = 0.01 \end{cases}$$

The probabilities assigned to c_1 and c_2 by the new program,

$$P_3 = P_2 \cup PA$$

are now approximately 0.8 and 0.2. So c_1 is the most likely cause of the symptom.

Finally, let us consider the case when after some direct or indirect observation the diagnostician establishes that c_2 is true. The probabilities assigned to c_1 and c_2 by program

$$P_4 = P_3 \cup \{obs(c_2)\}$$

are now 0.05 and 1 respectively. The latter observation is an example of a probabilistic phenomena called "explaining away" [Wellman & Henrion, 1993]: when you have competing possible causes for some event, and the chances of one of those causes increases, the chances of the other causes must decline since they are being "explained away" by the first explanation.

The example shows a fairly seamless combination of logical and probabilistic reasoning in search of causal explanations of a symptom. An author of the original P-log was not able to express this type of reasoning in the original P-log based on ASP. We could, however, do it in CR-Prolog [Balduccini & Gelfond, 2003] – extension of ASP by so called consistency-restoring rules. Consider program P_3' obtained from P_3 by replacing R with the rules:

$$R_2 = \begin{cases} random(c_1) \leftarrow include\_causes \\ random(c_2) \leftarrow include\_causes \end{cases}$$

adding a consistency-restoring rule:

$$include\_causes \stackrel{+}{\leftarrow}$$

and replacing each of the defaults

$$\neg c_1 \leftarrow not\ obs(s)$$

$$\neg c_2 \leftarrow not\ obs(s)$$

with classical closed worlds assumptions for c_1 and c_2:

$$\neg c_1 \leftarrow not\ c_1$$

$$\neg c_2 \leftarrow not\ c_2$$

It is easy to check that P'_3 is logically and probabilistically equivalent to P_3. Similarly, we can obtain a CR-Prolog program equivalent to P_4.

This, however, requires the programmer to learn the semantics of CR-Prolog. Moreover, currently there is no reasoning system that implements P-log with consistency-restoring rules, and the task of developing and efficiently implementing such a system seems to be non-trivial. In contrast, implementing P-log with rules containing actions and observations in their bodies seems to be less daunting.

There are other possible uses of observations in the body of rules. In our previous example we have already encountered unobservable attributes. In fact, in the original interpretation of *obs*, if $\bar{x}$ does not belong to the domain of f, i.e., no value is assigned to $f(\bar{x})$ by the program, then $f(\bar{x})$ is unobservable. Sometimes, however, $f(\bar{x})$ is undefined simply because the reasoner does not know the value of $f(\bar{x})$. In some of such cases this value can be obtained by a direct observations. (We refer to such $f(\bar{x})$ as *directly observable*.) In P-log this property can be expressed by the rule:

$$f(\bar{x}) = y \leftarrow obs(f(\bar{x}), y).$$

Note that, for a directly observable value of an attribute $f(\bar{x})$, expanding a program by $obs(f(\bar{x}), y)$ is (modulo atoms formed by *obs*) equivalent to expanding it by $f(\bar{x}) = y$.

Impossibility of observing $f(\bar{x})$ can be expressed as:

$$\leftarrow obs(f(\bar{x}), Y).$$

The ability to use observations in the bodies of rules allows for one more possible extension of P-log. We may relax the restrictions requiring our observations to be always accurate. To avoid the change of the existing meaning of the observations, the language can be extended by a new activity record of the form $imprecise\_measure(f(\bar{x}), y)$ which can be translated into:

$$random(f(\bar{x})) \leftarrow obs(f(\bar{x}), Y), imprecise\_measure(f(\bar{x}))$$

and a probabilistic information of the accuracy of the measurement.

New Subclass of Coherent Programs and Inference Algorithm. A naive inference in P-log requires the computation of all possible worlds of the program. An algorithm which uses the naive approach is described in Section 3.1 of [Zhu, 2012]. In order to compute the possible worlds, a P-log program Π is translated into an ASP program $\tau(\Pi)$. An ASP solver is used to obtain the answer sets of $\tau(\Pi)$, that are later mapped into the possible worlds of Π and their probabilities.

A more efficient algorithm for a class $\mathcal{S}$ of programs, called strongly causally ordered unitary (scou) programs, is defined in Sections 3.2 - 3.3 of [Zhu, 2012]. It is shown that, under certain conditions, the computation of all possible worlds can be avoided by computing partial interpretations each of which may correspond to several possible worlds. The probability of the query is computed, possibly using the pr-atoms of the program, from the collection of the computed partial interpretations whose size is typically smaller than the number of all the possible worlds.

Unfortunately, there are interesting and important programs which do not belong to $\mathcal{S}$ (for details, see Examples 1-3 from Section 3). Moreover, the description and implementation of the algorithm contain some ambiguities and typos which make it difficult to fully understand its properties.

In order to address these problems, in this dissertation we:

- introduce a new class, $\mathcal{B}$, of P-log programs containing all programs from $\mathcal{S}$ considered in [Zhu, 2012] and a number of useful programs not belonging to $\mathcal{S}$ and show the coherency of programs from this class,
- define and implement a new query answering algorithm for P-log, and
- show that the algorithm is sound and complete for programs from $\mathcal{B}$.

CHAPTER II
SYNTAX AND SEMANTICS OF P-LOG

A P-log program will be defined as a pair consisting of a sorted signature and a collection of P-log rules and causal probability atoms. The program will define the collection of possible worlds corresponding to beliefs of a rational reasoner associated with the programs as well as the probability function on the sets of these worlds describing degrees of the reasoner's beliefs.

2.1 Syntax of P-log

We start with an accurate definition of sorted signatures and their interpretations which will be used throughout the text and then define the syntax of P-log programs.

2.1.1 Sorted Signatures of P-log

A *sorted signature* Σ is a tuple $\langle S, O, F\rangle$ where S is a finite non-empty set of sort names, O is a finite set of object constants, and F is a finite non-empty set of functional symbols.

- Every sort name $s \in S$ is assigned the sort denoted by it - a collection of object constants from O. We say that an object o from this collection belongs to (or is an instance of) sort s and write $o \in s$. We also assume that Σ contains sorts $\mathbf{N}$ and $\mathbf{Q}$ of natural and rational numbers which are mapped into standard representations of these numbers viewed as elements of O. Whenever it is clear from the context, we will abuse the notation and use the same letter to refer to the sort name and the sort denoted by it.

- Every function symbol from F has sort names assigned to its parameters and its range. In what follows we use standard mathematical notation $f : s_1, \dots, s_n \rightarrow s$ to describe these assignments.

- Set F is partitioned into two parts: *attributes* and *arithmetic* functions $+$, $-$, etc. defined on natural or rational numbers.

A *ground term*, t, of Σ with *value belonging to sort* s (written as $t \in s$) is:

- a constant o such that $o \in s$,
- a string of the form $f(t_1, \ldots, t_n)$ where $f : s_1, \ldots, s_n \to s$ and $t_1, \ldots, t_n$ are ground terms with values from sorts $s_1, \ldots, s_n$. If f is an arithmetic function the term is called *arithmetic*[1]. Otherwise it is called an *attribute term*. The sort s is referred to as the range of $f(t_1, \ldots, t_n)$.

Note that the value of a ground term may belong to more than one sort. In the rest of this subsection we will use the word *term* to mean ground term.

Signatures of P-log program will always include special attribute terms listed below. (We use $f(\bar{x})$ to denote an attribute term and y to denote a variable or constant which can serve as the value of $f(\bar{x})$; p stands for a unary boolean attribute):

- $do(f(\bar{x}), y)$, which reads as "a random experiment assigning value to $f(\bar{x})$ is deliberately interfered with and $f(\bar{x})$ is assigned the value y",
- $obs(f(\bar{x}), y, true)$, which reads as "the value of $f(\bar{x})$ is observed to be y" and $obs(f(\bar{x}), y, false)$, which reads as "the value of $f(\bar{x})$ is observed to be different from y"[2],
- $random(f(\bar{x}), p)$, which says that "$f(\bar{x})$ may take the value from $\{X : p(X)\}$ as the result of a genuine or a deliberately interfered with random experiment", and
- $truly\_random(f(\bar{x}))$, which says that "$f(\bar{x})$ takes value as the result of a genuine random experiment (i.e., the one without any outside interference)".

[1] As usual for arithmetic terms we use infix notation.

[2] To simplify the notation we sometimes write $obs(f(\bar{x}), y, true)$ and $obs(f(\bar{x}), y, false)$ as $obs(f(\bar{x}), y)$ (or $obs(f(\bar{x}) = y)$) and $\neg obs(f(\bar{x}), y)$ (or $obs(f(\bar{x}) \neq y)$) respectively; if f is boolean then $obs(f(\bar{x}), true, true)$ will be written as $obs(f(\bar{x}))$.

The arguments of any of the special attribute terms cannot be formed by special attribute terms. For the sake of compatibility with original P-log, we will sometimes write $random(a : \{X : p(X)\})$ instead of $random(a, p)$.

Note that in the first case the value of $f(\bar{x})$ must belong to $\{Y : p(Y)\} \cap range(f)$; the argument p can be omitted, in which case the value of $f(\bar{x})$ is selected from the range of f.

Each of the special attribute terms has a boolean range.

An *atom* of Σ is a statement of one of the forms:

1. $t = y$ where t is an attribute term, y is an object constant such that $y \in range(t)$;

2. $t_1 \odot t_2$ where t_1 and t_2 are arithmetic terms and $\odot$ is one of the standard arithmetic relations, $=$, $\neq$, $>$, etc. These atoms are called *arithmetic*

The statement $t = y$ reads: "y is the value of t". Its negation, $\neg(t = y)$ or $t \neq y$ is read as "the value of t is different from y". If t is boolean then $t = true$ and $t = false$ will often be written as t and $\neg t$. An atom of the form $t = y$ is called *special* if t a special attribute term, otherwise it is called *regular*. If t is of the form $obs(f(\bar{x}), y, B)$ or $do(f(\bar{x}), y)$, the atom is called an *observation* or an *action* correspondingly.

Atoms and their negations are referred to as *positive literals* and *negative literals* of Σ correspondingly. A *literal* of Σ is either a positive literal of Σ or a negative literal of Σ.

A literal, possibly preceded by the default negation *not*, is called an *extended literal* or simply an *e-literal* of Σ. The e-literal **not** l reads as "l is not believed to be true" (which is, of course, different from "l is believed to be false").

Elements of a program such as terms and e-literals are called *ground* if they contain no variables and no names of arithmetic functions.

In what follows, by signature we will mean P-log signature. For a signature Σ, by $at(\Sigma)$, $lit(\Sigma)$, $e\text{-}lit(\Sigma)$, $attr(\Sigma)$ we will denote the sets of all ground atoms, ground literals, ground extended literals and attribute terms respectively.

2.1.2 P-log Programs

A *P-log rule* over signature Σ is of the form:

$$l \leftarrow body \tag{2.1}$$

where l is an atom of Σ, also referred to as the *head* of the rule, and *body* is a collection of e-literals of Σ, also referred to as the *body* of the rule. The head of the rule can optionally be omitted, in which case the rule is of the form

$$\leftarrow body \tag{2.2}$$

and is called *a constraint.*

For a rule r, by $head(r)$ and $body(r)$ we will denote the head of r and the body of r respectively.

If l is an observation or an action, we require the *body* to be empty and the rule is called *an activity record.*

If l is of the form $random(a : \{X : p(X)\})$, the rule (2.1) is called *a random selection rule.*

A rule which is not an activity record or a random selection rule is called *a regular rule.*

By a *P-log program* we mean a pair consisting of

1. A signature Σ and

2. A collection R of P-log rules and causal probability statements (also called *pr-atoms*) – expressions of the form

$$pr(f(\bar{x}) = y \mid B) = v \tag{2.3}$$

where $f(\bar{x})$ is a regular attribute term such that $y \in range(f(\bar{x}))$, B is a set of e-literals of Σ and $v \in [0, 1]$ is a rational number. The statement says that "if the value of $f(\bar{x})$ is generated randomly and B holds then the probability of the selection of y for the value of $f(\bar{x})$ is v. Moreover, there is a potential existence of a direct causal relationship between B and the possible value of $f(\bar{x})$."

We will refer to $f(\bar{x}) = y$ as the *head* of the pr-atom and to B as the *body* of the *pr*-atom. We will refer to v as the *probability assigned by the pr-atom*.

Unless otherwise stated, we will assume that R contains the following rules, also referred to as *general P-log axioms*:

- For every attribute term $f(\bar{x})$ of Σ which is not special, the rules:

$$\leftarrow not\ f(\bar{x}) = Y,\ obs(f(\bar{x}), Y, true). \tag{2.4}$$

$$\leftarrow not\ f(\bar{x}) \neq Y,\ obs(f(\bar{x}), Y, false). \tag{2.5}$$

  Intuitively, the rules (often referred to as *reality check axioms*) prohibit observations of undefined attribute terms as well as observations which contradict the agent's beliefs.

- For every random atom $random(f(\bar{x}) : \{X : p(X)\})$ of Σ such that $range(f) = \{y_1, \ldots, y_k\}$, the rules :

$$f(\bar{x}) = y_1 \textbf{ or } \ldots \textbf{ or } f(\bar{x}) = y_k \ \leftarrow random(f(\bar{x}) : \{X : p(X)\})^3 \tag{2.6}$$

$$\begin{aligned} truly\_random(f(\bar{x})) \ \leftarrow \ & random(f(\bar{x}) : \{X : p(X)\}), \\ & not\ do(f(\bar{x}), y_1), \ldots, not\ do(f(\bar{x}), y_k) \end{aligned} \tag{2.7}$$

$$\leftarrow f(\bar{x}) = Y,\ \textbf{not}\ p(Y), random(f(\bar{x}) : \{X : p(X)\}). \tag{2.8}$$

[3]Disjunction here is a so called shifted disjunction [Dix et al., 1996], that is, the disjunctive rule is viewed as a shorthand for the collection of rules:

$$\begin{gathered} f(\bar{x}) = y_1 \ \leftarrow random(f(\bar{x}) : \{X : p(X)\}), not\ f(\bar{x}) = y_2, \ldots, not\ f(\bar{x}) = y_k \\ \ldots \\ f(\bar{x}) = y_k \ \leftarrow random(f(\bar{x}) : \{X : p(X)\}), not\ f(\bar{x}) = y_1, \ldots, not\ f(\bar{x}) = y_{k-1}. \end{gathered}$$

$$\begin{aligned} \leftarrow \;\; & not\ f(\bar{X}) = Y, \\ & do(f(\bar{X}), Y). \end{aligned} \tag{2.9}$$

Intuitively, the rules (2.6) and (2.8) guarantee that if $random(f(\bar{x}) : \{X : p(X)\})$ is true, then $f(\bar{x})$ is assigned the value satisfying condition p, rule (2.7) makes sure that $truly\_random(f(\bar{x}))$ is true iff the value of $f(\bar{x})$ is assigned as the result of a truly random experiment, i.e. an experiment without any intervention, and rule (2.9) guarantees that the atoms made true by interventions are indeed true.

- The rule

$$\begin{aligned} \leftarrow \;\; & not\ random(f(\bar{X}) : \{X : p_1(X)\}), \\ & \ldots, \\ & not\ random(f(\bar{X}) : \{X : p_n(X)\}), \\ & do(f(\bar{X}), Y). \end{aligned} \tag{2.10}$$

  where $random(f(\bar{X}) : \{X : p_1(X)\}), \ldots, random(f(\bar{X}) : \{X : p_n(X)\})$ are all special attribute terms of Σ formed by *random* with $f(\bar{X})$ as the first argument. Intuitively, the rule guarantees that an attempt to apply *do* to a non-random $f(\bar{X})$ leads to inconsistency.

In addition, for every rule r which is not a general axiom, we disallow literals formed by *truly_random* and *random* to occur in the body of r.

We will sometimes refer to axioms of the forms (2.4), (2.5) and (2.9) as *value-checking* axioms. Also, we will refer to the rules of the program other than general axioms as *user-defined*.

As usual, a rule with variables[4] is understood as a shorthand for the collection of rules obtained by replacing the variables with the properly-sorted ground terms of Σ.

In what follows we assume that, unless otherwise stated, programs and other program elements we refer to are ground.

[4]A variable of P-log is an identifier starting with an upper case letter.

Note that our syntax differs from the syntax defined in [Baral et al., 2009] in the following ways.

a) We explicitly allow partial attributes and clarify the meaning of $a \neq y$ and **not** $a = y$.

b) We allow special attribute terms to occur in rules' bodies.

More details on the proposed changes and a more general definition of P-log syntax can be found in [Balai & Gelfond, 2017].

2.1.3 P-log Declarations

We will use *declarations* to describe the signatures of P-log programs.

Sort declarations are used to define sorts and the assignments of sort names to sorts. A sort declaration is of the form:

$$sort\_name = sort\_expression$$

where *sort_name* is a unique identifier preceded by the symbol # and *sort_expression* denotes a sort. For example, the declaration

$$\#block = [b][1..9]$$

defines sort $\#block$ consisting of elements $b1..b9$

The declaration

$$\#fluent\_on = on(\#block, \#block)$$

defines sort $\#fluent\_on$ consisting of records of the form $on(bi, bj)$, where bi, bj are elements of the sort $\#block$.

The declaration

$$\#s = \#s1 + \#s2$$

defines a sort $\#s$ whose elements are the union of elements of previously defined sorts $\#s1$ and $\#s2$.

For more details about the syntax and semantics of sort declarations, please refer to [Balai et al., 2013].

A statement

$$f : s_1, \dots, s_n \to s \tag{2.11}$$

is a declaration of attribute f with parameters $s_1, \dots, s_n$ and the range s. We will refer to (2.11) as an *attribute declaration*.

In what follows, we will write each P-log program as a sequence of sort declarations, followed by a sequence of attribute declarations, followed by a sequence of rules. Every declaration and every rule will end with a dot. We will often omit P-log general axioms from the rules, the declaration of the sort boolean $\#boolean$ assigned to the set of object constants $\{true, false\}$, and the facts of the form $p'(X)$ if the program denotation contains a shorthand $random(a)$ denoting $random(a : \{X : p'(X)\})$. We will write comments as lines starting with a percent sign (%). For example:

```
% Sorts
#n = {1,2,3}.

% Attributes
f: #n -> #boolean.
a,b: #boolean.

% Rules
random(a:{X:f(X)}).
random(b).
f(1) :- a = 2.
f(2).
```

```
f(3).
```

2.2 Semantics of P-log

We start with defining possible worlds of a (ground) P-log program. As in ASP, programs with variables will be viewed as shorthands for the sets of ground instantiations of their rules (which, of course, should be faithful to the declarations of the program). We introduce some basic terminology before defining the semantics.

2.2.1 Interpretations

An *interpretation* over signature Σ is a (possibly partial) mapping I from the attribute terms of Σ into values from their corresponding ranges. We assume that on arithmetic symbols I coincides with their standard interpretation.

In what follows, a denotes a ground attribute term, l denotes a literal, el denotes an extended literal, and B denotes a set of extended literals. The *satisfiability relation* between I and an element O (atom, literal, extended literal or a rule) of Σ (denoted by $I \models O$) is defined as follows:

1. $I \models a = y$ if $I(a) = y$,
2. $I \models \neg(a = y)$ if $I(a) = y'$ where $y' \neq y$,
3. $I \models \textbf{not } l$ if $I \not\models l$,
4. $I \models B$ if for every $el \in B$, $I \models el$,
5. $I \models l \leftarrow B$ if $I \not\models B \vee I \models l$, and
6. $I \models \leftarrow B$ if $I \not\models B$.

We say that an atom A of Σ is *true* in I if $I \models A$ and *false* in I if $I \models \neg A$. If A is neither *true* nor *false* in I then it is *undefined* in I.

We will often represent an interpretation I as the set of non-arithmetic atoms satisfied by I. We will use standard mathematical notation $I_1 \subseteq I_2$ to denote that I_1 is a subset of I_2 and $I_1 \subsetneq I_2$ to denote that I_1 is a proper subset of I_2.

Example 1. Consider a signature Σ with sort $\#s = \{1, 2\}$, attributes a and b with range $\#s$, boolean attribute p defined on $\#s$, and an interpretation

$$I = \{a = 1, b = 1, p(1), random(a : \{X : p(X)\}), truly\_random(a)\}$$

of Σ. Note that, while $a = 2$, $b = 2$, are *false* in I, $p(2)$ is not: it is *undefined* in I. Consequently, **not** $p(2)$ and a rule

$$random(b : \{X : p(X)\}) \leftarrow a = 1, a \neq 2, \textbf{not } p(2)$$

are satisfied by I.

□

2.2.2 Possible Worlds

Next we will define the possible worlds of a P-log program. As expected, the definition is very similar to the definition of answer sets for logic programs, and consists of two parts.

Definition 1 (Possible world, part I)**.**
Let Π be a ground P-log program not containing literals preceded by *not*. An interpretation I of the signature Σ of Π is called a *possible world* of Π if it satisfies the following conditions:

1. Every rule of Π is satisfied by I.

2. There is no interpretation I_0 such that $I_0 \subsetneq I$ and I_0 satisfies every rule of Π.

□

To define the semantics of programs with default negation, we will need the standard definition of the *reduct* [Gelfond & Lifschitz, 1988]: for a program Π and interpretation I the *reduct* of Π with respect to I (denoted by Π^I) is a P-log program obtained from Π by

1. removing all rules whose bodies contain a literal of the form *not* l such that $I \models l$, and
2. removing all other extended literals of the form *not* l from the program rules.

The second part of the definition of possible world deals with programs containing default negation.

Definition 2 (Possible world, part II)**.**
Let Π be an arbitrary ground P-log program. An interpretation I is a *possible* world of Π if I is a possible world of Π^I. □

Let us consider several examples. For attribute term a with range $\{y_1, \ldots, y_n\}$, we will often use shorthand

$$random(a) \leftarrow B$$

denoting a collection of rules

$$\begin{aligned} & random(a : \{X : p'(X)\}) \leftarrow B \\ & p'(y_1) \\ & \ldots \\ & p'(y_n) \end{aligned} \tag{2.12}$$

where p' is a boolean attribute term of Σ with a single parameter of the sort $range(a)$. We will only consider programs where p' does not occur in the rules other than those from (2.12) and the corresponding general axioms.

Example 2. Consider the program Π_1:

```
a,b,c: #boolean.
random(a).
b :- c, -a.
do(a, false).
random(c).
```

It is not difficult to see that the program has two possible worlds W_1 and W_2:

$$W_1 = \{\neg a, b, c, do(a, false), random(a), truly\_random(c), random(c)\}$$

and

$$W_2 = \{\neg a, \neg c, do(a, false), random(a), truly\_random(c), random(c)\}.$$

□

Example 3. Consider the program Π_2:

```
a: #boolean.
obs(a=true).
```

Π_2 has no possible worlds. Note that $W = \{obs(a = true)\}$ is not a possible world, because of the general reality check axiom

$$\begin{aligned} \leftarrow \quad & not\ a = Y, \\ & obs(a = Y). \end{aligned}$$

Since a is undefined in W, $not\ a$ is true, and hence W does not satisfy the axiom.

If we add the rule

$$random(a)$$

to Π_2, it will have one possible world

$$W = (\{obs(a = true), a, truly\_random(a), random(a)\}),$$

where the observation $obs(a = true)$ is consistent with the belief in a.

If however we were to replace $random(a)$ by

$$\neg a$$

Π_2 would become inconsistent again, because any interpretation that satisfies both rules $\neg a$ and $obs(a, true)$ will violate a reality check axiom.

□

Example 4. Consider the program Π_3:

```
a: #boolean.
p: #boolean -> #boolean.
random(a:{X:p(X)}).
```

The program has no possible worlds. Note that $W_1 = \{random(a, \{X : p(X)\}) = true\}$ is not a possible world of Π_3, because the axiom

$$a \textbf{ or } \neg a \leftarrow random(a : \{X : p(X)\})$$

is not satisfied by W_1.

Note that $W_2 = \{random(a, \{X : p(X)\}) = true, a\}$ is also not a possible world of Π_3, because the axiom

$$\leftarrow a = Y, not\ p(Y), random(a : \{X : p(X)\}).$$

is not satisfied by W_2. □

Example 5. Consider the following program Π_4:

```
a,q,r: #boolean.
random(a).
q :- not a.
r :- not -a.
```

and interpretation $I = \{random(a), truly\_random(a), a = true, r = true\}$. We show that I is a possible world of Π. The user-defined rules of Π_4 and the corresponding rules of the reduct Π_4^I are shown below.

Table 2.1: The rules of Π_4 and its reduct with respect to I

	Π_4	Π_4^I
r_1	`random(a).`	`random(a).`
r_2	`q :- not a.`	(removed)
r_3	`r :- not ¬a.`	`r.`

It is easy to see that I is a possible world of Π_4^I, therefore it is also a possible world of Π_4. Similarly, we can show that the interpretation $I = \{random(a) = true, a = false, q = true\}$ is a possible world of Π_4.

□

For a program Π, by $\Omega(\Pi)$ we will denote the collection of all possible worlds of Π. It is easy to check that the following proposition is true:

Proposition 1. Every possible world W of a program Π satisfies every rule of Π.

□

We will also state and prove the set-inclusion minimality of possible worlds in Proposition 13.

2.2.3 Probabilities

As in [Baral et al., 2009], we require a program Π to satisfy certain conditions.

Condition 1 (Unique selection rule)**.** [5]

If Π contains two rules r_1 and r_2, each of which is not an instance of a general axiom, such that for some attribute term a

[5]Note that this condition is stronger than the original Condition 1 from [Baral et al., 2009]. The original condition allows for a program with a possible world W to contain rules $r_1 : random(a) \leftarrow B_1$. and $r_2 : a = y \leftarrow B_2$. s.t. W satisfies both B_1 and B_2, while the new one prohibits such programs. We believe that the new condition better captures the intuition of a unique value selection for random attribute terms. Moreover, it is not clear whether or not a should be considered random in a possible world which satisfies the bodies of both of the rules r_1 and r_2.

- $head(r_1)$ is of the form $a = y$ or $random(a, p)$, and
- $head(r_2)$ is of the form $a = y$ or $random(a, p)$,

then no possible world of Π satisfies $body(r_1)$ and $body(r_2)$.

□

Condition 2 (Unique probability assignment).
If Π contains a random selection rule

$$random(a(\bar{t}) : \{Y : p(Y)\}) \leftarrow B$$

along with two different probability atoms

$$pr(a(\bar{t}) \mid B_1) = v_1 \text{ and } pr(a(\bar{t}) \mid B_2) = v_2$$

then no possible world of Π satisfies B, B_1, and B_2. □

Condition 3 (No probabilities assigned outside of dynamic range).
If Π contains a random selection rule

$$random(a(\bar{t}) : \{Y : p(Y)\}) \leftarrow B_1$$

along with probability atom

$$pr(a(\bar{t}) = y| \; B_2) = v$$

then no possible world of Π satisfies B_1 and B_2 but does not satisfy $p(y)$. □

Let Π be a P-log program with signature Σ, W be an interpretation of Σ, a be an attribute term of Σ, and r be a random selection rule of the form

$$random(a : \{X : p(X)\}) \leftarrow B$$

such that W satisfies B. Let $PO(W, r, a)$ be the set of terms defined as follows:

$$PO(W, r, a) = \{y \mid W \text{ satisfies } p(y) \text{ and } y \in range(a)\}.$$

We will refer to elements of the set $PO(W, r, a)$ as *possible outcomes* of a in W via rule r, and to every atom $a = y$ s.t. $y \in PO(W, r, a)$ as a *possible atom* in W via r.

Let Π be a P-log program and a be a random attribute term of the signature of Π. For every possible world W of Π such that $W \models truly\_random(a)$ and every possible atom $a = y$ in W via r, we will define the corresponding causal probability $P(W, a = y)$. Whenever possible, the probability of an atom $a = y$ will be directly assigned by pr-atoms of the program and denoted by $PA(W, a = y)$. To define probabilities of the remaining atoms we assume that by default, all values of a given attribute which are not assigned a probability by pr-atoms are equally likely. Their probabilities will be denoted by $PD(W, a = y)$. (PA stands for *assigned probability* and PD stands for *default probability*).

More precisely, for each atom $a = y$ possible in W via some rule r or Π:

1. Assigned probability:

   If Π contains $pr(a = y \mid B) = v$, $W \models B$, then

   $$PA(W, a = y) = v$$

   (note that condition 2 implies that the probability is uniquely defined).

2. Default probability:

   Let

   $$A_a(W) = \{y \mid a = y \text{ is possible in } W \text{ and } PA(W, a = y) \text{ is defined}\},$$

   $$D_a(W) = \{y \mid a = y \text{ is possible in } W\} \setminus A_a(W)$$

and $\alpha_a(W) = \sum_{y \in A_a(W)} PA(W, a = y)$.

The default probability of $a = y$ in W is defined as follows:

$$PD(W, a = y) = \frac{1 - \alpha_a(W)}{|D_a(W)|}$$

3. Finally, the causal probability $P(W, a = y)$ of $a = y$ in W is defined by:

$$P(W, a = y) = \begin{cases} PA(W, a = y) & \text{if } y \in A_a(W) \\ PD(W, a = y) & \text{otherwise.} \end{cases}$$

Definition 3 (Measure)**.**

1. Let W be an interpretation of Π. The *unnormalized probability*, $\hat{\mu}_\Pi(W)$, of W *induced by* Π is

$$\hat{\mu}_\Pi(W) = \prod_{W(a)=y} P(W, a = y)$$

where the product is taken over atoms for which $P(W, a = y)$ is defined.

2. Suppose Π is a P-log program having at least one possible world with nonzero unnormalized probability. The *measure*, $\mu_\Pi(W)$, of a possible world W *induced by* Π is the unnormalized probability of W divided by the sum of the unnormalized probabilities of all possible worlds of Π, i.e.,

$$\mu_\Pi(W) = \frac{\hat{\mu}_\Pi(W)}{\sum_{W_i \in \Omega(\Pi)} \hat{\mu}_\Pi(W_i)}$$

When the program Π is clear from the context we may simply write $\hat{\mu}$ and μ instead of $\hat{\mu}_\Pi$ and μ_Π respectively. □

Definition 4 (Probability)**.**

Suppose Π is a P-log program having at least one possible world with nonzero unnormalized probability. The *probability*, $P_\Pi(E)$, of a set E of possible worlds of program Π is the sum of the measures of the possible worlds from E, i.e.

$$P_\Pi(E) = \sum_{W \in E} \mu_\Pi(W).$$

□

When Π is clear from the context we may simply write P instead of P_Π.

Definition 5 (Probability of a literal)**.**
The *probability* with respect to program Π of a literal l of Π, $P_\Pi(l)$, is the sum of the measures of the possible worlds of Π in which l is true, i.e.

$$P_\Pi(l) = \sum_{W \models l} \mu_\Pi(W).$$

□

Note that, given that conditions 1-3 are satisfied, the function P_Π is defined iff

$$\sum_{W_i \in \Omega(\Pi)} \hat{\mu}_\Pi(W_i) \neq 0$$

2.3 A Note on Activity Records in the Bodies of Rules

As we discussed in the introduction, the new version of P-log allows programs where observations and actions may occur in the bodies of user-defined program rules. However, as we will see in this section, they can always be eliminated. More precisely, let U be the set of activity records of a program Π. A *simplification* of Π, denoted by Π_U is obtained from Π by:

1. removing all user-defined rules whose bodies include an e-literal formed by *do* or *obs* which is not satisfied by U (viewed as a collection of atoms), and

2. removing all remaining extended literals formed by *do* and *obs* from the bodies of user-defined rules.

As stated by the following proposition, a program Π is equivalent to its simplification:

Proposition 2. Let Π be a P-log program and U be the set of activity records of Π. There exists a bijection $\psi : \Omega_{\Pi} \rightarrow \Omega_{\Pi_U}$ such that for every possible world W of Π

1. $W = \psi(W)$, and

2. $\mu_{\Pi}(W) = \mu_{\Pi_U}(W)$

□

To simplify the future discussion, we will only consider programs not containing observations and actions in user-defined rules.

CHAPTER III
DYNAMICALLY CAUSALLY ORDERED P-LOG PROGRAMS

Causally ordered programs were first introduced in [Baral et al., 2009] where they were used to prove the coherency of P-log programs. Later, a query answering algorithm developed in [Zhu, 2012] was shown to be sound for programs from a broad subset of this class. We start this section by restating the original definition of causally ordered programs from [Baral et al., 2009], adapted to the new syntax and semantics. We also correct several errors confirmed by at least one of the authors of [Baral et al., 2009]. We then define a new class of programs, called dynamically causally ordered (dco), and show some interesting examples of programs in this class that are not causally ordered. In the next sections, we will prove the coherency of dco unitary programs and describe a query answering algorithm for them.

Let Π be a (ground) P-log program with signature Σ.

3.1 Causally Ordered Programs

We start this section by restating the original definition of causally ordered programs from [Baral et al., 2009], adapted to the new syntax and semantics. We also correct several errors confirmed by at least one of the authors of [Baral et al., 2009]. We will use these definitions in examples given in Section 3.3.

As in [Baral et al., 2009], for a random selection rule

$$random(a : \{X : p(X)\}) \leftarrow B \tag{3.1}$$

we will say that every atom of the form $a = y$ *occurs* in the head of (3.1), and that any ground instance of $p(X)$ and literals occurring in B *occur* in the body of (3.1). We will also say that atom $random(a : \{X : p(X)\})$ occurs in the head of (3.1). Also, we will say that an atom $a = y$ occurs in an observation $obs(a = y)$, literal $a \neq y$ occurs in observation $obs(a \neq y)$, and that atom $a = y$ occurs in action $do(a, y)$. We will use these notions of occurrence throughout this dissertation.

Definition 6 (Dependency relations)**.**

Let l_1 and l_2 be literals of Σ. We say that

1. l_1 is *immediately dependent* on l_2, written as $l_1 \leq_i l_2$, if there is a rule r of Π such that l_1 occurs in the head of r and l_2 occurs in the body of r;

2. l_1 *depends* on l_2, written as $l_1 \leq l_2$, if the pair $\langle l_1, l_2 \rangle$ belongs to the reflexive transitive closure of relation $\leq_i$;

3. An attribute term $a_1(\bar{t}_1)$ *depends* on an attribute term $a_2(\bar{t}_2)$ if there are literals l_1 and l_2 formed by $a_1(\bar{t}_1)$ and $a_2(\bar{t}_2)$ respectively such that l_1 depends on l_2. □

□

Definition 7 (Leveling function)**.**

A *leveling function*, $|\ |$, of Π maps the attribute terms of Σ onto a set $\{0..n\}$ of natural numbers. It is extended to other syntactic entities over Σ as follows:

$$|a(\bar{t}) = y| = |a(\bar{t}) \neq y| = |not\ a(\bar{t}) = y| = |not\ a(\bar{t}) \neq y| = |a(\bar{t})|$$

We'll often refer to $|e|$ as the *level* of e. Finally, if B is a set of expressions then $|B| = max(\{|e| : e \in B\})$.

□

Definition 8 (Random attribute term)**.**

A attribute term $a(\bar{t})$ of Σ is called *random* if Π contains a rule of the form:

$$random(a, p) \leftarrow B$$

□

Definition 9 (Strict probabilistic leveling and reasonable programs).
A leveling function $|\ |$ of Π is called *strict probabilistic* if

1. no two random attribute terms of Σ have the same level under $|\ |$,

2. for every random selection rule $[r]\ random(a(\bar{t}) : \{Y : p(Y)\}) \leftarrow B$ of Π we have $|a(\bar{t})| > |\{p(y) : y \in range(a)\} \cup B|$,

3. for every probability atom $pr_r(a(\bar{t}) = y \mid B)$ of Π we have $|a(\bar{t})| > |B|$,

4. if $a_1(\bar{t}_1)$ is a random attribute term, $a_2(\bar{t}_2)$ is a non-random attribute term, and $a_2(\bar{t}_2)$ depends on $a_1(\bar{t}_1)$ then $|a_2(\bar{t}_2)| \geq |a_1(\bar{t}_1)|$, and

5. if $a_1(\bar{t}_1)$ and $a_2(\bar{t}_2)$ are random attribute terms of Π such that $a_1(\bar{t})_1$ depends on $a_2(\bar{t}_2)$, then $|a_1(\bar{t}_1)| > |a_2(\bar{t}_2)|$.

A P-log program Π which has a strict probabilistic leveling function is called *reasonable.*

□

Let Π be a reasonable program with signature Σ and leveling $|\ |$, and let $a_1(\bar{t}_1), \ldots, a_n(\bar{t}_n)$ be an ordering of its random attribute terms induced by $|\ |$. By L_i for $0 \leq i \leq n$ we denote the set of literals of Σ which do not depend on literals formed by $a_j(\bar{t}_j)$ where $i < j$. Π_i for $0 \leq i \leq n$ consists of all declarations of Π, along with the regular rules, random selection rules, actions, and observations of Π such that every literal occurring in them belongs to L_i. We'll often refer to $\Pi_0, \ldots, \Pi_n$ as a $|\ |$-induced structure of Π.

Before proceeding we introduce some terminology.

Definition 10. *(Random Attribute Term Active in a Possible World of* Π*)*
Let a be a random attribute term of Π and W an interpretation of Σ. Term $a(\bar{t})$ is *active* in W with respect to Π if there is y such that $a(\bar{t}) = y$ is possible in W via some rule of Π.

□

Definition 11 (Causally ordered program)**.**

Let Π be a P-log program not containing activity records with a strict probabilistic leveling $|\ |$ and let a_i be the i^{th} random attribute of Π with respect to $|\ |$. Let $\Pi_0, \ldots, \Pi_n$ be the $|\ |$-induced structure of Π. We say that Π is *causally ordered* if

1. Π_0 has exactly one possible world,

2. if W is a possible world of Π_{i-1} and atom $a_i(\bar{t}_i) = y_0$ is possible in W with respect to Π_i then the program $W \cup \Pi_i \cup \{\leftarrow not\ a_i(\bar{t}_i) = y_0\}$ has exactly one possible world, and

3. if W is a possible world of Π_{i-1} and $a_i(\bar{t}_i)$ is not active in W with respect to Π_i then the program $W \cup \Pi_i$ has exactly one possible world.

□

For the examples of causally ordered programs, please refer to [Baral et al., 2009].

3.2 Dynamically Causally Ordered Programs

In this section we introduce dynamically causally ordered programs. We start with a few auxiliary definitions. We first introduce a new notion of dependency. Unlike in the previous definition (Def. 6), pr-atom $pr(a = y \mid B)$ introduces dependencies of $a = y$ on the literals in B.

Definition 12 (Dependency relations #2)**.**

Let Π be a P-log program and l_1 and l_2 be literals of the signature of Π. We say that

1. l_1 is *immediately dependent* on l_2 in Π, written as $dep^i_{\Pi}(l_1, l_2)$, if one of the following two conditions hold: there is a rule or pr-atom r of Π such that l_1 occurs in the head of r and l_2 occurs in the body of r

2. l_1 *depends* on l_2 in Π, written as $dep_{\Pi}(l_1, l_2)$, if the pair $\langle l_1, l_2 \rangle$ belongs to the reflexive transitive closure of relation dep^i_{Π}, and

3. Attribute term a_1 *depends* on attribute term a_2 in Π, written as $dep_\Pi(a_1, a_2)$ if there are literals l_1 and l_2 formed by a_1 and a_2 respectively such that $dep_\Pi(l_1, l_2)$.

□

In what follows, unless otherwise specified, we will assume that the new notion of dependency is used.

Definition 13 (Probabilistic leveling for random attribute terms)**.**
A *probabilistic leveling* of a program Π is an ordering $a_1, \ldots, a_k$ of random attribute terms of Π. □

Given a probabilistic leveling $a_1, \ldots, a_k$, attribute term a_i (where $1 \leq i \leq k$) has *level* i.

We will next define a total leveling of Π which assigns levels to non-random attribute terms of Π as well. We will need some auxiliary definitions.

Definition 14 (Program base)**.**
Let Π be a program. Let Σ_{base} be the signature consisting of attribute terms which do not depend on any random attribute terms of Π, and let R_{base} be the collection of rules of Π s.t. every literal occurring in R_{base} is a literal of Σ_{base}. We will refer to the program with signature Σ_{base} and rules R_{base} as the *base* of Π.

□

Definition 15 (Useless rules elimination)**.**
Let Π be a program such that the base of Π has a unique possible world W_{base}. By $red(\Pi)$ we will denote the program obtained from Π by removing all pr-atoms and rules whose bodies contain an e-literal from Σ_{base} not satisfied by W_{base}. □

Example 6. Consider the program Π_5

```
a,b,c,d:boolean
a.
```

```
random(b) :- not a.
random(c) :- a,b.
pr(c|b,d) = 0.5
pr(c|-b,a) = 0.5
```

Attribute terms a and d do not depend on b and c. The base of Π_5 consists of the fact a. Therefore, it has a unique possible world $W_{base} = \{a\}$.

$red(\Pi_5)$ is:

```
a,b,c,d:boolean
a.
random(c) :- a,b.
pr(c|-b,a) = 0.5
```

The pr-atom $pr(c \mid b, d) = 0.5$ was removed because d has level 0 in Π_5, and W_{base} does not satisfy d.

□

Now we are ready to define total leveling:

Definition 16 (Total leveling)**.**
Let Π be a program such that $red(\Pi)$ is defined. A probabilistic leveling $a_1, \ldots, a_k$ is expanded as follows into *total leveling of* Π, $|\ |$, which also assigns levels to all non-random attribute terms of Π. If a is a non-random attribute term of Π of the form $random(b, p)$, then $|random(b, p)| = |b|$. Otherwise:

1. $|a| = 0$ iff a does not depend on any random attribute term of Π in $red(\Pi)$.

2. $|a| = i$ iff i is the level of the random attribute a_i such that

   (a) a depends on a_i in $red(\Pi)$ and

   (b) there is no random attribute a_j with level j such that a depends on a_j in $red(\Pi)$ and $j > i$.

We will say that the total leveling $|\,|$ is *determined* by probabilistic leveling $a_1, \ldots, a_k$.

□

Unless otherwise stated, in what follows, by leveling of a program Π we will mean total leveling of Π.

Similarly to the leveling function from Definition 7, total levelings are extended to the e-literals of Σ as follows:

$$|not\ a = y| = |not\ a \neq y| = |a = y| = |a \neq y| = |a|$$

Definition 17 (Dynamic structure)**.**
Let Π be a program such that $red(\Pi)$ is defined and $L = a_1, \ldots, a_k$ be a probabilistic leveling of Π. We say that Π has a *dynamic structure* $\Pi_0, \ldots, \Pi_k$ *induced by* $a_1, \ldots, a_k$ if Π_i $(0 \leq i \leq k)$ is the program such that

1. the signature Σ_i of Π_i consists of all attribute terms of Π whose levels are i or less in the total leveling of Π determined by $a_1, \ldots, a_k$,
2. if r is a rule or a pr-atom of Π, then Π_i contains r iff all the literals occurring in r are of Σ_i.

□

Example 7. For example, consider the program Π:

```
f: #boolean.
f :- not -f.
-f :- not f.
```

Π_0 contains all rules of Π and, therefore, has two possible worlds $\{f\}$ and $\{\neg f\}$. Thus, condition 1 of definition 17 is violated and Π has no dynamic structure.

□

Definition 18 (Falsified set of e-literals)**.**
We will say that a set B of e-literals of signature Σ is *falsified* by an interpretation I of signature Σ' if B contains a member of signature Σ' that is not satisfied by I.

□

Definition 19 (Active random selection rule)**.**
A random selection rule

$$random(a : \{X : p(X)) \leftarrow B$$

is *active* with respect to an interpretation W of a signature Σ' (which may be different from Σ) if

1. W satisfies B,
2. all atoms of the form $p(y)$ where y is in the range of a are atoms of Σ', and
3. W satisfies an atom $p(y)$ for some y from the range of a.

□

Now we are ready to give the definition of a dynamically causally ordered (dco) program. The definition will consist of three parts. We will first define dco programs via a given probabilistic leveling. We next define dco programs not containing activity records. Finally, we will define arbitrary dco programs.

Definition 20 (Program dynamically causally ordered via a probabilistic leveling)**.**
Let Π be a program not containing activity records. Π is *dynamically causally ordered (dco)* via a probabilistic leveling $a_1, \ldots, a_k$ of Π if there exists a dynamic structure $\Pi_0, \ldots, \Pi_k$ induced by $a_1, \ldots, a_k$ such that Π_0 has a unique possible world and for every $i \in \{1..k\}$, if W_{i-1} is a possible world of Π_{i-1}, then

1. if r is a rule or a pr-atom of Π with a_i in the head, and r is not a ground instance of a general axiom of Π, then the body of r is either falsified or satisfied by W_{i-1}. Moreover, if r is of the form

$$random(a_i : \{X : p(X)\}) \leftarrow B \tag{3.2}$$

and W_{i-1} satisfies B, then r is active in W_{i-1},

2. if Π_i contains a rule of the form (3.2) such that W_{i-1} satisfies B, then for every $y \in range(a_i)$ s.t. W_{i-1} satisfies $p(y)$, the program $W_{i-1} \cup \Pi_i \cup \{\leftarrow not\ a_i = y\}$ has exactly one possible world, and

3. if for every rule of Π of the form (3.2), W_{i-1} falsifies B, then $W_{i-1} \cup \Pi_i$ has exactly one possible world.

□

Definition 21 (Dynamically causally ordered program - I)**.**
Let Π be a program not containing activity records. Π is *dynamically causally ordered* if Π is dynamically causally ordered via some probabilistic leveling of Π.

□

Definition 22 (Dynamically causally ordered program - II)**.**
Let Π be an arbitrary program, and Π' be the program obtained from Π by removing all activity records. Π is *dynamically causally ordered via probabilistic leveling* A of Π if Π' is dynamically causally ordered via A (by definition 20). Π is *dynamically causally ordered* iff Π is dynamically causally ordered via some probabilistic leveling of Π. We will also say that A is a *dynamic causal probabilistic leveling of* Π iff Π is dynamically causally ordered via A.

□

In the next section we will give examples of some interesting dco programs that are not causally ordered.

3.3 Examples

In this section we give examples of dynamically causally ordered programs that do not belong to $\mathcal{S}$ (the class mentioned in the introduction (Chapter I)). For each of such programs, we will prove a stronger claim that it does not belong to neither $\mathcal{S}$ nor to the class of causally ordered programs defined in [Baral et al., 2009]. Not that, since $\mathcal{S}$ is a subclass of causally ordered programs, it is sufficient to show that each program does not belong to causally ordered class.

When discussing the examples, we will often use the following definition:

Definition 23 (Possible outcomes of a random attribute in an interpretation)**.** Let random selection

$$random(a : \{X : p(X)) \leftarrow B$$

of Π be *active* with respect to an interpretation W of a signature Σ'. We will say that every member of $\{X \mid W$ satisfies $p(X)\}$ is a possible outcome of a in W.

□

3.3.1 Die

We throw a die until we get outcome 1 or make 5 throws. What's the probability that we will make 5 throws?

A P-log representation of the story, the program Π^d, is given below:

```
% Sorts
#outcome = 1..6.
#step = 1..5.



% Attributes
throw : #step -> #outcome.
made_5th_throw: #boolean.
```

```
% Rules

% the outcome of the die at step 1 is random
random(throw(1)).

% if the value of the die at the previous step, T2, was not 1,
% then the outcome of the die at current step, T, is random
random(throw(T)) :- throw(T2) != 1, T = T2+1.

% the fifth throw was made if the die takes some value, X, at step 5
made_5th_throw :- throw(5) = X.
```

Claim 1. Π^d *is not causally ordered.*

Proof. We will prove a stronger claim:

$$\text{there does not exists a strict probabilistic leveling (Def. 9) of } \Pi^d \tag{3.3}$$

For the sake of contradiction, suppose there exists a strict probabilistic $|\,|$ of Π^d. From the rule

$$random(throw(T)) \leftarrow throw(T2) \neq 1, T = T2 + 1 \tag{3.4}$$

and condition 2 of definition 9 it follows:

$$\forall T, T2 \in \#step : |throw(T)| > |throw(T2)| \tag{3.5}$$

in particular, from (3.5) we have $|throw(1)| > |throw(2)|$ and $|throw(2)| < |throw(1)|$. Contradiction. □

Claim 2. Π^d *is dynamically causally ordered.*

Proof. Consider the following probabilistic leveling of Π^d:

$$L = throw(1), throw(2), throw(3), throw(4), throw(5), throw(6)$$

By | | we will denote the total leveling of Π^d determined by L. Recall that the rule

$$random(throw(X))$$

is a shorthand for $random(throw(X), p)$ for a fresh boolean attribute term p s.t. $p(1), \ldots, p(6)$ are facts of Π . The base of Π^d, Π^d_{base} has no random attribute in its signature, and its rules consist of general axioms, facts $\{p(i) \mid i \in \{1..6\}\}$, and general axioms constructed from attribute terms of Π of level 0.

Clearly, Π^d_{base} has a unique possible world. Let us call it W_0. Therefore, there exists a dynamic structure $\Pi_0, \ldots, \Pi_6$ induced by L. For every $i > 0$, the rules of Π_i consist of

1. the rules of Π_{i-1}
2. if $i = 1$, the rule $random(throw(1))$
3. if $i > 1$, the rule

$$random(throw(i)) \leftarrow throw(i-1) \neq 1, i = i - 1 + 1 \tag{3.6}$$

   and for every $k \in \{0..i-2\} \cup \{i\}$, the rules:

$$random(throw(i)) \leftarrow throw(k) \neq 1, i = k + 1 \tag{3.7}$$

4. if $i = 6$, the rule

$$made\_6th\_throw \leftarrow throw(6) = X \tag{3.8}$$

5. general axioms for $throw(1), \ldots throw(i)$ and, if $i = 6$, for $made\_6h\_throw$

We will prove conditions 1-3 of definition 20 hold for every $i \geq 1$ for every possible world W_{i-1} of Π_{i-1}.

1. We prove condition 1.

   (a) $i = 1$. Clearly, $random(throw(1))$ is active in W_0. The bodies of other rules of Π contain an arithmetic e-literal (and arithmetic literals belongs to every P-log sorted signature) not satisfied by W_0.

   (b) $i \in \{2..5\}$. By construction of Π_{i-1}, we have that the signature of every possible world W_{i-1} contains $throw(i-1)$.

   We consider two cases:

      i. W_{i-1} assigns 1 to $throw(i-1)$, or does not assign a value to $throw(i-1)$. In this case it is easy to see that the body of every random selection rule for $throw(i)$ contains an e-literal $throw(i-1) \neq 1$ of the signature of W_{i-1} which is not satisfied by W_{i-1}. Thus, condition 1 is satisfied.

      ii. W_{i-1} assigns a value different from 1 to $throw(i-1)$. In this case it is easy to see that the rule

$$random(throw(i)) \leftarrow throw(i-1) \neq 1, i = i - 1 + 1$$

      is active in W_{i-1}, and all other random selection rule for $throw(i)$ contain an arithmetic e-literal not satisfied by W_{i-1}.

2. To prove condition 2, suppose the rule

$$random(throw(i)) \leftarrow throw(i-1) \neq 1, next(i-1, i).$$

   is active in W_{i-1}. The possible outcomes of $throw(i)$ with respect to W_{i-1} are $\{1, 2, 3, 4, 5, 6\}$. For each $v \in \{1, 2, 3, 4, 5, 6\}$, the program $W_{i-1} \cup \Pi_i \cup \{\leftarrow not\ throw(i) = v\}$ has exactly one possible world $W_{i-1} \cup \{throw(i) = v\}$.

3. Finally, we prove condition 3. Suppose the rule

$$random(throw(i)) \leftarrow throw(i-1) \neq 1, next(i-1, i).$$

is not active with respect to W_{i-1}. In that case the program $\Pi_i \cup W_{i-1}$ has exactly one possible world which coincides with W_{i-1}.

□

3.3.2 Random Tree

Consider a tree defined by a collection of facts of the form arc(X,Y) (arc(X,Y) = true when there is a directed arc from node X to node Y of the tree). Each node of the tree is assigned a value. If a node is a leaf, a value is selected (uniformly) at random from $\{1,2,3,4,5,6\}$. *If a node is not a leaf, its value is selected randomly from the values of the node's children.*

The corresponding program Π^t is:

```
% Sorts
#node = {1,2,3,4,5}.
#value = {1,2,3,4,5,6}.

% Attributes
arc: #node, #node -> #boolean.
value_of : #node -> #value.
possible_value: #value, #node -> #boolean.

% Rules

% Tree arcs. arc(i,j) means there is an arc from i to j
arc(4,5).
```

```
arc(3,5).
arc(2,4).
arc(1,4).

% Tree definitions:

% Node X not a leaf if there is a directed arc with the end in X
leaf(X) = false :- arc(Y,X).

% Node X is a leaf if there is no reason to believe that it is not
leaf(X) = true :- not leaf(X) = false.

% Random selections:

% Every leaf node takes a value at random
random(value_of(N)) :- leaf(N).

% Every non-leaf node X takes a value from the set of possible
% values {X:possible_value(X,N)}
random(value_of(N):{X:possible_value(X,N)}) :- -leaf(N).

% Value N is possible in Node X if it a value of its child
possible_value(X,N) :- arc(N1,N), value_of(N1) = X.
```

Claim 3. *The program Π^t is not causally ordered.*

Proof. We will prove a stronger claim:

$$\text{there does not exists a strict probabilistic leveling (Def. 9) of } \Pi^t \tag{3.9}$$

For the sake of contradiction, suppose there exists a strict probabilistic | | of Π^t. From the rule

$$possible\_value(X, N) \leftarrow arc(N1, N), value\_of(N1) = X \quad (3.10)$$

and condition 4 of definition 9 it follows:

$$\forall N, N1 \in \#step : \forall X \in \#value : |possible\_value(X, N)| \geq |value\_of(N1)| \quad (3.11)$$

In in particular, from (3.11) we have

$$|possible\_value(1, 1)| \geq |value\_of(1)| \quad (3.12)$$

On the other hand, from the rule

$$random(value\_of(N) : \{X : possible\_value(X, N)\}) \leftarrow \neg leaf(N) \quad (3.13)$$

and condition 2 of definition 9 we have:

$$\forall N \in \#step : \forall X \in \#value : |value\_of(N)| > |possible\_value(X, N)| \quad (3.14)$$

In particular, from (3.14) we have:

$$value\_of(1) > possible\_value(1, 1) \quad (3.15)$$

From (3.15) and (3.12) we have a contradiction. Therefore, (3.9) holds.

□

Claim 4. *The program Π^t is dynamically causally ordered.*

Proof. Consider the following probabilistic leveling of Π^t:

$$L = value\_of(1), value\_of(2), value\_of(3), value\_of(4), value\_of(5)$$

By $|\ |$ we will denote the total leveling of Π^t induced by L. Recall that the rule

$$random(value\_of(N)) \leftarrow leaf(N)$$

is a shorthand for

$$random(value\_of(N) : \{X : p(X)\}) \leftarrow leaf(N)$$

s.t. $p(1), \ldots, p(6)$ are facts of Π^t .

Π^t_{base} consists of facts of the form $p(i)$ for $i \in \{1..6\}$, the rules defining the tree:

$$arc(4, 5)$$
$$arc(3, 5)$$
$$arc(2, 4)$$
$$arc(1, 4)$$
$$leaf(X) = false \leftarrow arc(Y, X)$$
$$leaf(X) = true \leftarrow not\ leaf(X) = true$$

and the general axioms constructed from the attribute terms of level 0. Clearly, the base has exactly one possible world

$$W_0 = \{arc(4, 5), arc(3, 5), arc(2, 4), arc(1, 4), leaf(2), leaf(1), leaf(3), \neg leaf(4), \neg leaf(5)\}.$$

Therefore, there exists a dynamic structure $\Pi_0, \ldots, \Pi_6$ induced by L, where $\Pi_0 = \Pi^t_{base}$. We next construct $\Pi_1, \ldots \Pi_6$ By $rv_1 - rv_5$ we denote ground rules (3.16) -

(3.20) of Π^t below:

$$random(value\_of(1)) \leftarrow leaf(1). \tag{3.16}$$

$$random(value\_of(2)) \leftarrow leaf(2). \tag{3.17}$$

$$random(value\_of(3)) \leftarrow leaf(3). \tag{3.18}$$

$$random(value\_of(4) : \{X : possible\_value(X, 4)\}) \leftarrow \neg leaf(4). \tag{3.19}$$

$$random(value\_of(5) : \{X : possible\_value(X, 5)\}) \leftarrow \neg leaf(5). \tag{3.20}$$

By $rp_1..rp_4$ we denote the rules 3.21 - 3.24 below:

$$possible\_value(X, 4) \leftarrow arc(1, 4), value\_of(1) = X. \tag{3.21}$$

$$possible\_value(X, 4) \leftarrow arc(2, 4), value\_of(2) = X. \tag{3.22}$$

$$possible\_value(X, 5) \leftarrow arc(3, 5), value\_of(3) = X. \tag{3.23}$$

$$possible\_value(X, 5) \leftarrow arc(4, 5), value\_of(4) = X. \tag{3.24}$$

For $1 \leq i \leq 5$, the program Π_i is the union of:

1. the rules from Π_0,
2. the rules $rv_1, \ldots, rv_i$,
3. the rules rp_1 and rp_2, if $i \geq 2$,
4. the rules rp_3 and rp_4, if $i \geq 4$,
5. other rules of Π^d whose bodies are falsified by W_0,
6. general axioms involving attribute terms of the signature of Π_i.

We will prove conditions 1-3 of definition 20 hold for every $i \geq 1$ and for every possible world W_{i-1} of Π_{i-1}. Clearly, $W_0 \subseteq W_{i-1}$, and W_{i-1} does not contains atoms of level

0 other than those in W_0 [1].

1. We prove condition 1.

   (a) $i \leq 3$. It is easy to see that the rule

   $$random(value\_of(i)) \leftarrow leaf(i) \tag{3.25}$$

   is active in W_{i-1}, and the second random selection rule with $value\_of(i)$ in the head contains a literal $\neg leaf(i)$ falsified by W_0 (and, therefore, by W_{i-1}).

   (b) $i > 3$. It can be shown that the rule:

   $$random(value\_of(i) : \{X : possible\_value(X, i)\}) \leftarrow \neg leaf(i)$$

   is active in W_{i-1}. Indeed, the atoms $possible\_value(X, i)$ depend on only on attribute terms $value(1), value(2)$ in $red(\Pi)$, and, therefore, belong to the signature of W_{i-1}. $possible\_value(X, i)$ is true for at least one X. Indeed, it is easy to show that

   - if $i = 4$, then for each of the rules rp_1, rp_2, there is a ground instance whose body is satisfied by W_{i-1}, and,
   - if $i = 5$, then for each of the rules rp_3, rp_4, there is a ground instance whose body is satisfied by W_{i-1}.

2. We prove condition 2.

   (a) $i \leq 3$. The rule (3.25) is active in W_{i-1}. The possible outcomes of $value(i)$ with respect to W_{i-1} are $\{1, \ldots, 6\}$. $\Pi_i \cup \{W_{i-1}\} \cup \{\leftarrow not\ value(i) = v\}$ has a unique possible world $W_{i-1} \cup \{value(i) = v\}$.

[1] We will generalize and prove this result in Lemma 12.

(b) $i > 3$. The possible outcomes of $value(i)$ with respect to W_{i-1} are $1, \ldots, 6$. $\Pi_i \cup \{W_{i-1}\} \cup \{\leftarrow not\ value(i) = v\}$ has a unique possible world containing $W_{i-1} \cup \{value(6-i) = v\}$ and atoms formed by attribute $possible\_value$.

3. Condition 3 is vacuously satisfied, since, as we have shown, there exists a random selection with $value\_of(i)$ in the head active in W_{i-1}.

□

3.3.3 Blood Type Problem

The problem description is based on section 4.1.3 in [Zhu, 2012].

The ABO blood group system distinguishes four types of bloods: ***A, B, AB*** *and* ***O****. The type of blood of each individual is determined by two genes inherited from his/her parents (one gene is inherited from each parent). The pair of genes is also called a genotype. There are three types of genes:* ***a****,* ***b*** *and* ***o****, and 6 corresponding genotypes:* ***ao,bo,ab,aa,bb,oo****. The genotypes* ***ao, bo, ab, aa, bb, oo*** *are distributed in generation* 1 *with probabilities 0.24, 0.24, 0.18, 0.09, 0.09, 0.16 correspondingly. The corresponding blood type of a person for each combination of inherited genes (which determines his/her genotype) is given in Table 3.1.*

Table 3.1: ABO blood group system

Mother's gene \ Father's gene	**a**	**b**	**o**
a	**A**	**AB**	**A**
b	**AB**	**B**	**B**
o	**A**	**B**	**O**

If an individual A has genes of types X and Y, and an individual B has genes of types F and H, their child will have one of the pairs of genes (X,F), (Y,F), (X,H), (Y,H); where each pair is inherited with probability 0.25.

P-log program Π^b represents the story:

```
%% Blood Type Problem
%% Sorts
#person={mary, todd, john}.
#gene ={g_a,g_b,g_o}.
#genotype = g(gene(X), gene(Y)) : X<=Y.
#bloodtype={b_a,b_b,b_o,b_ab}.
#generation = {1,2}.

%% Attributes
genotype_of: #person -> #genotype.
bloodtype_of: #person -> #bloodtype.
mother_of: #person -> #person.
father_of: #person -> #person.
generation_of: #person -> #generation.
possible_combination: #genotype, #genotype, #genotype -> #boolean.
belongs_to: #gene, #genotype -> #boolean.

%% Rules
% generations
generation_of(john) = 2.
generation_of(mary) = 1.
generation_of(todd) = 1.

% family tree
mother_of(john)=mary.
father_of(john)=todd.
```

```
% blood_type(X)=G : the blood type of person X
% determined by the genes he or she inherits from parents
% as described in table 1

bloodtype_of(X)=b_a :- genotype_of(X) = g(g_a,Y), Y!=g_b.
bloodtype_of(X)=b_b :- genotype_of(X) = g(g_b,Y), Y!=g_a.
bloodtype_of(X)=b_ab :- genotype_of(X) = g(g_a,g_b).
bloodtype_of(X)=b_o :- genotype_of(X) = g(g_o,g_o).

% the genotypes of the parents of a person X in the old generation are
% distributed as it is given in the problem statement

random(genotype_of(P)):- generation(P) = 1.
pr(genotype_of(X) = g(g_a,g_o)|generation(X) = 1) = 24/100.
pr(genotype_of(X) = g(g_b,g_o)|generation(X) = 1) = 24/100.
pr(genotype_of(X) = g(g_a,g_b)|generation(X) = 1) = 18/100.
pr(genotype_of(X) = g(g_a,g_a)|generation(X) = 1) = 9/100.
pr(genotype_of(X) = g(g_b,g_b)|generation(X) = 1) = 9/100
pr(genotype_of(X) = g(g_b,g_b)|generation(X) = 1) = 16/100.


% the genotypes of a person in the new generation are randomly
% inherited from his/her parents

random(genotype_of(P):{G:possible_genotype(P,G)}) :-
                                        generation_of(P) = 2.

possible_genotype(P,G) :- father(P,F),
```

```
                              mother(P,M),
                              genotype_of(F) = U,
                              genotype_of(M) = V,
                              possible_combination(G,U,V).



% possible_combination(G,U,V) is true if G can be the genotype of a
% child whose parents have genotypes U and V

possible_combination(g(G1,G2),U,V) :- belongs_to(G1,U),
                                      belongs_to(G2,V).
possible_combination(g(G1,G2),U,V) :- belongs_to(G2,U),
                                      belongs_to(G1,V).

% belongs_to(G,GT) is true if gene G belongs to the pair of genes in
% genotype GT
belongs_to(G,g(G,X)).
belongs_to(G,g(X,G)).
```

Claim 5. Π^b *is not causally ordered.*

Proof. We will prove, by contradiction, a stronger claim:

$$\text{there does not exists a strict probabilistic leveling (Def. 9) of } \Pi^b. \tag{3.26}$$

Suppose there exists a strict probabilistic | | of Π^b. From the rule:

$$random(genotype\_of(P) : \{G : possible\_genotype(P, G)\}) \leftarrow generation\_of(P) = 2$$

By condition 2 of definition 9, for all $p \in \#person, g \in \#genotype$, we have that

$$|genotype\_of(p)| > |possible\_genotype(p, g)|. \tag{3.27}$$

On the other hand, from the

$$\begin{aligned} possible\_genotype(P, G) \leftarrow & father(P, F), \\ & mother(P, M), \\ & genotype\_of(F) = U, \\ & genotype\_of(M) = V, \\ & possible\_combination(G, U, V) \end{aligned}$$

by condition 4 of definition 9 we obtain that for all $p \in \#person, g \in \#genotype, f \in \#person$

$$|possible\_genotype(p, g)| \geq |genotype\_of(f)| \tag{3.28}$$

By replacing p and f in equations (3.27) and (3.28) with the same element of the sort $\#person$, and g with some element of the sort $\#genotype$ we get a contradiction to (3.26). Therefore, strict probabilistic leveling does not exist for Π^b and it is not causally ordered.

□

Claim 6. *The program Π^b is dynamically causally ordered.*

Proof. Recall that the rule

$$random(genotype\_of(P)) \leftarrow generation\_of(P) = 1$$

is a shorthand for

$$random(genotype\_of(P) : \{X : p(X)\}) \leftarrow generation\_of(P) = 1$$

s.t.

$$\{p(G) \mid G \in \#genotype\} \tag{3.29}$$

are facts of Π^b. Π^b_{base} consists of facts (3.29), the rules:

$generation\_of(john) = 2$

$generation\_of(mary) = 1$

$generation\_of(todd) = 1$

$mother\_of(john) = mary$

$father\_of(john) = todd$

$possible\_combination(g(G1, G2), U, V) \leftarrow belongs\_to(G1, U), belongs\_to(G2, V)$

$possible\_combination(g(G1, G2), U, V) \leftarrow belongs\_to(G2, U), belongs\_to(G1, V)$

$belongs\_to(G, g(G, X))$

$belongs\_to(G, g(X, G))$

and the general axioms for each of the attribute terms occurring in them. Clearly, Π^b_{base} has a unique possible world. Let us refer to the possible world as W_0. Therefore, there exists a dynamic structure $\Pi_0, \ldots, \Pi_3$ of Π^b induced by the leveling

$$L = genotype\_of(mary), genotype\_of(todd), genotype\_of(john)$$

where, as it can be shown, $\Pi_0 = \Pi^b_{base}$.

We next construct $\Pi_1, \ldots, \Pi_3$. The rules of Π_1 are the union of the rules of Π_0, the rule

$$random(genotype\_of(mary)) \leftarrow generation\_of(mary) = 1, \tag{3.30}$$

some rules whose bodies are falsified by W_0, and and the general axioms for attribute term $genotype\_of(mary)$.

The rules of Π_2 are the union of the rules of Π_1 and the rule

$$random(genotype\_of(todd)) \leftarrow generation\_of(todd) = 1 \tag{3.31}$$

some rules whose bodies are falsified by W_0, and the general axioms for attribute term $genotype\_of(todd)$.

Finally,Π_3 contains all the rules of Π, where the only rule with $genotype\_of(john)$ in the head whose body is not falsified by W_0 is

$$\begin{aligned} random(genotype\_of(john) : \{G : possible\_genotype(john, G)\}) \leftarrow \\ generation\_of(john) = 2 \end{aligned} \tag{3.32}$$

We will prove conditions 1-3 of definition hold 20 for every $1 \leq i \leq 3$ for every possible world W_{i-1} of Π_{i-1}. We will use the facts that [2].

$$W_0 \subseteq W_{i-1} \tag{3.33}$$

and

$$W_{i-1} \text{ does not contains atoms of level 0 other than those in } W_0 \tag{3.34}$$

1. We prove condition 1.

   (a) $i \leq 2$. Let $parent$ be $mary$ if $i = 1$, and $todd$ if $i = 2$. It is easy to see that the rule

$$random(genotype\_of(parent)) \leftarrow generation\_of(parent) = 1 \tag{3.35}$$

[2]We will generalize and prove this result in 12.

is active in W_{i-1}, and the body of the second random selection rule that has $genotype\_of(parent)$ in the head has literal $generation\_of(parent) = 2$ falsified by W_0 (and, therefore, as it follows from (3.33)), by W_{i-1}).

(b) $i = 3$. It can be shown that the rule (3.32) is active in W_{i-1}. Indeed,

$$generation\_of(john) = 2 \in W_0 \subseteq W_{i-1},$$

so the body of the rule is satisfied. The attribute terms of the form

$$possible\_genotype(john, G)$$

depend on only on attribute terms $genotype\_of(mary)$, $genotype\_of(todd)$ in $red(\Pi)$, and, therefore, belong to the signature of W_{i-1}. We next show that

$$possible\_genotype(john, G) \text{ is true in } W_2 \text{ for at least one } G \tag{3.36}$$

Since $W_0 \subseteq W_2$, we have that

$$W_2 \text{ satisfies the bodies of rules (3.30) and (3.31)} \tag{3.37}$$

Since both (3.30) and (3.31) belong to Π_2 by construction, and W_2 is a possible world of Π_2, we have that

$$W_2 \text{ assigns values to both } genotype\_of(mary) \text{ and } genotype\_of(todd) \tag{3.38}$$

Let m and t be the values of $genotype\_of(mary)$ and $genotype\_of(todd)$ in W_2 respectively. That is,

$$genotype\_of(mary) = m \in W_2 \tag{3.39}$$

and

$$genotype\_of(todd) = t \in W_2 \tag{3.40}$$

From the rules of Π_0 and the fact that $W_0 \subseteq W_2$, we have that there exists at least one constant g s.t.

$$possible\_combination(g, t, m) \in W_2 \tag{3.41}$$

Since the rule r'

$$\begin{aligned} possible\_genotype(john, g) \leftarrow & father(john, todd), \\ & mother(john, mary), \\ & genotype\_of(todd) = t, \\ & genotype\_of(mary) = m, \\ & possible\_combination(g, t, m) \end{aligned}$$

belongs to Π_2. W_2 is a possible world of Π_2,

$$\{father(john, todd), mother(john, mary)\} \subseteq W_0 \subseteq W_2,$$

and, by (3.39) and (3.40), and (3.41), W_2 satisfies the body of r'. Therefore, W_2 satisfies the head $possible\_genotype(john, g)$ of r', and

$$possible\_genotype(john, g) \in W_2.$$

Therefore, (3.36) holds.

2. We prove condition 2.

   (a) $i \leq 2$. Let *parent* be *mary* if $i = 1$, and *todd* if $i = 2$. It is easy to see that the rule (3.35) is active in W_{i-1}. The possible outcomes of $genotype\_of(parent)$ with respect to W_{i-1} are all elements of $\#genotype$. For each $v \in \#genotype$, the program has a unique possible world W, where

      i. if $i = 2$, then $W = W_{i-1} \cup \{genotype\_of(parent) = v\}$

      ii. if $i = 3$, then $W = W_{i-1} \cup \{genotype\_of(parent) = v\} \cup$
      $\{possible\_genotype(john, X) \mid \exists U, V : genotype\_of(todd) = U,$
      $genotype\_of(mary) = V, possible\_combination(X, U, V) \in W_0\}$

   (b) $i = 3$. Rule (3.32) is active in W_2. The possible outcomes of attribute term $genotype\_of(john)$ are

   $$\{Y \mid possible\_genotype(john, Y) \in W_2\}$$

   For each value $y \in \{Y \mid possible\_genotype(john, Y)\}$, the program $\Pi_i \cup \{W_{i-1}\} \cup \{\leftarrow not\ value(i) = v\}$ has a unique possible world $W_{i-1} \cup \{genotype\_of(parent) = v\}$.

3. Condition 3 is vacuously satisfied, because the rules (3.30), (3.31) and (3.32) are active for $i = 1, 2$ and 3 respectively.

□

3.3.4 Not Dynamically Causally Ordered

In this subsection we show an example of a program which is causally ordered, but not dynamically causally ordered. Consider the program Π_6:

```
a,b,h,x,y:#boolean.
```

```
p: #boolean -> #boolean

h. p(true). p(false).
random(x:{X:p(X)}).
random(y:{X:p(X)}) :- x.
a:- not b, x.
b:- not a, x.
a :- not h, y.
:- a,y.
:- a,-y.
```

Claim 7. Π_6 *is causally ordered.*

Proof. Let Σ_6 be the signature of Π_6. Consider the leveling $|\ |$ which maps attributes to natural numbers as follows:

- $|p(true)| = 0, |p(false)| = 0, |x| = 1, |y| = 2, |a| = |2|, |b| = 2, |random(x, p)| = 1,$
  $|truly\_random(x)| = 1, |truly\_random(y)| = 2, |random(y, p)| = 2,$ $|h| = 0$
- for every attribute term z of Π_6 and boolean z, $|obs(z) = y| = 0, |do(z, y)| = 0$

It is easy to check that $|\ |$ is strict probabilistic (Def 9).

We now construct the structure $\Pi'_0, \ldots, \Pi'_2$ of Π_6 induced by $|\ |$. L_0 consists of literals formed by:

$$\{h, p(true), p(false), random(x, p)\}$$
$$\cup \{obs(z, y_1, y_2)|z \in attr(\Sigma_6)), y_1, y_2 \in \{true, false\}\}$$
$$\cup \{do(z, y)|z \in attr(\Sigma_6)), y \in \{true, false\}\}$$

L_1 is the union of L_0 and the literals formed by

$$\{x, random(y, p), truly\_random(x)\} \tag{3.42}$$

L_2 consists of all literals of Σ_6. Π'_0 consists of rules

$$h.$$
$$p(true).$$
$$p(false).$$

and the general axioms of Π with all literals occurring in them being from L_0. Π'_1 consists of the union of the rules of Π'_0 and the general axioms of Π_6 with all literals occurring in them being from L_1.

$\Pi'_2 = \Pi_6$.

We now check the condition of causally ordered program (Def. 11).

1. Condition 1 is true. Π'_0 has a unique possible world $\{h, p(true), p(false)\}$[3]

2. We check condition 2. We do it separately for $i = 1$ and $i = 2$.

   (a) $i = 1$. The atoms $x = true$ and $x = false$ are possible in W_0 w.r.t Π'_1. The program $\Pi'_1 \cup W_0 \cup \{\leftarrow not\ x = true\}$ has a unique possible world:

   $$W_1 = \{x, h, p(true), p(false)\}.$$

   The program $\Pi'_1 \cup W_0 \cup \{\leftarrow not\ x = false\}$ has a unique possible world:

   $$W'_2 = \{\neg x, h, p(true), p(false)\}$$

[3] as before, special atoms are omitted

(b) $i = 2$. The program Π'_1 has two possible worlds: W_1 and W_2. We check the condition for each of them separately.

i. W_1. The atoms $y = true$ and $y = false$ are possible in W_1 w.r.t Π'_2. The program $\Pi'_2 \cup W_1 \cup \{\leftarrow not\ y\}$ has a unique possible world:

$$\{x, h, p(true), p(false), y, b\}.$$

The program $\Pi'_2 \cup W_1 \cup \{\leftarrow not\ y = false\}$ has a unique possible world:

$$\{x, h, p(true), p(false), \neg y, b\}$$

ii. W_2. The attribute term y is not active in W_2 w.r.t Π'_2, thus this condition is trivially satisfied.

3. We check condition 3. We do it separately for $i = 1$ and $i = 2$.

(a) $i = 1$. Since a is active in W_0 w.r.t Π_1, this condition is trivially satisfied.

(b) $i = 2$. The program Π'_1 has two possible worlds: W_1 and W_2. We check the condition for each of them separately.

i. W_1. Since x is active in W_1 w.r.t Π'_2, this condition is trivially satisfied.

ii. W_2. The program $\Pi'_2 \cup W_2$ has a unique possible world:

$$\{\neg x, h, p(true), p(false)\}$$

□

Claim 8. Π_6 *is not dynamically causally ordered.*

Proof. Suppose

$$\Pi_6 \text{ is dco} \tag{3.43}$$

The signature of the base Π_{base}, Σ_{base}, consists of attribute terms:

$$\{h, p(true), p(false)\}$$
$$\cup \{obs(z, y_1, y_2) | z \in attr(\Sigma_6)), y_1, y_2 \in \{true, false\}\}$$
$$\cup \{do(z, y) | z \in attr(\Sigma_6)), y \in \{true, false\}\}$$

The rules of Π_0 consist of

$$h.$$
$$p(true).$$
$$p(false).$$

and the general axioms of Π with all literals occurring in them being from Σ_0. Clearly, Π_{base} has a unique possible world:

$$W_0 = \{h, p(true), p(false)\}$$

The program $red(\Pi_6)$ consists of rules:

$h.$

$p(true).$

$p(false).$

$random(x : \{X : p(X)\}.$

$random(y : \{X : p(X)\} \leftarrow x$

$a \leftarrow not\ b, x$

$b \leftarrow not\ a, x$

$\leftarrow a, y$

$\leftarrow a, \neg y$

$truly\_random(x) \leftarrow random(x, p), not\ do(x, true), not\ do(x, false)$

$truly\_random(y) \leftarrow random(y, p), not\ do(y, true), not\ do(y, false)$

In 1 (2) we show that Π_6 is not dynamically causally ordered via x, y (via y, x). Since x, y and y, x are the only two probabilistic levelings of Π_6, this will imply that Π_6 is not dco.

1. We consider probabilistic leveling $L_1 = x, y$. The total probabilistic leveling $|\ |_1$ induced by L_1 is defined as follows:

   - $|p(true)|_1 = 0, |p(false)|_1 = 0, |x|_1 = 1, |y|_1 = 2, |a|_1 = 1, |b|_1 = 1,$ $|random(x, p)|_1 = 1, |truly\_random(x)|_1 = 1, |truly\_random(y)|_1 = 2,$ $|random(y, p)|_1 = 2,\ |h|_1 = 0$
   - for every attribute term z of Π_6 and boolean z, $|obs(z) = y|_2 = 0$, $|do(z, y)|_2 = 0$

   Since Π_{base} has a unique possible world, there exists a dynamic structure Π_0^1, Π_1^1, Π_2^1 induced by L_1, where $\Pi_0^1 = \Pi_{base}$.

The signature Σ_1^1 of Π_1^1 consists of the union of the set of attribute terms in Σ_0 and

$$\{a, b, x, random(x, p)\}$$

and the rules of Π_1^1 consist of:

$$h$$
$$p(true)$$
$$p(false)$$
$$random(x : \{X : p(X)\}$$
$$a \leftarrow not\ b, x$$
$$b \leftarrow not\ a, x$$

and the general axioms of Π with all literals occurring in them being from Σ_1^1.

We will show that condition 2 of definition 21 is violated for $i = 1$, thus deriving a contradiction to (3.43). Clearly, the rule

$$random(x : \{X : p(X)\})$$

is active w.r.t W_0 since x is possible in W_0 w.r.t Π_1^1. However, the program $\Pi_1^1 \cup W_0 \cup \{\leftarrow not\ x\}$ has two possible worlds:

$$W_0 \cup \{a, x\}$$

and

$$W_0 \cup \{b, x\}$$

which is contradiction to condition 2 of definition 21.

2. We consider probabilistic leveling $L_2 = y, x$.

   We will show that condition 1 of definition 21 is violated for $i = 1$. W_0 is a possible world of Π_0^1, Π_6 contains a rule

$$random(y) \leftarrow x$$

   however, $\{x\}$ is neither satisfied nor falsified by W_0, since x does not belong to Σ_{base}.

□

CHAPTER IV
COHERENCY RESULT

In this section we give a sufficient condition of coherency of programs for the new version of P-log. More precisely, we show that dynamically causally ordered programs defined in chapter III satisfying an extra condition (unitarity) are coherent.

We start from restating the definitions from [Baral et al., 2009] related to *coherent* and *unitary* programs.

4.1 Coherent Programs

The notion of coherent programs was introduced in [Baral et al., 2009]. Intuitively, a program P is coherent if it is logically and probabilistically consistent. The former means that P has possible worlds. The latter says that causal probabilities, given by pr-atoms, entail corresponding conditional probabilities defined by the program. To provide a better intuition of coherency, we give some examples.

Example 8 (Coherent program)**.**
Consider program Π_7:

```
a,b:boolean.
random(a).
pr(a) = 0.3.
b :- a.
```

Π_7 is consistent and it has two possible worlds: $W_1 = \{a, b\}$ and $W_2 = \{\neg a\}$ with corresponding probabilistic measures 0.3 and 0.7. The probability of a in Π is therefore equal to

$$P_{\Pi_7}(a) = 0.3/(0.3 + 0.7) = 0.3$$

The probability of a defined by Π_7 matches the probability from the pr-atom, and the program is coherent.

□

Example 9 (Incohrent program)**.**

Consider program Π_8:

```
a,b:boolean.
random(a).
pr(a) = 0.3.
b :- a, not -b.
-b :- a, not b.
```

Π_8 has three possible worlds: $W_1 = \{a, b\}$, $W_2 = \{a, \neg b\}$ and $W_3 = \{\neg a\}$ with corresponding unnormalized probabilistic measures 0.3, 0.3 and 0.7. The probability of a is therefore equal to:

$$P_{\Pi_8}(a) = (0.3 + 0.3)/(0.3 + 0.3 + 0.7) \approx 0.462$$

$0.462 \neq 0.3$, so Π_8 is not coherent. Intuitively, the probabilistic inconsistency of Π_8 is explained by the rules $b \leftarrow a, not\ \neg b$, $\neg b \leftarrow a, not\ b$ which create non-determinism that is not resulted from random selections.

□

We next define coherent programs formally, starting from some notation. Let Π be an arbitrary program with signature Σ. We will extend Σ with a fresh attribute term a_B for every set B of e-literals of Σ. For a set of e-literals B of Σ , by $obs(B)$ we denote a set of two rules:

$$a_B \leftarrow B$$

$$\leftarrow \textbf{not } a_B$$

Intuitively, $obs(B)$ can be viewed as a generalization of observations of single literals. If a program Π is extended with the rules from $obs(B)$, the possible worlds of the new program will be the possible world of Π satisfying B.

Definition 24 (Program coherency)**.**
Let Π be a P-log program and Π' be a program obtained from Π by removing activity records. We will say that Π is *coherent* if:

- $P_{\Pi'}$ is defined.
- For every selection rule r of the form $random(a : \{X : p(X)\}) \leftarrow K$ and every probability atom of the form $pr(a = y \mid B) = v$ of Π, if $P_{\Pi'}(B \cup K)$ is not equal to 0 then $P_{\Pi' \cup obs(B) \cup obs(K)}(a = y) = v$.

□

Note that the first condition implies consistency of Π'. For the examples of coherent programs, please refer to [Baral et al., 2009].

4.2 Unitary Programs

Let Π be a ground P-log program containing random selection rule r of the form

$$random(a : \{X : p(X)\}) \leftarrow K \tag{4.1}$$

and

$$pr(a = y \mid B) = v \tag{4.2}$$

be a pr-atom of Π.

Let W_1 and W_2 be possible worlds of Π satisfying K from rule (4.1). We say that W_1 and W_2 are *probabilistically equivalent with respect to* r if

1. for all y, $W_1 \models p(y)$ if and only if $W_2 \models p(y)$, and
2. for every *pr*-atom of the form (4.2), $W_1 \models B$ if and only if $W_2 \models B$.

A *scenario* for r is an equivalence class of possible worlds of Π satisfying K, under probabilistic equivalence with respect to r. Let s be a non-empty scenario for rule r of Π, a be an attribute term of Π.

We will define the set of possible outcomes of a via r in s, denoted by $PO(s, r, a)$ to be equal to the set of possible outcomes of a via r in an arbitrary member W of s (the latter was introduced in section 2.2 and was denoted by $PO(W, r, a)$).

For a random selection rule r of the form (4.1) and scenario s for r, let $at_r(s)$ denote the set of probability atoms of the form (4.2) whose bodies are satisfied by every possible world in s.

Definition 25 (Unitary rule)**.**
Rule r of the form (4.1) is *unitary in* Π, or simply *unitary*, if for every scenario s of r, one of the following conditions holds:

1. For every y in $PO(s, r, a)$, $at_r(s)$ contains a *pr*-atom of the form (4.2), and, moreover, the sum of the values of the probabilities assigned by members of $at_r(s)$ is 1; or

2. There is a y in $PO(s, r, a)$ such that $at_r(s)$ contains no *pr*-atom of the form (4.2), and the sum of the probabilities assigned by the members of $at_r(s)$ is less than or equal to 1.

□

Definition 26 (Unitary program)**.**
A P-log program is *unitary* if each of its random selection rules is unitary. □

In what follows, we will refer to the class of dynamically causally ordered unitary programs as class $\mathcal{B}$, mentioned in the introduction (Chapter I).

For the examples of coherent programs, please refer to [Baral et al., 2009].

4.3 Coherency Theorem

Theorem 1. Every program from $\mathcal{B}$ is coherent.

□

Note that, in the condition 1 of Definition (20), it is necessary to consider pr-atoms of the form

$$pr(a_i = y \mid B) = v \tag{4.3}$$

for correctness of the theorem. We will next show that an alternative formulation of Definition 20 that omits this condition makes the theorem incorrect.

We will refer to the instance of the condition 1 when r is of the form (4.3) as condition A. The following example shows that there is a program which satisfies all conditions of dynamically causally ordered definitions excluding A, is unitary, but is not coherent.

Consider the program Π_9:

```
a,b,c: #boolean.
random(a).
pr(a | not -b, not b) = 0.3.
random(b) :- -a.
pr(-a | b) = 0.4.
pr(-a | - b)  = 0.6.
```

The program has 3 possible worlds;

$$W_1 = \{a\}, W_2 = \{\neg a, b\}, W_3 = \{\neg a, \neg b\}$$

with corresponding unnormalized measures:

$$\hat{\mu}(W_1) = 0.3, \hat{\mu}(W_2) = 0.2, \hat{\mu}(W_3) = 0.3$$

Claim 9. Π_9 satisfies all conditions of a dynamically causally ordered program except, possibly, A.

□

Proof. Consider the leveling a, b. We will show that Π_9 satisfies all the conditions of 20, omitting condition A. The dynamic structure $\Pi_0 - \Pi_2$ of Π_9 induced by the leveling is as follows:

- Π_0 is an empty program with an empty possible world W_0.
- Π_1 is

```
a: #boolean.
random(a).
```

- Π_2 coincides with Π.

We next check the conditions from 20, not including condition A (that is, we will only prove condition 1 for rules).

1. We prove condition 1 for rules.
   (a) $i = 1$. Clearly, $random(a)$ is active in W_0.
   (b) $i = 2$. The program Π_1 has two possible worlds: $W_1^1 = \{a\}$ and $W_1^2 = \{\neg a\}$[1]. We prove the condition for each of them separately.
      i. The only rule to consider is

$$random(b) \leftarrow \neg a,$$

whose body is falsified by W_1^1.

[1]As before, we are omitting atoms of the form $p(true)$ and $p(false)$ defining the dynamic ranges in the shorthands $random(a)$ and $random(b)$

ii. Again, the only rule to consider is

$$random(b) \leftarrow \neg a,$$

whose body is satisfied by W_1^2. Clearly, the rule is active in W_1^2.

2. We prove condition 2.

(a) $i = 1$. The rule $random(a)$ is active in W_0, where $true$ and $false$ are possible outcomes of a. The programs $W_0 \cup \Pi_1 \cup \{\leftarrow \textit{not}\ a = true\}$ and $W_0 \cup \Pi_1 \cup \{\leftarrow \textit{not}\ a = false\}$ have unique possible worlds $\{a = true\}$ and $\{a = false\}$ respectively.

(b) $i = 2$. The program Π_1 has two possible worlds: $W_1^1 = \{a\}$ and $W_1^2 = \{\neg a\}$.

i. The condition is vacuously satisfied for W_1^1, since the body of the only random selection rule of Π_9 containing b in the head is not satisfied by W_1^1.

ii. The body of the rule

$$random(b) \leftarrow \neg a$$

is satisfied by W_1^2. b has two possible outcomes in W_1^2: $true$ and $false$. The program $W_1^2 \cup \Pi_2 \cup \{\leftarrow \textit{not}\ b\}$ has a unique possible world: $\{\neg a, b\}$. The program $W_1^2 \cup \Pi_2 \cup \{\leftarrow \textit{not}\ \neg b\}$ has a unique possible world: $\{\neg a, \neg b\}$.

3. We prove condition 3.

(a) $i = 1$ The condition is trivially satisfied, because W_0 does not falsify the body of the rule $random(a)$.

(b) $i = 2$ The program Π_1 has two possible worlds: $W_1^1 = \{a\}$ and $W_1^2 = \{\neg a\}$. We consider them separately in i. and ii. respectively.

i. The body of the only random selection rule with b in the head,

$$random(b) \leftarrow \neg a,$$

is falsified by W_1^1. The program $\Pi_9 \cup W_1^1$ has a unique possible world: $\{\neg a\}$

ii. The condition is trivially satisfied, because W_1^2 does not falsify the body of the rule

$$random(b) \leftarrow \neg a$$

□

Claim 10. Π_9 is unitary.

□

Proof. To show that Π_9 is unitary, by definition 26, we need to show that both of the random selection rules

$$r_1 = random(a)$$

and

$$r_2 = random(b) \leftarrow \neg a$$

are unitary. The second rule is unitary by condition 2 of definition 25, because $at_{r_2}(s)$ is, clearly, empty, for every scenario s of r_2

We show that the rule r_1 is unitary. The scenarios for the rule are

$$s_1 = \{\{a\}\}$$

$$s_2 = \{\{\neg a, b\}\}$$

$$s_3 = \{\{\neg a, \neg b\}\}$$

Note that:

1. the possible worlds $\{a\}$, $\{\neg a, b\}$ are not in the same scenario because only one of them satisfies the body of $pr(a \mid b) = 0.4$.

2. the possible worlds $\{a\}$, $\{\neg a, \neg b\}$ are not in the same scenario because only one of them satisfies the body of pr-atom $pr(\neg a \mid \neg b) = 0.6$.

3. The possible worlds $\{\neg a, -b\}, \{\neg a, b\}$ are not in the same scenario because only one of them satisfies pr-atom $pr(\neg a \mid b) = 0.4$.

$PO(s_1, r_1, a)$ is $\{true, false\}$. We have only one pr-atom with a in the head in $at_{r_1}(s_1)$ whose body is satisfied by members of s_1:

$$pr(a \mid not\ \neg b, not\ b) = 0.3$$

and $0.3 < 1$.

For the scenario s_2 $PO(s_2, r_1, a)$ is $\{true, false\}$. We have only one pr-atom in $at_{r_1}(s_2)$ whose body is satisfied by members of s_2

$$pr(-a \mid b) = 0.4$$

and $0.4 < 1$.

For the scenario $s_3 = \{\neg a, b\}$, $PO(s_3, r_1, a)$ is $\{true, false\}$. We have only one pr-atom in $at_{r_1}(s_3)$ whose body is satisfied by members of s_3:

$$pr(a \mid \neg b) = 0.6$$

and $0.6 < 1$.

Therefore, for every scenario s of $random(a)$, condition 2 of Definition 25 is satisfied. Therefore, $random(a)$ is a unitary rule. Since both of the random selection rules of Π_9 are unitary, Π_9 is unitary.

□

Condition 1 from Definition 20 for random selection and regular rules is necessary to guarantee that the value of attribute a_i is decided in every possible world W_i. Even though there might be alternative sufficient conditions for coherency similar to the one we have and not requiring this condition, we believe this guarantee simplifies the presentation substantially.

The proof of Theorem 1 is given in appendix A.2. An important intermediate result of the proof is a formulation of splitting set theorem for P-log, originally introduced in [Lifschitz & Turner, 1994] for logic programs, is given in subsection A.2.2 of the proof.

CHAPTER V
ALGORITHMS

5.1 Introduction

We address the question of finding a probability $P_{\Pi}(Q)$, where Q is a literal. We will also refer to the literal as a *query* to Π. More complex conditional probabilities can be easily reduced to this case.

A naive approach to this question suggests the computation of all possible worlds of a given program. Consider, for instance, the following program Π_{10}:

```
a, b, f : #boolean.
random (a).
pr(a) = 0.3
random (b).
pr(b) = 0.6.
f :- a.
f :- b.
-f :- not f.
```

and a query f.

The program has 4 possible worlds :

$$W_1 = \{a, f, b\}, W_2 = \{a, f, \neg b\}, W_3 = \{\neg a, b, f\}, W_4 = \{\neg a, \neg f, \neg b\}$$

whose corresponding measures are:

$$\mu(W_1) = 0.18,\ \mu(W_2) = 0.12,\ \mu(W_3) = 0.42,\ \mu(W_4) = 0.28$$

The probability of f is the sum of the measures of the possible worlds satisfying the query:

$$P_{\Pi_{10}}(f) = \mu(W_1) + \mu(W_2) + \mu(W_3) = 0.18 + 0.12 + 0.42 = 0.72$$

We will design an algorithm that allows to compute the probability of f without computing all possible worlds of Π_{10}. For program Π_{10}, the algorithm will construct a tree shown in Figure 5.1, looking for partial assignments satisfying or falsifying f and representing collections of the program's possible worlds. The construction starts from an empty interpretation $\{\}$. Then random attribute a is selected[1] (shown as a square node a), and two children are added to the node a, each corresponding to possible outcomes, $true$ and $false$, of the attribute and their immediate consequences. In a similar manner, after selecting b, node $\{\neg a\}$ is further expanded by attaching the subtree rooted at b.

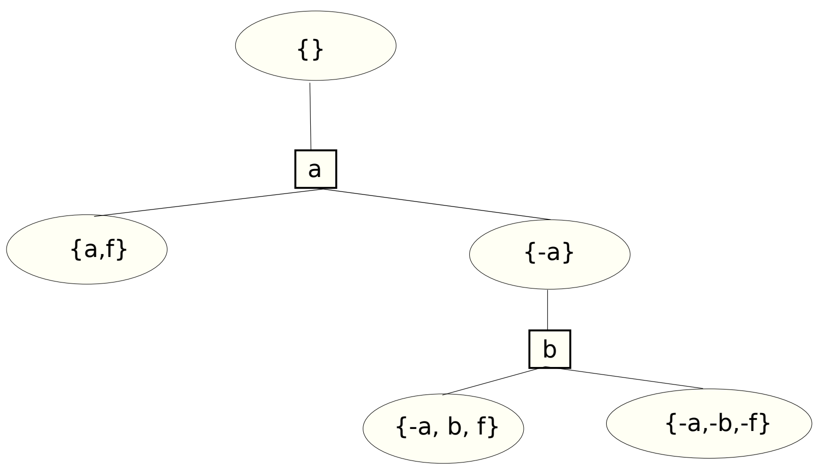

Figure 5.1: A tree for Π_{10}

The node $\{a, f\}$, however, is not extended further. There are two possible worlds of Π_{10} which are compatible with the node $\{a, f\}$: $W_1 = \{a, b, f\}$ and $W_2 = \{a, \neg b, f\}$, both assigning value true to f. The so called *unnormalized measure* of $\{a, f\}$, $\hat{\mu}(\{a, f\})$, is equal to the sum of the unnormalized measures of W_1 and W_2 (which

[1]The choice of the attribute actually depends on the heuristic. In this case we are assuming a is selected.

coincide with normalize measures, $\mu(W_1)$ and $\mu(W_2)$ in this case):

$$\hat{\mu}(\{a, f\}) = \mu(W_1) + \mu(W_2) \tag{5.1}$$

By definition of $\mu(W_1)$ and $\mu(W_2)$, we have:

$$\begin{aligned}
\hat{\mu}(\{a, f\}) &= \mu(W_1) + \mu(W_2) \\
&= P(W_1, a) \cdot P(W_1, b) + P(W_2, a) \cdot P(W_2, \neg b) \\
&= P(W_1, a) \cdot P(W_1, b) + P(W_1, a) \cdot P(W_2, \neg b) \text{ (since } P(W_1, a) = P(W_2, a)) \\
&= P(W_1, a) \cdot (P(W_1, b) + P(W_2, \neg b)) \\
&= P(W_1, a) \cdot 1 \\
&= P(W_1, a)
\end{aligned} \tag{5.2}$$

Therefore, the unnormalized measure of node $\{a, f\}$ is equal to the probability of a in this node, $P(\{a, f\}, a)$, and, therefore, can be simply read from the corresponding pr-atom $pr(a) = 0.3$. Hence, there is no need to extend the node $\{a, f\}$ further in order to build a tree which allows to compute the probability of f: the probability can be computed from the measures of the tree leaves $\{a, f\}$ and $\{a, b, f\}$ as follows:

$$\begin{aligned}
P_{\Pi_{10}}(f) &= \mu(W_1) + \mu(W_2) + \mu(W_3) \\
&= \mu(\{a, f\}) + \mu(W_3) && \text{(by (5.1))} \\
&= 0.3 + 0.42 && \text{(by (5.2) and the discussion above)} \\
&= 0.72
\end{aligned}$$

Note that the idea of stopping the tree construction in a particular node was first described in [Zhu, 2012]. We adapt the ideas from there for the new class $\mathcal{B}$ of coherent programs by defining a new type of search space (tree) and refining the corresponding conditions. We also address and correct a number of errors found in [Zhu, 2012].

In the next sections we give more details on the algorithm. In section 5.2 we will describe a transformation γ which maps programs from $\mathcal{B}$ containing user-defined rules of the form

$$a = y \leftarrow B \tag{5.3}$$

where a is a random attribute term into equivalent programs not containing such rules.

The transformation allows us to only consider programs from $\mathcal{B}$ which do not contain rules of the form (5.3) with random attribute term in the head. We believe this transformation allows to simplify the algorithm as well as the proof of its correctness substantially. According to our observations, programs containing such rules are rare, so this transformation will not affect most programs at all in practice.

In Section 5.3 we give auxiliary definitions needed for the algorithm and state propositions necessary for its correctness. Section 5.4 will contain the description of the algorithm, including pseudocodes and examples.

5.2 Transformation γ

Let Π be a program with signature Σ. $\gamma(\Pi)$ is defined as follows:

1. The signature of $\gamma(\Pi)$ consists of

   (a) all attribute terms of Σ,

   (b) a fresh boolean random attribute term $f_{do}(a)$ for every action $do(a = y)$ of Π, and

   (c) a fresh boolean attribute term $p_r(y)$ for every user-defined rule r of the form

$$a = y \leftarrow B$$

such that a is a random attribute term of Π.

2. The rules of $\gamma(\Pi)$ are obtained from the rules of Π by

   (a) for every user-defined rule of Π of the form

   $$a = y \leftarrow B$$

   such that a is a random attribute and Π contains an action $do(a, y')$ for some y', adding the rules:

   $$f_{do}(a) \leftarrow B$$
   $$\neg f_{do}(a) \leftarrow not\ f_{do}(a)$$
   $$obs(\neg f_{do}(a))$$

   (b) replacing every user-defined r rule of Π of the form

   $$a = y \leftarrow B$$

   such that a is a random attribute term of Π with the random selection rule:

   $$random(a : \{Y : p_r(Y)\}) \leftarrow B$$

   and the fact

   $$p_r(y).$$

Proposition 3. Let Π be a program from $\mathcal{B}$. We have:

1. $\gamma(\Pi)$ is from $\mathcal{B}$, and

2. there is a bijection ϕ from the possible world of Π to the possible worlds of $\gamma(\Pi)$ such that for every possible world W of Π:

(a) $\mu_{\Pi}(W) = \mu_{\gamma(\Pi)}(\phi(W))$, and

(b) W and $\phi(W)$ coincide on the atoms of Π.

□

5.3 Definitions

In this section we define the notions needed for the algorithm. The search space of the algorithm will be defined by *AI-trees*, and the algorithm will look for so called *solution trees* for a query Q – special subtrees of the search space (section 5.3.3) that will be used to compute the query probability. The nodes of the search tree will contain attribute terms and *e-interpretations* (Section 5.3.1). The tree will be built by gradually extending subtrees of an AI-tree: selecting a random attribute term and adding children assigning values to it (such attributes and assignments are defined in section 5.3.2). Sections 5.3.4 - 5.3.7 will discuss ideas and present notions related to the algorithm efficiency.

5.3.1 E-interpretations

We found it convenient to consider interpretations consisting of e-literals, rather than atoms. To define the consistency of such interpretations, we need an auxiliary definition.

Let Π be a program with signature Σ.

Definition 27 (Contrary e-literals)**.**
The e-literals l_1 and l_2 of Σ are called *contrary* if at least one of the following two conditions is satisfied:

1. l_1 is of the form $a = y$ and l_2 is of one of the forms:
    - $a \neq y$, or
    - $a = y_1$ where $y_1 \neq y$, or

2. l_2 is of the form *not* l_1

□

Definition 28 (Consistent set of e-literals).
A set S of e-literals is called *consistent* if it does not contain a pair of contrary e-literals.

□

Moreover, such interpretation will include simple consequences. For instance, assignment $\{\}$ will define an e-interpretation containing also e-literals $a \neq 2$ and *not* $a = 2$. This property is formalized by the following definition:

Definition 29 (Saturated set of e-literals).
A set S of e-literals is called *saturated* if it satisfies the following conditions:

1. if $a = y$ belongs to S, then for every $y_1 \in range(a) \setminus \{y\}$, $a \neq y_1$ belongs to S
2. if $a = y$ belongs to S, then *not* $a \neq y$ belongs to S
3. if $a \neq y$ belongs to S, then *not* $a = y$ belongs to S
4. if *not* $a \neq y$ belongs to S, then for every $y_1 \in range(a) \setminus \{y\}$, *not* $a = y_1$ belongs to S
5. if a is an attribute and there exists $y \in range(a)$ such that $\{not\ a = y' \mid y' \in range(a) \setminus \{y\}\} \subseteq S$, then *not* $a \neq y$ belongs to S

□

For a set I of e-literals, by $satr(I)$ we will denote the smallest superset of I which is saturated.

We are now ready to define e-interpretations:

Definition 30 (E-interpretation).
An *e-interpretation* I of Σ is a consistent saturated set of e-literals of Σ. □

In denoting e-interpretations, we will sometimes omit e-literals that can be obtained by saturation. For example, $\{a\}$ denotes a e-interpretation $satr(\{a\}) = \{a, not\ \neg a, a \neq false\}$. In addition, whenever it is clear from the context, we will omit e-literals formed by special attribute terms.

We will use shorthand $a = u$ to denote the set $\{not\ a = y \mid y \in range(a)\}$. For example, if a is a boolean attribute term, $a = u$ denotes $\{not\ a = true, not\ a = false\}$.

We next define satisfiability and falsification of program elements w.r.t e-interpretations. Let I be an e-interpretation of Σ.

Definition 31 (An e-interpretation satisfying an extended literal)**.**
I *satisfies* an e-literal l of Σ if $l \in I$.

□

Definition 32 (An e-interpretation satisfying a set of extended literals)**.**
A set L of e-literals is satisfied by I, if I satisfies every e-literal from L.

□

Definition 33 (An e-interpretation falsifying an extended literal)**.**
I *falsifies* an e-literal l of Σ if I contains a literal l_2 such that l_2 and l are contrary.

□

Definition 34 (An e-interpretation falsifying a set of extended literals)**.**
A set L of e-literals is *falsified* by I, if I falsifies some e-literal from L. □

We next define relations between e-interpretations of Σ and possible worlds of Π.

Definition 35 (A possible world compatible with an e-interpretation)**.**
A possible world W of Π *is compatible* with I if W satisfies every e-literal from I. □

Definition 36 (A consequence)**.**
Let I_1, I_2 be e-interpretations of Σ. We will say that I_2 is a consequence of I_1 w.r.t Π if:

1. $I_1 \subseteq I_2$, and

2. every possible world of Π compatible with I_1 is also compatible with I_2.

□

Proposition 4. Let I be an e-interpretation of Σ and W be a possible world of Π compatible with I. We have:

- if I satisfies an e-literal l of Σ, then W satisfies l,
- if I falsifies a literal l of Σ, then W does not satisfy l.

□

We next define an unnormalized measure of an e-interpretation, used by the algorithm for computing probabilities:

Definition 37 (E-interpretation's unnormalized measure)**.**
Let I be an interpretation of Σ and $W_1, \ldots, W_n$ be the possible worlds compatible with I. We will define the *unnormalized measure*, or simply *the measure* of I, denoted by $\hat{\mu}(I)$, as follows:

$$\hat{\mu}(I) = \prod_{i=1}^{n} \hat{\mu}(W_i)$$

□

5.3.2 Random Attributes Ready in an E-interpretation

The next collection of definitions is related to the selection and assigning values to attribute terms in an e-interpretation. Such attribute terms are called ready (for selection). Intuitively, if Π contains a rule

$$random(a : \{X : p(X)\}) \leftarrow B$$

such that B is satisfied by an e-interpretation I, and the set $\{X : p(X)\}$ is "decided" in I, then a is ready in I. If every rule with a in the head has a body falsified in I, a is ready as well (in this case, we can claim that a takes no value in possible worlds compatible with I).

Definitions 38-41 formalize these concepts.

Definition 38 (Decided attribute term)**.**
An attribute term a is *decided* in an e-interpretation I of Σ if $a = y \in I$ for some y in $range(a)$, or $a = u \subseteq I$

□

Let a be a random attribute term of Π, r be a random selection rule of the form:

$$random(a : \{X : p(X)) \leftarrow B, \text{ and}$$

I be an e-interpretation of the signature of Π such that I satisfies the body of r. We define $PO(I, r, a)$ as:

$$PO(I, r, a) = \{y \mid I \text{ satisfies } p(y) \text{ and } y \in range(a)\}$$

Definition 39 (Random attribute term active in an e-interpretation via a rule r)**.**
A random attribute term a of Π is *active* in an e-interpretation I of Σ via rule r if the following conditions are satisfied:

1. a is not decided by I.

2. r is the only rule of Π satisfying the following conditions:

   (a) the head of r is of the form $random(a : \{X : p(X)\})$[2];

   (b) I satisfies the body of r;

   (c) every attribute term $p(x)$, where $x \in range(a)$, is decided in I;

[2]recall we view $random(a)$ as a shorthand for $random(a : \{X : p(X)\})$, where r is the range of a

(d) $PO(I, r, a) \neq \emptyset$;

(e) for every $y \in PO(I, r, a)$, $satr(I \cup a = y)$ is consistent.

3. For every probability atom

$$pr(a = y \mid B_2) = v$$

in Π B_2 is either falsified or satisfied by I.

□

Definition 40 (Disabled random attribute term).
A random attribute term a of Π is *disabled* in an e-interpretation I of Σ if the following conditions are satisfied:

1. a is not decided by I, and
2. for every random selection rule of the form

$$random(a : \{X : p(X)\}) \leftarrow B$$

B is falsified by I.

□

Definition 41 (Ready random attribute term).
A random attribute term a of a program Π is *ready* in an e-interpretation I of Σ if a is either active or disabled in I. □

Definition 42 (Possible values of an attribute term ready in an e-interpretation).
Let I be an e-interpretation and a be an attribute term ready in I. We will say that y is a possible value of a in I iff:

1. a is active in I via rule r of Π, and $y \in PO(I, r, a)$, or

2. a is disabled in I, and $y = u$.

□

5.3.3 AI-trees

AI-trees (where 'A' stands for attribute term, and 'I' stands for interpretation) will be used to represent the search space of a family of algorithms computing the probability of a query of programs from $\mathcal{B}$.

Let Π be a program from $\mathcal{B}$ with signature Σ. By $int(\Sigma)$ we will denote the set of e-interpretations of Σ.

Definition 43 (AI-tree)**.**
The *AI-tree* of Π, parameterized by a partial function $f_\Pi : int(\Sigma) \rightsquigarrow int(\Sigma)$ is a tree T such that the following conditions are satisfied:

1. Each node of the tree is labeled with a random attribute term of Π (also referred to as an *a-node*), or an e-interpretation of Σ (also referred to as an *i-node*).

2. The root of T is labeled with $f_\Pi(\{\})$.

3. A node N_2 is a child of a node N_1 if at least one of the following two conditions is satisfied:

   (a) N_2 is an a-node with label a, N_1 is an i-node with label I, and a is ready in I,

   (b) N_1 is an a-node with label A whose parent is labeled with interpretation I, $f_\Pi(I \cup \{A = y\}, \Pi)$ is defined for every possible value y of A in I , and N_2 is an i-node with label $f_\Pi(I \cup \{A = y'\}, \Pi)$, where y' is a possible value of A in I.

□

By $T_\Pi\langle f\rangle$ we will denote the AI-tree of Π parameterized by f.

Whenever it is clear from the context, we will identify nodes with their labels.

We next define a special class of functions which will normally be used as parameters of AI-trees.

Definition 44 (Consequence function)**.**
We will say that a partial function $f : int(\Sigma) \rightsquigarrow int(\Sigma)$ is a consequence function of Π if:

1. $f(\{\})$ is defined, and
2. for $I \in int(\Sigma)$, if $f(I)$ is defined, then $f(I)$ is a consequence of I w.r.t. Π and no e-literal in $f(I) \setminus I$ is formed by a random attribute term of Π.

□

In what follows we will only consider AI-trees parameterized by consequence functions.

We now describe how, given an AI-tree, the probability of query Q can be computed. We will start from some auxiliary definitions.

Definition 45 (Compatible e-interpretation)**.**
An interpretation I of Σ is *compatible* if there is a possible world of Π compatible with I. Otherwise, I is *incompatible*.

□

Definition 46 (Conclusive e-interpretation)**.**
We will say that an e-interpretation I of Σ is *conclusive* with respect to query Q of Π if Q is decided in I.

□

Definition 47 (Cut)**.**
Let $T = T_\Pi\langle f\rangle$ be an AI-tree. A tree T_c is *a cut* of T if T_c is a subtree of T such that:

1. the root of T_c is the root of T,
2. each i-node I of T_c has at most one child in T_c, and
3. for each a-node a of T_c, I is a child of a in T_c iff I is a child of a in T.

□

Next we define a special class of subtrees of $T_\Pi\langle f\rangle$, called solution trees w.r.t Q, which, as we will later show, can be used to compute the probability of Q.

Definition 48 (f-solution tree with respect to Q)**.**
A tree T_{sc} is an *f-solution tree*, or just an *f-solution*, of Π with respect to query Q if there is an AI-tree $T_\Pi\langle f\rangle$ such that T_{sc} is a cut of $T_\Pi\langle f\rangle$ and every leaf node of T_{sc} is either incompatible or conclusive with respect to Q. □

Definition 49 (Solution tree w.r.t. Q)**.**
A tree T_{sc} is a *solution tree*, or just a *solution*, of Π w.r.t. Q if T_{sc} is an f-solution of Π w.r.t. Q for some consequence function f.

□

The following proposition explains how to compute the probability of a query Q given a solution tree with respect to Q.

Proposition 5. Given a program Π from $\mathcal{B}$, a query Q of Π and a solution tree S of Π with respect to Q, let $\mathcal{L}$ be the set of compatible leaves of S, and $\mathcal{L}_Q$ be the subset of $\mathcal{L}$ such that each member of $\mathcal{L}_Q$ satisfies Q. We have:

1. if P_Π is defined, then
$$P_\Pi(Q) = \frac{\sum\limits_{I\in\mathcal{L}_Q} \hat{\mu}_\Pi(I)}{\sum\limits_{I\in\mathcal{L}} \hat{\mu}_\Pi(I)} \tag{5.4}$$
2. otherwise,
$$\sum_{I\in\mathcal{L}} \hat{\mu}_\Pi(I) = 0 \tag{5.5}$$

□

The efficiently of the search of a solution depends significantly on the procedure which checks if a node is incompatible. In section 5.3.4 we will define sufficient conditions on a consequence function which allow to perform these checks in linear time. Another difficulty of the computation is related to the computation of measure $\hat{\mu}$ for compatible leaves of a solution. Section 5.3.5 describes a method for doing this efficiently. In section 5.3.6 we will describe a special class of solutions which can be used to compute the probability of a query by doing the incompatibility checks as described in 5.3.4 and computing the measures using the method from 5.3.5. We will also discuss the conditions under which there exists such a solution of Π w.r.t Q there. In addition, in section 5.3.7 we give examples of several consequence functions that can be used to produce such solutions and discuss their impact on the size of the solution trees.

5.3.4 Detecting Incompatible Nodes Efficiently

In this section we will define sufficient and necessary conditions for a node of an AI-tree to be incompatible that can be checked efficiently. We will start from defining a special class of consequence functions. As before, let Π be a program from class $\mathcal{B}$ with signature Σ. The set of e-literals *conflicting* with activity record $do(a = y)$, denoted by $conf(do(a = y))$, is

$$\{not\ do(a = y) = true, not\ do(a = y) \neq false, do(a = y) = false, do(a = y) \neq true\}.$$

The set of e-literals *conflicting* with an observation $obs(l)$ is

$$\{not\ obs(l) = true, not\ obs(l) \neq false, obs(l) = false, obs(l) \neq true\}.$$

For an activity record r, $conf(r)$ denotes the set of e-literals conflicting with r.

Definition 50 (Admissible consequence function)**.**

Let AR be the set of activity records of Π. Let Π' be the program $\Pi \setminus AR$. Let $f' : int(\Sigma) \rightsquigarrow int(\Sigma)$ be a consequence function of Π'. We describe partial function $f : int(\Sigma) \rightsquigarrow int(\Sigma)$ determined by Π' and f'. Consider $T = \{truly\_random(a) : \exists y\ do(a = y) \in AR\}$ and let L_{tr} be the set of literals formed by attribute terms from T. Let TU be the set of e-literals:

$$TU = \{not\ l \mid l \in L_{lr}\}.$$

Let AR_{NOT} be the set of e-literals:

$$AR_{NOT} = \bigcup_{rec \in AR} conf(rec)$$

We will define $f(I)$ for every interpretation I such that:

1. $I \cap L_{tr} = \emptyset$
2. $I \cap AR_{NOT} = \emptyset$
3. $f'(I \setminus (TU \cup satr(AR)))$ is defined,

as follows:

$$f(I) = satr((f'(I \setminus (TU \cup satr(AR)) \setminus L_{tr} \setminus AR_{NOT}) \cup TU \cup AR)$$

We will say that f is *an admissible consequence function* w.r.t Π *induced by* f'.

□

Proposition 6. Let f be an admissible consequence function of program Π from $\mathcal{B}$. We have f is a consequence function of Π.

□

For a program Π, by $\mathcal{X}(\Pi)$ we will denote the collection of general axioms of Π of each of the forms:

$$\leftarrow do(a = y), not\ a = y$$

$$\leftarrow obs(l), not\ l$$

$$\leftarrow do(a = y), not\ random(a, p_1), \ldots, not\ random(a, p_n).$$

Definition 51 (Definite node)**.**

We will say that an e-interpretation I of Σ is *definite* (w.r.t Π) if

1. I falsifies the body of every axiom in $\mathcal{X}(\Pi)$, or

2. I satisfies the body of some axiom in $\mathcal{X}(\Pi)$.

□

Finally, we have the following property for incompatible AI-tree nodes.

Proposition 7. Let f be an admissible consequence function of Π and I be a definite i-node of $T_\Pi\langle f\rangle$. I is incompatible iff there exists an axiom in $\mathcal{X}(\Pi)$ whose body is satisfied by I.

□

5.3.5 Computing Node Measures Efficiently

In order to use Proposition 5 to compute $P_\Pi(Q)$, we need to compute the measures of compatible leaf nodes efficiently. In this section we will define a special function $\hat{\mu}^*$ on e-interpretations, called a candidate measure, which can be computed efficiently, and show that under certain conditions $\hat{\mu}^*(I) = \hat{\mu}(I)$. The definitions for $\hat{\mu}^*$ will mostly repeat the definitions of unnormalized measures $\hat{\mu}$ of possible worlds from the P-log semantics, with the normal satisfiability being replaced with satisfiability for e-interpretations.

Let I be an e-interpretation satisfying conditions 4 - 6 as defined below.

Condition 4 (Unique selection rule for e-interpretations).
If Π contains two rules r_1 and r_2 of the forms

$$random(a : \{X : p_1(X)) \leftarrow B_1$$

and

$$random(a : \{X : p_2(X)) \leftarrow B_2$$

then I does not satisfy at least one of the bodies B_1 and B_2.

□

Condition 5 (Unique probability assignment for e-interpretations).
If Π contains a random selection rule

$$random(a(\bar{t}) : \{Y : p(Y)\}) \leftarrow B$$

along with two different probability atoms

$$pr(a(\bar{t}) \mid B_1) = v_1 \text{ and } pr(a(\bar{t}) \mid B_2) = v_2$$

then I does not satisfy at least one of bodies B, B_1, and B_2. □

Condition 6 (No probabilities assigned outside of dynamic range for e-interpretations).
If Π contains a random selection rule

$$random(a(\bar{t}) : \{Y : p(Y)\}) \leftarrow B_1$$

along with probability atom

$$pr(a(\bar{t}) = y \mid B_2) = v$$

then if I satisfies B_1 and B_2 then I satisfies $p(y)$.

□

For every atom $a = y$ in I such that

$$y \in PO(I, r, a) \text{ for some random selection rule } r : random(a, p) \leftarrow B \text{ of } \Pi \text{ such that } B \subseteq I, \quad (5.6)$$

$$truly\_random(a) \in I, \quad (5.7)$$

$$\text{for every pr-atom } pr(a = y_1 \mid B) = v \text{ of } \Pi, \text{ either } B \subseteq I, \text{ or } B \text{ is falsified by } I, \quad (5.8)$$

$$\text{if there is a random selection rule } random(a, p) \leftarrow B \text{ s.t. } B \subseteq I, \text{ then for every } y \in range(a),\ p(y) \text{ is decided in } I, \quad (5.9)$$

we will define the corresponding causal probability $P(I, a = y)$. Whenever possible, the probability of an atom $a = y$ will be directly assigned by pr-atoms of the program and denoted by $PA(I, a = y)$. To define probabilities of the remaining atoms we will use the principle of indifference, as we did for defining probabilities for possible worlds. The probabilities of those remaining atoms will be denoted by $PD(I, a = y)$. (PA stands for *assigned probability* and PD stands for *default probability*).

More precisely, for each atom $a = y$ in I satisfying conditions (5.6) - (5.9) we have:

1. Assigned probability:

   If Π contains $pr(a = y \mid B) = v$, I guarantees B, then

   $$PA(I, a = y) = v$$

   (note that Condition 5 implies that $PA(I, a = y)$ is uniquely defined).

2. Default probability:

   Let $A_a(I) = \{y \mid PA(I, a = y) \text{ is defined}\}$, and y be a member of $PO(I, r, a)$ such that $y \notin A_a(I)$. Then let

$$\alpha_a(I) = \sum_{y \in A_a(I)} PA(I, a = y)$$

$$\beta_a(I) = |\{y \mid y \in PO(I, r, a) \text{ is possible in } I \text{ and } y \notin A_a(I)\}|$$

$$PD(I, a = y) = \frac{1 - \alpha_a(I)}{\beta_a(I)}.$$

3. Finally, the causal probability $P(I, a = y)$ is defined by:

$$P(I, a = y) = \begin{cases} PA(I, a = y) & \text{if } y \in A_a(I) \\ PD(I, a = y) & \text{otherwise.} \end{cases}$$

Definition 52 (E-interpretation's candidate measure)**.**
Let I be an e-interpretation of Π and $D = \{a = y \in I \mid P(I, a = y) \text{ is defined}\}$. The *candidate measure*, $\hat{\mu}^*_\Pi(I)$, of I *induced by* Π is

$$\hat{\mu}^*_\Pi(I) = \prod_{a=y \in D} P(I, a = y).$$

□

When the program Π is clear from the context we may simply write $\hat{\mu}^*$ instead of $\hat{\mu}^*_\Pi$.

Notice that, the intuitive meaning of the candidate measure for e-interpretation may not be immediately clear. In what follows, we introduce a class of e-interpretations with a useful property: the candidate measures of such e-interpretations are equal to their corresponding measures.

Definition 53 (Informative e-interpretation)**.**

An e-interpretation I of Σ is *informative* iff

$$\hat{\mu}^*(I) = \hat{\mu}(I)$$

□

Example 10. Consider the program Π_{11}:

```
a,b:boolean.
random(a).
pr(a) = 0.3.
```

The e-interpretation $I = \{a, random(a), truly\_random(a)\}$ is informative. There is a unique possible world W of Π_{11} compatible with I, and

$$\hat{\mu}^*(I) = \hat{\mu}(I) = \hat{\mu}(W) = 0.3.$$

□

Clearly, not every e-interpretation is informative.

Example 11. Consider the program Π_{12}:

```
a,b:boolean.
random(a).
random(b) :- a.
pr(b) = 0.3.
```

The e-interpretation

$$I = \{\neg a, random(a), truly\_random(a), b, random(b), truly\_random(b)\}$$

is not informative. There are no possible worlds compatible with I, so $\hat{\mu}(I) = 0$. However,

$$\hat{\mu}^*(I) = P(\neg a, I) \cdot P(b, I) = 0.5 \cdot 0.7 = 0.35.$$

□

The following proposition gives sufficient conditions for a node to be informative.

Proposition 8. Let Π be a program from $\mathcal{B}$, f be an admissible consequence function of Π, $T_\Pi\langle f\rangle$ be an AI-tree of Π and I be an i-node of $T_\Pi\langle f\rangle$. If

1. I is compatible and definite, and
2. for every random attribute term a decided in I, $truly\_random(a)$ is decided in I

then I is informative (see definition 53).

□

5.3.6 Efficient Solutions

In this section we will describe a special class of solutions which can be used to compute the probability of the query efficiently, using the results of the previous two sections and the formula from Proposition 5.

Definition 54 (Efficient solution)**.**
Let Π be a program from $\mathcal{B}$ with signature Σ, f a consequence function of Π, Q a query of Π and S a cut of $T_\Pi\langle f\rangle$ that is a solution of Π w.r.t Q. S is *efficient* iff:

1. f is an admissible consequence function,
2. every leaf of S is definite, and
3. every compatible leaf of S is informative.

□

Suppose S is an efficient solution. Since every leaf is definite, and it is a cut of a tree parameterized by an admissible consequence function, incompatible leaves can be efficiently found using the results from Proposition 7. The measure of the informative compatible leaves of S can be efficiently computed using the results from Proposition 8, Definition 53 and the formulas for candidate measures.

When searching for a solution, we will look for cuts of an AI-tree parameterized by an admissible consequence function whose leaves are either:

1. definite and incompatible, or
2. definite, conclusive w.r.t given Q and informative (i.e, deciding $truly\_random(a)$ for each decided random attribute term a).

We will refer to such leaves as *final* w.r.t. Q.

The existence of final solutions in an AI-trees obtained from certain consequence functions is guaranteed by Proposition 12.

5.3.7 Consequence Functions

The efficiency of the algorithm depends on the number of nodes of $T_\Pi\langle f\rangle$ visited before a solution tree is found. The choice of f largely impacts the efficiency.[3] Usually, the larger consequences are computed by f, the less number of nodes in a tree will be visited. This section discusses the details.

For a given program Π with signature Σ, we will define 3 different admissible consequence functions: $f_1 - f_3$ whose corresponding AI-trees contain efficient solutions. Function f_1 computes a minimal collection of consequence needed for the correctness of our algorithm. Functions f_2 and f_3 are the refinements of f_1 which are designed to compute more consequences. Function f_2, defined using f_1, allows to obtain larger consequences than f_1. Function f_3, defined using f_2, allows to compute consequences of some interpretations for which f_2 is undefined.

Before describing the functions, we will introduce some notation and abbreviations.

By $RT(\Pi)$ we will denote the set of all random attribute terms of Π.

[3]another important factor is the order in which the tree is explored, affected by the heuristics used to choose an attribute term (see section 5.4)

Let I be an e-interpretation of Π. In what follows we use the following abbreviations:

- $DRT_{\Pi}(I)$ – the set of random attribute terms of Π decided in I.
- $NRT_{\Pi}(I)$ – the set of non-random attribute terms of Π such that:
  - each member of $NRT_{\Pi}(I)$ does not depend on $RT(\Pi) \setminus DRT_{\Pi}(I)$ in $red(\Pi)$, and
  - for every member of $NRT_{\Pi}(I)$ of the form $random(a, p)$, $a \in DRT_{\Pi}(I)$ and $p(x) \in NRT_{\Pi}(I)$ for every $x \in range(a)$,
  - for every member of $NRT_{\Pi}(I)$ of the form $truly\_random(a)$ we have

$$random(a, p) \in NRT_{\Pi}(I)$$

for every attribute term of the form $random(a, p)$ of Σ.

We will sometimes omit Π from $DRT_{\Pi}(I)$, $NRT_{\Pi}(I)$ when the program is clear from the context. When listing the attribute terms of $NRT(I)$, will will sometimes omit (some) special attribute terms.

Example 12. Consider the program Π_{13}:

```
a,b,f,h: #boolean.
random(a).
random(b).
f:- a.
h:- a,b.
```

and e-interpretations:

$$I_1 = \{random(a), a\}, \text{ and}$$
$$I_2 = \{random(a), a, random(b), b\}$$

Clearly, we have:

$$RT(\Pi_{13}) = \{a, b\},$$
$$DRT(I_1) = \{a\},$$
$$DRT(I_2) = \{a, b\}.$$

Attribute term f depends only on a, and h depends on both a and b. The attribute terms formed by $truly\_random$ and $random$ are included only with the corresponding random attributes. So,

$$NRT(I_1) = \{random(a), truly\_random(a), f\},$$
$$NRT(I_2) = \{random(a), truly\_random(a), random(b), truly\_random(b), h, f\}.$$

□

For a partial interpretation I, by $ENC(I)$ (read "the encoding of I") we will denote a collection of rules obtained from I as follows:

1. for every atom $a = y \in I$, $ENC(I)$ contains fact $a = y$,

2. for every e-literal $not\ l \in I$, $ENC(I)$ contains a constraint $\leftarrow l$, and

3. for every literal of the form $a \neq y$ in I, $ENC(I)$ contains two constraints:

$$\leftarrow a = y$$

   and

$$\leftarrow not\ a = y_1, \ldots, not\ a = y_k$$

   where $range(a) = \{y_1, \ldots, y_k\}$.

We will define functions $f_1 - f_3$ below.

1. Given an e-interpretation I, let $\Pi_{cons}(I)$ be a P-log program consisting of:

- all the declarations of Π,
- every rule r of $red(\Pi)$ such that:
  - the head of r is not formed by a random attribute term,
  - every e-literal occurring in r is formed by an attribute term from $NRT(I) \cup DRT(I)$,
  - r is not an activity record,
  - r is not of the form

$$truly\_random(a) \leftarrow random(a, p), \quad not\ do(a, y_1), \dots, not\ do(a, y_k)$$

    where for some $y \in \{y_1, \dots, y_k\}$, $do(a, y) \in \Pi$.

Rather than defining the function directly, we first define an auxiliary function f_1', a consequence function for the program obtained from Π by removing activity records. For every interpretation such that $\Pi_{cons}(I) \cup ENC(I)$ has a unique possible world W we have:

$$f_1'(I) = I \cup satr(W \cup \bigcup_{a \in A} a = u)$$

where

$$A = \{a \mid a \in NRT(I) \text{ and } W \text{ does not contain atoms formed by } a\}.$$

We define f_1 to be the consequence function of Π induced by f_1' (see Definition 50).

Example 13. Consider the program Π_{14}

```
a,b,h,q,g: #boolean.
```

```
random(a).
random(b).
g :- a.
h :- a.
h :- b.
q:- a,b.
```

and two interpretations

$$I_1 = \{a, random(a), truly\_random(a)\}$$

and

$$I_2 = \{\neg a, random(a), truly\_random(a)\}$$

We will compute $f_1(I_1)$ and $f_1(I_2)$.

$NRT(I_1) = \{g\}$, $DRT(I_1) = \{a\}$. $\Pi_{cons}(I_1)$ contains the rule $g \leftarrow a$, general axioms with no random attribute terms in the head, and the random selection rule $random(a)$.

$ENC(I_1)$ contains rules:[4]

$$a.$$
$$\leftarrow \neg a.$$
$$\leftarrow a \neq true.$$
$$\leftarrow not\ a, not\ \neg a.$$

Note that the last 3 rules were obtained from the e-literals: $a \neq false, not\ a = false, not\ a \neq true$, that are present in I_1 due to the saturation of a.

[4]Note that there are also rules for e-literals of I_1 formed by special attribute terms, such as $random(a)$, however they are omitted here and in the next examples.

$\Pi_{cons}(I_1) \cup ENC(I_1)$ has a unique possible world $W_1 = \{a, g\}$,

$$f_1(I_1) = f_1'(I_1) = I_1 \cup \{g\}$$

(modulo e-literals not containing special attribute terms).

$NRT(I_2)$, $DRT(I_2)$ and $\Pi_{cons}(I_2)$ are the same as $NRT(I_1)$, $DRT(I_1)$ and $\Pi_{cons}(I_1)$ respectively. $ENC(I_2)$ contains rules:

$$\begin{aligned} \neg a&. \\ &\leftarrow a. \\ &\leftarrow a \neq false. \\ &\leftarrow not\ a, not\ \neg a. \end{aligned}$$

$\Pi_{cons}(I_2) \cup ENC(I_2)$ has a unique possible world $W_2 = \{\neg a\}$, and

$$f_1(I_2) = f_1'(I_2) = I_2 \cup g = u$$

(modulo e-literals not containing special attribute terms)

□

2. To define the second function, f_2, we need some definitions. Let $nr(\Pi)$ be the set of rules of Π, whose heads are not formed by random attribute terms and are not activity records. We will define three functions $N : 2^{e\text{-}lit(\Sigma)} \rightarrow 2^{e\text{-}lit(\Sigma)}$, $H : 2^{e\text{-}lit(\Sigma)} \rightarrow 2^{e\text{-}lit(\Sigma)}$ and $least : int(\Sigma) \rightarrow 2^{e\text{-}lit(\Sigma)}$.

   - $N(L)$ is the set of e-literals of the form $not\ a = y$, such that
     - a is a non-random attribute term, and

  - the body of every rule whose head is $a = y$ contains a literal contrary to some literal from L.

- $H(L) = satr(L \cup \{head(r) \mid r \in nr(\Pi), body(r) \subseteq L\} \cup N(L))$.
- To define $least(I)$, we first state the following proposition:

  Proposition 9. For every e-interpretation I of Σ, there exists a fixed point X of H such that

  (a) $I \subseteq X$,

  (b) no fixed point of H is a proper subset of X, and

  (c) no other fixed point of H satisfies conditions (a), (b).

  We will refer to X satisfying conditions (a) - (c) as *the least fixed point of H relevant to I.*

  □

  We define $least(I)$ to be the least fixed point of H relevant to I.

As in case with f_1, we next define an auxiliary function f'_2, a consequence function for Π without activity records.

If $least(I)$ is not consistent (we can show that, if this is the case, then no possible world of Π is compatible with I), or $f'_1(I)$ is undefined, then $f'_2(I)$ is undefined, otherwise

$$f'_2(I) = least(f'_1(I)).$$

We define f_2 to be the consequence function of Π induced by f'_2.

Example 14. Consider the program Π_{14} and interpretation I_1 and I_2 from example 13. We will compute $f_2(I_1)$ and $f_2(I_2)$.

We first compute $f_2(I_1)$. From example 13, we have

$$f'_1(I_1) = \{a, g\}.$$

It can be shown that $I_F^1 = \{a, g, h\}$ is the least fixed point of H relevant to $f_1'(I_1)$. To see that I_F^1 is a fixed point, notice that $\{head(r) \mid r \in rgn(\Pi), body(r) \subseteq I_F^1\} = \{g, h\}$, and $N(I_F^1) = \{\}$. Therefore,

$$\begin{aligned} H(I_F^1) &= satr(I_F^1 \cup \{head(r) \mid r \in nr(\Pi), body(r) \subseteq I_F^1\} \cup N(I_F^1)) \\ &= satr(I_F^1 \cup \{g, h\} \cup \{\}) \\ &= \{a, g, h\} \\ &= I_F^1, \text{ and} \end{aligned}$$

$least(f'(I_1))$ is I_F^1. So,

$$\begin{aligned} f_2(I_1) &= f_2'(I_1) \\ &= least(f_1'(I_1)) \\ &= I_F^1 \\ &= \{a, g, h\}. \end{aligned}$$

The equalities above are modulo e-literals not formed by special attribute terms.

We now compute $f_2(I_2)$. From Example 13, we have:

$$f_1'(I_2) = \{\neg a, not\ g, not\ \neg g\}$$

It can be shown that $I_F^2 = \{\neg a, not\ \neg g, not\ \neg g, not\ q, not\ \neg q\}$ is the least fixed point of H relevant to $f_1'(I_2)$. To see that I_F^2 is a fixed point, notice that $\{head(r) \mid r \in nr(\Pi), body(r) \subseteq I_F^2\} = \{\}$, and $N(I_F^2) = (q = u \cup g = u)$. So,

$$
\begin{aligned}
H(I_F^2) &= satr(I_F^2 \cup \{head(r) \mid r \in rgn(\Pi), body(r) \subseteq I_F^2\} \cup N(I_F^2)) \\
&= satr(I_F^2 \cup \{\} \cup q = u \cup g = u) \\
&= \{a\} \cup q = u \cup g = u \\
&= I_F^2, \text{ and}
\end{aligned}
$$

$least(f_1'(I_2))$ is I_F^2. So,

$$
\begin{aligned}
f_2(I_2) &= f_2'(I_2) \\
&= least(f_1'(I_2)) \\
&= I_F^2 \\
&= \{a\} \cup q = u \cup g = u
\end{aligned}
$$

The equalities above are modulo e-literals not formed by special attribute terms. Notice that $f_1(I_1) \subsetneq f_2(I_1)$ and $f_1(I_2) \subsetneq f_2(I_2)$, so, f_2 computes more consequences than f_1 for these two interpretations. □

3. The third function, f_3, expands the domain of f_2.

   In order to define f_3, we first define 3 auxiliary functions: $G : int(\Sigma) \times 2^{e\text{-}lit(\Sigma)} \to 2^{e\text{-}lit(\Sigma)}$, $pc : int(\Sigma) \to 2^{e\text{-}lit(\Sigma)}$ and $most : int(\Sigma) \to 2^{e\text{-}lit(\Sigma)}$

   - We define G first:

$$
\begin{aligned}
G(I, J) = satr(J \cup \{head(r) \mid r \in \Pi_{cons}(I), body^+(r) \subseteq J \\
\text{and } body^-(r) \text{ is not falsified by } I\})
\end{aligned}
$$

By $G_I : 2^{e\text{-}lit(\Sigma)} \to 2^{e\text{-}lit(\Sigma)}$ we will denote the function obtained from G by fixing the value of its first argument:

$$G_I(J) = G(I, J).$$

- To define $pc(I)$, we first state the following proposition:

  Proposition 10. For every e-interpretation I of Σ, there exists a fixed point X of G_I such that:

  (a) $I \subseteq X$,

  (b) no fixed point of G_I is a proper subset of X, and

  (c) no other fixed point of G_I satisfies conditions (a), (b).

  We will refer to X satisfying conditions (a) - (c) as *the least fixed point of G_I relevant to I.*

  □

  We define $pc(I)$ to be the least fixed point X of G_I relevant to I.

- We next define *most*:

$$most(I) = \{not\ a = y \mid a \in NRT(I), y \in range(a), a = y \notin pc(I)\}$$

We define $f_3'(I)$, an auxiliary consequence function for Π with activity records removed.

If $f_2'(I)$ is defined, then

$$f_3'(I) = f_2'(I),$$

otherwise, if $I_m = satr(I \cup most(I))$ and $least(I_m)$ are both consistent, then

$$f_3'(I) = least(I_m)$$

In all other cases, $f'_3(I)$ is undefined. Finally, f_3 is defined as the consequence function of Π induced by f'_3.

Example 15. Consider the following program Π_{15}:

```
a,b,g,h,q: boolean.
random(a).
random(b).
g:- not h,a.
h:- not g,a.
q:- a.
:- b,h.
:- -b,h.
```

and interpretation $I_3 = \{a\}$. We first show that both $f_1(I_3)$ and $f_2(I_3)$ are undefined. $NRT(I_3) = \{g, h, q\}$, $DRT(I_1) = \{a\}$, $\Pi_{cons}(I_1)$ consists of rules:

$$g \leftarrow not\ h, a.$$

$$h \leftarrow not\ g, a.$$

$$random(a).$$

and general axioms not shown here. $ENC(I_1)$ contains rules:

$$a.$$

$$\leftarrow \neg a.$$

$$\leftarrow a \neq true.$$

$$\leftarrow not\ a, not\ \neg a.$$

$\Pi_{cons}(I_1) \cup ENC(I_1)$ has two possible worlds:

$$W_1 = \{a, q, g\}$$

and

$$W_2 = \{a, q, h\}.$$

Therefore, $f_1'(I_3)$ and $f_2'(I_3)$ are undefined.

We now compute $f_3'(I_3)$. It can be shown that

$$I_F^3 = \{a, g, h, q\}$$

is the least fixed point of G_{I_3} relevant to I_3. To see that I_F^3 is a fixed point of G_{I_3}, notice that

$$NRT(I_3) = \{h, g, q\}$$

$$DRT(I_3) = a.$$

$\Pi_{cons}(I_3)$ contains rules:

$$g \leftarrow not\ h, a.$$

$$h \leftarrow not\ g, a.$$

$$q \leftarrow a.$$

(and, as before, general axioms not shown here.)

The set $\{head(r) \mid r \in \Pi_{cons}(I_3), body^+(r) \subseteq I_F^3, body^-(r)$ is not falsified by $I_3\}$ is $\{g, h, q\}$. So,

$$\begin{aligned} G_{I_3}(I_F^3) &= satr(I_F^3 \cup \{head(r) \mid r \in \Pi_{cons}(I_3), body^+(r) \subseteq I_F^3 \\ &\qquad \text{and } \ body^-(r) \text{ is not falsified by } I_3\}) \\ &= satr(I_F^3 \cup \{g, h, q\}) \\ &= satr(I_F^3) \\ &= I_F^3 \end{aligned}$$

Hence,

$$pc(I_3) = I_F^3 = \{a, g, h, q\}.$$

Next,

$$\begin{aligned} most(I_3) &= \{not\ a = y \mid a \in NRT(I_3)\ and\ y \in range(a) \\ &\qquad and\ a = y \notin pc(I_3)\} \\ &= \{not\ a = y \mid a \in \{g, h, f\}\ and\ y \in range(a) \\ &\qquad and\ a = y \notin \{a, g, h, f\}\} \\ &= \{\}. \end{aligned}$$

Finally,

$$\begin{aligned} I_m &= satr(I_3 \cup \{\}) \\ &= I_3 \\ &= \{a\}. \end{aligned}$$

It can be shown that $least(I_m)$ is $\{a, f\}$. Therefore,

$$\begin{aligned} f_3(I_m) &= f_3'(I_m) \\ &= least(I_m) \\ &= \{a, q\} \end{aligned}$$

(as before, the last equalities are modulo e-literals not formed by special attribute terms). As we have demonstrated, $f_2(I_3)$ is undefined, while $f_3(I_3)$ is an interpretation. So, f_3 extends the domain of f_2.

□

The following propositions guarantee that the functions we have constructed indeed satisfy the intended properties.

Proposition 11. Let Π be an arbitrary program from $\mathcal{B}$ with signature Σ. f_1, f_2 and f_3 are admissible consequence functions of Π.

□

The existence of an efficient solution with respect to Q in an AI-tree parameterized by f_1, f_2, f_3 is claimed by the following proposition.

Proposition 12. Let Π be a program from $\mathcal{B}$, and T is one of the AI-trees in $\{T_\Pi\langle f_1\rangle, T_\Pi\langle f_2\rangle, T_\Pi\langle f_3\rangle\}$. For every query Q of Π, there exists a cut of T which is an efficient solution of Π w.r.t Q.

□

More Examples.

We will now show how the choice of consequence function affects the size of solution trees (and, therefore, possibly, the efficiency of the algorithm which searches for such trees to compute probabilities).

Example 16. Consider the program Π_{14} from Example 13. We will consider two queries: q and h.

An efficient f_1- solution of Π_{14} w.r.t. h (which is also a solution w.r.t. q) is shown in Figure 5.2. The a-nodes are shown as squares, and the i-nodes are ovals. When writing an interpretation in i-nodes, we will use a shorthand $l_1, \dots, l_n, a_1 = u_1, \dots, a_m = u$ to denote the set of literals $\{l_1, \dots, l_n\} \cup a_1 = u \cup \dots \cup a_m = u$.

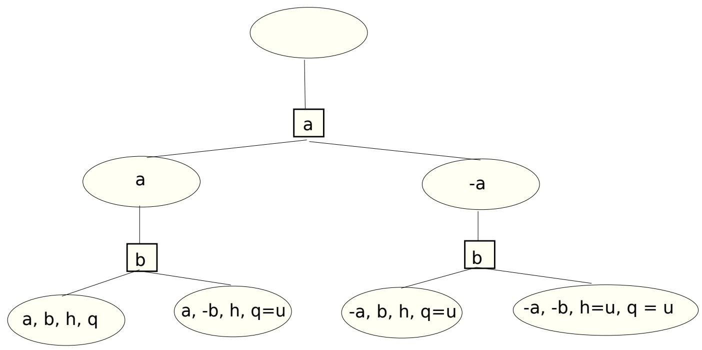

Figure 5.2: f_1-solution of Π_{14} w.r.t to query h (and q)

For any consequence function f, f-solutions differ by the order in which a-nodes were selected. In the tree from Figure 5.2, attribute a was selected first. Another f_1-solution can be constructed by selecting b first and adding it to the root. In can be shown that all f_1 solutions of Π_{14} will consist of 10 nodes. However, f_2 - solutions w.r.t queries h and q may contain less than 10 nodes. f_2 solutions w.r.t queries h and q are shown in Figures 5.3 and 5.4 respectively.

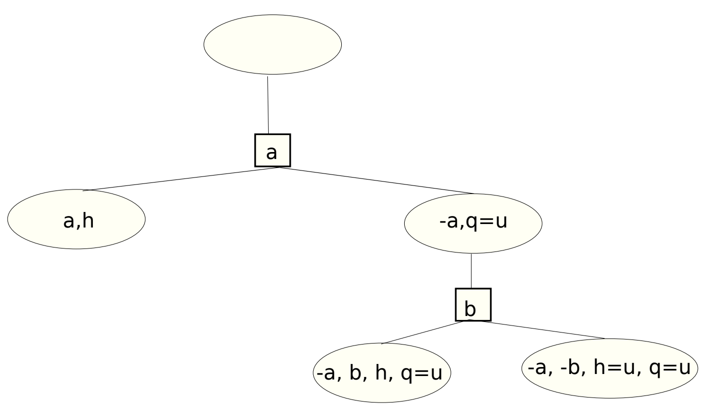

Figure 5.3: f_2-solution of Π_{14} w.r.t. h

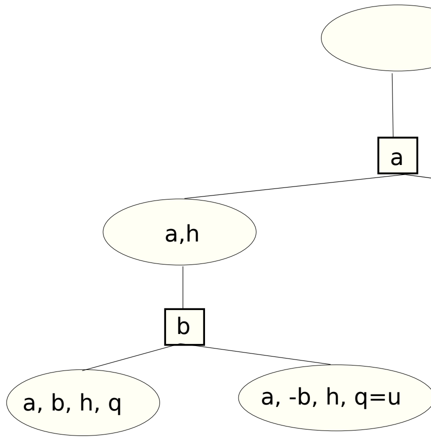

Figure 5.4: f_2-solution of Π_{14} w.r.t. q

It is easy to check that all the leaves of the tree on Figure 5.3 (Figure 5.4) are final w.r.t. corresponding query. Π_{14} has no actions and observations, and every attribute is decided in every leaf node. In this case, the size of f_2-solution depends on the attribute term which was selected first. All f_2-solutions of Π with b being the child of the root have 10 nodes, unlike those from Figures 5.3 and 5.4.

Example 17. This example demonstrates the difference between f_3 and f_2. Consider program Π_{15} from Example 15. Figure 5.5 below shows an efficient f_3 - solution w.r.t query q.

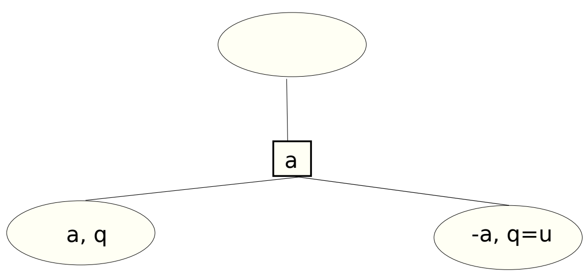

Figure 5.5: f_3-solution of Π_{15} w.r.t. f

It can be checked that every f_2-solution and f_1 solution of Π_{15} w.r.t q has at least 10 nodes, while f_3-solution from Figure 5.5 only has 4 nodes. Even though Π_{15} has only one probabilistic causal leveling: $V = [b, a]$, f_3 is able to compute consequences even in case the first selected attribute term (a) does not form a prefix of V, while f_1 and f_2 are not able to do that.

5.4 Algorithm Description and Implementation

In this section we will describe an algorithm that computes the probability $P_\Pi(Q)$ of query Q of program Π from class $\mathcal{B}$. In addition to Π and Q, the algorithm will accept an admissible consequence function $f : int(\Sigma) \rightarrow int(\Sigma)$ of Π such that:

for an arbitrary query Q of Π, there exists an f-solution of Π w.r.t Q

Examples of such functions for an arbitrary query are given in Section 5.3.7 (functions f_1 - f_3).

We first describe the main routine. The routine consists of two parts: finding the f-solution and computing the probability from its leaves.

Function 1 Probability

Input: Π: a program from $\mathcal{B}$ with signature Σ
Q: query of Π
f: an admissible consequence function of Π s.t. there is an f-solution of Π w.r.t Q

Output: A rational number or $False$, where
- if P_Π is defined, return $P_\Pi(Q)$
- otherwise, return $False$

Vars: T: a cut of $T_\Pi\langle f\rangle$
P, P_Q: rational numbers
I: e-interpretation of Σ

$P := 0$
$P_Q := 0$
% *Note that* $GetSolution(\Pi, Q, f)$ *returns an efficient* f*-solution w.r.t* Q *due to the condition on* f
$T := GetSolution(\Pi, Q, f)$
for every leaf I of T s.t. I does not falsify any activity record of Π **do**
    $P := P + \hat{\mu}^*_\Pi(I)$
    if I satisfies Q **then**
        $P_Q := P_Q + \hat{\mu}^*_\Pi(I)$
if $P = 0$ **then**
    return $False$
else
    return P_Q/P

When searching for the solution, we will start from a cut consisting of a single node containing e-interpretation $f(\{\})$, and gradually extend the cut until a solution is found (that is, every interpretation in its leaves is final w.r.t Q). The extension of the tree is done by selecting a leaf node containing interpretation I and an attribute term a ready in I and forming descendants containing assignments of possible values to a.

If the extension of some leaf N is not possible, the search continues from an ancestor A of N by removing all $A's$ current descendants and adding new descendants, starting with an attribute term ready in A which has not been tried to extend A yet. In order to avoid skipping some of the solutions, A is chosen to be the closest ancestor of N which is possible to extend.

In order to keep track of the attribute terms which have not been explored yet from a given node, we will extend each i-node with a set of such attributes. We will refer to extended nodes as ei-nodes and to the new trees as candidate trees.

Definition 55 (Ei-node)**.**
Let I be an e-interpretation of Σ and A be a subset of attribute terms of Σ ready in I. We will refer to the pair $\langle I, A \rangle$ as an *ei-node* of Σ.

□

Definition 56 (Candidate tree)**.**
Let T_{cut} be an arbitrary cut of the AI-tree of Π parameterized by f. Let T_{cand} be a tree obtained from T_{cut} by replacing every i-node I with an ei-node $\langle I, A \rangle$ of the signature of Π. We will say that T_{cand} is a *candidate tree* which *represents* T_{cut}. For a candidate tree T by $repr(T)$ we will denote the tree represented by T.

□

We will next define leaves that need to be extended in order to find a solution.

Definition 57 (Open leaf)**.**
Let T be a candidate tree of Π. We will say that a leaf node N of T is *open* if $N = \langle I, A \rangle$ is an ei-node such that I is not final w.r.t Q.

□

The set of *u-terms* of Σ is the union of terms of Σ and $\{u\}$.

Function 2 GetSolution

Input: Π: program from $\mathcal{B}$ with signature Σ
Q: query of Π
f: admissible consequence function of Π

Output: efficient f-solution of Π w.r.t Q, if such a solution exists
$False$, otherwise

Vars: T: candidate tree of Π
$\langle A, I\rangle, \langle A', I'\rangle$: ei-nodes of Σ
N : a-nodes of Σ
a : attribute term of Σ
y : u-term of Σ
Y : set of u-terms of Σ

```
1  I := f({})
2  Let A be the set of attribute terms ready in I
3  Let T be the candidate tree consisting of single node ⟨I, A⟩
4  while T has an open leaf do
5      Select an open leaf ⟨I, A⟩ of T
6      if A ≠ ∅ then
7          Select and delete a from A
8          Let Y be the set of possible values of a in I
9          % note that Y is non-empty, see Def. 42 and Def. 39, clause 2 (d)
10         if there is no y ∈ Y such that f(I ∪ a = y) is undefined then
11             Create a-node N with label a
12             Make N a child of ⟨I, A⟩
13             for every y ∈ Y do
14                 I' := f(I ∪ a = y)
15                 Let A' be the set of attribute terms ready in I'
16                 Make ⟨I', A'⟩ a child of N
17     else
18         if there is no ancestor ⟨I', A'⟩ of ⟨I, A⟩ s.t. A' ≠ ∅ then
19             return False
20         else
21             Let ⟨I', A'⟩ be the closest ancestor of ⟨I, A⟩ s.t. A' ≠ ∅
22             Remove all descendants of ⟨I', A'⟩ in T
23 return repr(T)
```

Selection of attribute term. Different solution trees with respect to a given query Q may have different sizes. The way how an attribute term is selected in line 8 of function *GetSolution* affects the size of the solution tree which is produced, and thus, possibly, the efficiency of the algorithm. At this point, we will do a random selection from the set of ready attribute terms.

Implementation. A preliminary implementation of this algorithm for consequence function f_3 is available at `https://github.com/iensen/plog2.0/wiki`. Using an appropriate data structure for storing the current candidate, the implementation only consumes $O(Attrs(\Sigma))$ memory (note that a naive data structure may require memory exponential in $Attrs(\Sigma)$, since the size of candidates trees can be exponential in terms of $Attrs(\Sigma)$).

The implementation performs better than an existing implementation created by Dr. Zhu (based on an ASP solver Smodels [Simons et al., 2002]), described in [Zhu, 2012], on some of the benchmarks, and worse on the others.

As expected, our implementation is faster for programs where the value of the query can be determined from a partial assignment to a small subset of random attribute terms of the program. Consider, for example, the program based on the squirrel problem from [Gelfond & Kahl, 2014][5], where the query depends on a small number of random attribute terms. Table 5.1 shows the performance of both solvers for this example.

Table 5.1: Performance on squirrel example

Query	**Dr.Zhu's Solver**	**My Solver**
found(1)	43s	2.2s
found(2)	44s	2.6s
found(3)	43s	2.2s
found(4)	44s	2.3s
found(20)	41s	4.1s
found(15)	42s	2.2s

[5] P-log programs for my and Dr.Zhu's solvers can be found at `https://github.com/iensen/plog2.0/blob/master/plogapp/tests/squirrel.plog` and `https://github.com/iensen/plog2.0/blob/master/oldplog/Examples/Squirrel/pr.plog` respectively.

However, the following results [6] for the BlockMap problem from [Zhu, 2012], given in Table 5.2, show that, in some cases, the proposed optimization does not give an improvement:

Table 5.2: Performance on block map problem example

Grid Size	**Dr. Zhu's Solver**	**My Solver**
20×1	0.387s	0.400s
20×2	0.949s	0.881s
20×3	2.313s	6.222s
20×4	7.649s	37.630s
20×5	18.728s	165.162s

In this problem, it is often the case that the solution tree construction does not stop until all the random attribute terms were selected. While we expect it to be possible to achieve a better performance of our solver by improving the data structures and the consequence function, we do not expect a significant advantage over the naive approach which computes all the possible worlds.

Finally, below are the results for the programs used to implement safety cases for a recent NASA R&D project, that involve a substantial amount of both logical and probabilistic reasoning. The first program[7] computes the probability of less than 5 components being broken in the system. The performance results for this program are shown in Table 5.3.

Table 5.3: Performance on components failure example

Dr. Zhu's Solver	**My Solver**
17m 44s	1m 36s

[6] P-log programs for my and Dr.Zhu's solvers can be found at `https://github.com/iensen/plog2.0/tree/master/plogapp/tests/weijuns_testsuite/Blocks` and `https://github.com/iensen/plog2.0/tree/master/oldplog/Examples/BlockWorld` respectively.

[7] See `https://github.com/iensen/plog2.0/blob/master/plogapp/tests/nasa/A4n.plog` and `https://github.com/iensen/plog2.0/blob/master/oldplog/Examples/NASA/A4.plog` for mine and Dr.Zhu's system respectively.

The second program[8] computes the probability of system's failure given that less than 5 components of the system are broken. The performance is shown in Table 5.4.

Table 5.4: Performance on system failure example

Dr. Zhu's Solver	**My Solver**
3m 41s	1m 39s

In both of the examples, the query can be decided in a node in which enough components were selected and randomly assigned a failure status.

These results, however, are not conclusive. The implementation is preliminary and the testing is done on a small number of examples. We believe, however, that the results are promising and a good implementation will substantially improve the performance in many interesting cases, and maintain parity with the naive implementation on others. Such an implementation, however, requires a substantial amount of work and is beyond the scope of this dissertation.

[8]See `https://github.com/iensen/plog2.0/blob/master/plogapp/tests/nasa/F.plog` and `https://github.com/iensen/plog2.0/blob/master/oldplog/Examples/NASA/Fprmod.plog` for mine and Dr.Zhu's system respectively.

CHAPTER VI
CONCLUSION AND FUTURE WORK

In this work, we have accomplished the following:

- defined extensions of the language which increase its usability and expressive power,
- defined a new class of P-log programs, $\mathcal{B}$, and proved their coherency,
- designed an inference algorithm for programs from $\mathcal{B}$ and proved its correctness,
- developed a preliminary implementation of the algorithm and showed its advantages on a number of examples.

Future work may include:

- improving the implementation,
- investigating other possible extensions of the language (e.g., with aggregates) and adapting the algorithm for them,
- designing an algorithm which computes an approximation to queries' probabilities (we believe the data structures and other ideas from the current work can be reused for this purpose).

BIBLIOGRAPHY

[Balai & Gelfond, 2017] Balai, E. & Gelfond, M. (2017). Refining and generalizing p-log - preliminary report. In *Proceedings of the 10th Workshop on Answer Set Programming and Other Computing Paradigms co-located with the 14th International Conference on Logic Programming and Nonmonotonic Reasoning, ASPOCP@LPNMR 2017, Espoo, Finland, July 3, 2017.*

[Balai et al., 2013] Balai, E., Gelfond, M., & Zhang, Y. (2013). Towards answer set programming with sorts. In *Logic Programming and Nonmonotonic Reasoning, 12th International Conference, LPNMR 2013, Corunna, Spain, September 15-19, 2013. Proceedings* (pp. 135–147).

[Balduccini, 2012] Balduccini, M. (2012). Answer set solving and non-herbrand functions. In *Proceedings of the 14th International Workshop on Non-Monotonic Reasoning (NMR'2012)(Jun 2012).*

[Balduccini & Gelfond, 2003] Balduccini, M. & Gelfond, M. (2003). Logic programs with consistency-restoring rules. In *International Symposium on Logical Formalization of Commonsense Reasoning, AAAI 2003 Spring Symposium Series* (pp. 9–18).

[Baral, 2003] Baral, C. (2003). *Knowledge representation, reasoning and declarative problem solving*. Cambridge university press.

[Baral et al., 2004] Baral, C., Gelfond, M., & Rushton, N. (2004). Probabilistic reasoning with answer sets. In *International Conference on Logic Programming and Nonmonotonic Reasoning* (pp. 21–33).: Springer.

[Baral et al., 2009] Baral, C., Gelfond, M., & Rushton, N. (2009). Probabilistic reasoning with answer sets. *Theory and Practice of Logic Programming*, 9(01), 57–144.

[Dix et al., 1996] Dix, J., Gottlob, G., & Marek, V. W. (1996). Reducing disjunctive to non-disjunctive semantics by shift-operations. *Fundam. Inform.*, 28(1-2), 87–100.

[Gelfond & Kahl, 2014] Gelfond, M. & Kahl, Y. (2014). *Knowledge representation, reasoning, and the design of intelligent agents: The answer-set programming approach*. Cambridge University Press.

[Gelfond & Lifschitz, 1988] Gelfond, M. & Lifschitz, V. (1988). The stable model semantics for logic programming. In *Logic Programming, Proceedings of the Fifth International Conference and Symposium, Seattle, Washington, August 15-19, 1988 (2 Volumes)* (pp. 1070–1080).

[Gelfond & Lifschitz, 1991a] Gelfond, M. & Lifschitz, V. (1991a). Classical negation in logic programs and disjunctive databases. *New Generation Computing*, 9(3/4), 365–386.

[Gelfond & Lifschitz, 1991b] Gelfond, M. & Lifschitz, V. (1991b). Classical negation in logic programs and disjunctive databases. *New generation computing*, 9(3-4), 365–385.

[Lifschitz, 2008] Lifschitz, V. (2008). What is answer set programming? In *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence, AAAI 2008, Chicago, Illinois, USA, July 13-17, 2008* (pp. 1594–1597).

[Lifschitz & Turner, 1994] Lifschitz, V. & Turner, H. (1994). Splitting a logic program. In *Logic Programming, Proceedings of the Eleventh International Conference on Logic Programming, Santa Marherita Ligure, Italy, June 13-18, 1994* (pp. 23–37).

[Pearl, 2009] Pearl, J. (2009). Causality.

[Sacca & Zaniolo, 1990] Sacca, D. & Zaniolo, C. (1990). Stable models and non-determinism in logic programs with negation. In *Proceedings of the ninth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems* (pp. 205–217).: ACM.

[Simons et al., 2002] Simons, P., Niemelä, I., & Soininen, T. (2002). Extending and implementing the stable model semantics. *Artificial Intelligence*, 138(1-2), 181–234.

[Wellman & Henrion, 1993] Wellman, M. P. & Henrion, M. (1993). Explaining'explaining away'. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(3), 287–292.

[Zhu, 2012] Zhu, W. (2012). *Plog: Its algorithms and applications*. PhD thesis, Texas Tech University.

APPENDIX: PROOFS

A.1 Proofs of Propositions from Chapters I - III

A.1.1 Proof of Proposition 1

Proposition 1. Every possible world W of a program Π satisfies every rule of Π □

Proof. Let W be a possible world of Π and $a = y \leftarrow B$ be the rule of Π. Let N be the subset of B consisting of e-literals having a default negation. If N contains an e-literal *not* l s.t. W does not satisfy *not* l, then B is not satisfied by W, and, r is satisfied by W. Otherwise, there is a rule $a = y \leftarrow B'$ in Π^W, where $B' = B \setminus N$. Now there are only two possibilities:

1. W does not satisfy B'. In this case W does not satisfy B, and W satisfies r
2. W satisfies B'. In this case, since W is a possible world of Π, it satisfies the rules of Π^W, including $a = y \leftarrow B'$. Therefore, $a = y \in W$, and W satisfy r.

□

A.1.2 Proof of Proposition 2

Lemma 1. Let Π be a program, where U is the set of activity records, and Π_2 be a program obtained from Π by removing a rule $b = v \leftarrow l, B$ s.t. l is formed by $do(l)$ or $obs(l)$, and U does not satisfy l. We have $\Omega_{\Pi_2} = \Omega_{\Pi}$.

□

Proof. We will consider two cases:

1. Suppose l is of the form *not* l'. Then, since U does not satisfy l, we have l' is satisfied by U. Let W be a possible world of Π. It is easy to check, that, since W is a possible world of Π, it contains U, and, therefore, l' is satisfied by W. Then we have $\Pi^W = \Pi_2^W$. Thus, clearly, W is the minimal set satisfying Π_2^W, and, a possible world of Π_2. The proof of the other direction ($\Omega_{\Pi_2} \subseteq \Omega_{\Pi}$) is symmetrical (with Π and Π_2 switched).

2. Suppose l is a literal. We first prove $\Omega_{\Pi} \subseteq \Omega_{\Pi_2}$. Let W be a possible world of Π. We will prove:

$$W \text{ is a possible world of } \Pi_2 \tag{A.1}$$

By definition of a possible world, we have:

$$W \text{ satisfies } \Pi^W \tag{A.2}$$

We have

$$\Pi_2^W = (\Pi \setminus r)^W \tag{A.3}$$

From (A.2) and (A.3) we have:

$$W \text{ satisfies } \Pi_2^W \tag{A.4}$$

For the sake of contradiction, suppose there is $W' \subsetneq W$ such that:

$$W' \text{ satisfies } \Pi_2^W \tag{A.5}$$

We have:

$$\Pi^W = \Pi_2^W \cup \{r\}^W \tag{A.6}$$

Clearly, W contains no atoms formed by *do* and *obs* other than those in U (or else, the set obtained from W by removing such atoms would satisfy Π^W, which is a contradiction to the minimality of W). Since the body of R contains a literal not satisfies by U, and $W' \subsetneq W$, we have that W' contains no atoms formed by *do* and *obs* other than those in U, and thus, does not satisfy l. Therefore, if $\{r\}^W$ is not-empty, it contains a literal in the body not satisfied by W'. Therefore, W' satisfies Π^W which is a contradiction to the fact that W is a possible world of Π.

We now prove $\Omega_{\Pi_2} \subseteq \Omega_{\Pi}$. Let W be a possible world of Π_2. Since W does not satisfy a literal in the body of r, we have that W satisfies $\{r\}^W$. Therefore, W satisfies $\Pi_W^W \cup \{r\}^W$. For the sake of contradiction, suppose there is $W' \subseteq W$ that satisfies $\Pi_2^W \cup \{r\}^W$. Then, W' satisfies Π_2^W, which is a contradiction to the fact that W is a possible world of Π_2.

□

Lemma 2. Let Π be a program, where U is the set of activity records, and Π_2 be a program obtained from Π by replacing a rule $b = v \leftarrow l, B$ s.t:

- l is formed by $do(l)$ or $obs(l)$, and
- U satisfies l

with $b = v \leftarrow B$. We have $\Omega_{\Pi_2} = \Omega_{\Pi}$. □

Proof. Let L be the set of e-literals constructed from all attribute terms formed by *do* and *obs*. Since all rules with *obs* and *do* in heads have empty bodies, L is a splitting set of Π. We prove:

$$\text{every possible world of } \Pi \text{ satisfies } l \tag{A.7}$$

Let W be a possible world of Π. Since $bot_L(\Pi)$ consists of U and constraints, we have that the only possible for $bot_L(\Pi)$ is U. By Lemma 5, $W \setminus U \cap L = \emptyset$. That is, W contains no atoms formed by *do* and *obs* other than those from U. Therefore, W satisfies l, and (A.7) holds. Similarly,

$$\text{every possible world of } \Pi_2 \text{ satisfies } l \tag{A.8}$$

Now, there are two possibilities:

1. l is of the form *not* l'. By 5, W satisfies *not* l'. Then we have

$$\Pi^W = \Pi_2^W \tag{A.9}$$

and, therefore, W is a possible world of Π_2. On the other hand, from (A.9) we also have that every possible world of Π_2 is also a possible world of Π.

2. l is a literal. Let W be a possible world of Π. We have $\Pi_2^W = ((\Pi \setminus \{b = v \leftarrow l, B\}) \cup \{b = v \leftarrow B\})^W \subseteq \Pi^W \cup \{b = v \leftarrow B\})^W$. There are two cases:

   (a) W does not satisfy B. In this case,$\{b = v \leftarrow B\}^W$, and, therefore, Π_2^W is satisfied by W. For the sake of contradiction suppose there is $W' \subsetneq W$ such that W' satisfies Π_2^W. We have $\Pi^W \subseteq \Pi_2^W \cup \{b = v \leftarrow l, B\}^W$. Since W' satisfies Π_2^W, W' contains U. Since W' is a subset of W, it does not contain atoms formed by *do* and *obs* other than those from U.

   Therefore,

   $$W' \text{ satisfies } l \tag{A.10}$$

   Since W' satisfies Π_2^W, and $\{b = v \leftarrow B\}^W \subseteq \Pi_2^W$,

   $$W' \text{ satisfies } \{b = v \leftarrow B\})^W \tag{A.11}$$

   From (A.10) and (A.11) we have:

   $$W' \text{ satisfies } \{b = v \leftarrow l, B\}^W \tag{A.12}$$

   Therefore, W' satisfies $\Pi^W \subseteq \Pi_2^W \cup \{b = v \leftarrow l, B\}^W$, which is a contradiction to the fact that W is a possible world of Π.

   (b) W satisfies B. By (A.7) we have W satisfies l. Therefore, since the rules $b = v \leftarrow l, B$ belongs to Π, by Proposition 1, we have $b = v \in W$. Therefore, W satisfies $\Pi^W \cup \{b = v \leftarrow B\}^W$, and Π_2^W. For the sake of contradiction, suppose there is $W' \subsetneq W$ such that W' satisfies Π_2^W. By the reasoning identical to the one from (a) we obtain:

   $$W' \text{ satisfies l} \tag{A.13}$$

Since W' satisfies Π_2^W, it satisfies $\{b = v \leftarrow B\}^W$. Therefore, from (A.13), W' satisfies $\{b = v \leftarrow l, B\}^W$. Since $\Pi^W \subseteq \Pi_2^W \cup \{b = v \leftarrow l, B\}^W$, we have that W' satisfies Π^W, which is a contradiction to the fact that W is a possible world of Π.

Let W be a possible world of Π_2^W.

We have $\Pi^W \subseteq \Pi_2^W \cup \{b = v \leftarrow l, B\}^W$. There are two cases:

(a) W does not satisfy B. In this case,$\{b = v \leftarrow l, B\}^W$, and, therefore, Π^W is satisfied by W. For the sake of contradiction suppose there is $W' \subsetneq W$ such that W' satisfies Π^W.

We have $\Pi_2^W \subseteq \Pi^W \cup \{b = v \leftarrow B\}^W$. Since W' satisfies Π^W, W' contains U. Since W' is a subset of W, it does not contain atoms formed by *do* and *obs* other than those from U.

Therefore,

$$W' \text{ satisfies } l \tag{A.14}$$

Since W' satisfies Π^W, and $\{b = v \leftarrow l, B\}^W \subseteq \Pi^W$,

$$W' \text{ satisfies } \{b = v \leftarrow l, B\}^W \tag{A.15}$$

From (A.14) and (A.15) we have:

$$W' \text{ satisfies } \{b = v \leftarrow B\}^W \tag{A.16}$$

From (A.16) and (A.14) we have W satisfies $\{b = v \leftarrow l, B\}^W$. Therefore, W' satisfies $\Pi^W \subseteq \Pi_2^W \cup \{b = v \leftarrow l, B\}^W$, which is a contradiction to the fact that W is a possible world of Π.

(b) W satisfies B. By (A.7) we have W satisfies l. Therefore, since the rules $b = v \leftarrow l, B$ belongs to Π, by Proposition 1, we have $b = v \in W$.

Therefore, W satisfies $\Pi^W \cup \{b = v \leftarrow B\}^W$, and Π_2^W. For the sake of contradiction, suppose there is $W' \subsetneq W$ such that W' satisfies Π_2^W. By the reasoning identical to the one from (a) we obtain:

$$W' \text{ satisfies } l \tag{A.17}$$

Since W' satisfies Π_2^W, it satisfies $\{b = v \leftarrow B\}^W$. Therefore, from (A.17), W' satisfies $\{b = v \leftarrow l, B\}^W$. Since $\Pi^W \subseteq \Pi_2^W \cup \{b = v \leftarrow l, B\}^W$, we have that W' satisfies Π_2^W, which is a contradiction to the fact that W is a possible world of Π_2.

Let W be a possible world of Π. □

Proposition 2. Let Π be a P-log program and U be the set of activity records of Π. There exists a bijection $\psi : \Omega_\Pi \to \Omega_{\Pi_U}$ such that for every possible world W of Π

1. $W = \psi(W)$, and
2. $\mu_\Pi(W) = \mu_{\Pi_U}(W)$

□

Proof. It is sufficient to show that $\Omega_\Pi = \Omega_{\Pi_U}$, which follows immediately from Lemmas 1 and 2. □

A.2 Coherency Theorem Proof

We prove Theorem 1 in 3 steps. In section A.2.1 we describe a translation τ from P-log programs into ASP programs and show the relationship between the possible worlds of a given P-log program Π and answer sets of its translation $\tau(\Pi)$. Then, in section A.2.2 we formulate splitting set theorem for P-log originally defined in [Lifschitz & Turner, 1994] for Answer Set Prolog programs. Finally, in section A.2.3 we prove theorem 1 using the results from sections A.2.1 and A.2.2. The proof refines many of the results used in [Baral et al., 2009] to prove the coherency of causally ordered unitary programs in the original P-log language.

A.2.1 Translation from P-log to ASP

For every P-log program Π, not necessarily containing general axioms, with signature Σ we define an ASP program $\tau(\Pi)$ whose answer sets correspond to possible worlds of Π. More precisely, τ is defined on elements of Π as follows:

1. if $f(\bar{x}) = y$ is a literal of Σ, $\tau(f(\bar{x}) = y)$ is $f(\bar{x}, y)$;
2. if $f(\bar{x}) \neq y$ is a literal of Σ, $\tau(f(\bar{x}) \neq y)$ is $\neg f(\bar{x}, y)$;
3. if r is a rule of Π, $\tau(r)$ is an ASP rule obtained from r by replacing all occurrences of literals in the rule with their translations;
4. if Π is a P-log program with signature Σ, $\tau(\Pi)$ is an ASP program consisting of
   (a) the rules in the set $\{\tau(r) \mid r \text{ is a rule of } \Pi\}$; and
   (b) the rules of the form
   $$\neg f(\bar{x}, y_1) \leftarrow f(\bar{x}, y_2) \tag{A.18}$$
   for each two atoms $f(\bar{x}) = y_1$ and $f(\bar{x}) = y_2$ of Σ such that $y_1 \neq y_2$;
5. if A is a set of atoms of Σ, then $\tau(A)$ is the set of ASP literals
   $$\{f(\bar{x}, y) \mid f(\bar{x}) = y \in A\} \cup \{\neg f(\bar{x}, y) \mid f(\bar{x}) = y_1 \in A \wedge y_1 \neq y \wedge y \in range(f)\}$$
6. If L is a set of literals of Σ, then $\tau(L)$ is the set of ASP literals:
   $$\tau(\{f(\bar{x}) = y \mid f(\bar{x}) = y \in L\}) \cup \{\tau(f(\bar{x}) \neq y) \mid f(\bar{x}) \neq y \in L\}$$

Lemma 3. If I is an interpretation of Σ, then I satisfies a literal l of Σ if and only if $\tau(I)$ satisfies $\tau(l)$

Proof.

$\Rightarrow$

1. if l is of the form $f(\bar{x}) = y$ and I satisfies $f(\bar{x}) = y$, $\tau(I)$ contains an atom $\tau(l) = f(\bar{x}, y)$.

2. If l is of the form $f(\bar{x}) \neq y$, and I satisfies l, by definition of satisfiability there must exists an atom $f(\bar{x}) = y_1$, where $y_1 \neq y$, such that I satisfies $f(\bar{x}) = y_1$. Therefore, from part 5 of the definition of τ, $\tau(I)$ contains $\neg f(\bar{x}, y)$.

$\Leftarrow$

1. if l is of the form $f(\bar{x}, y)$ and $\tau(I)$ satisfies $f(\bar{x}, y)$, then, by construction of $\tau(I)$, we have $f(\bar{x}) = y \in I$.

2. If l is of the form $\neg f(\bar{x}, y)$, and $\tau(I)$ satisfies l, by construction of $\tau(I)$, I must contain an atom $f(\bar{x}) = y_1$ for $y_1 \neq y$. Therefore, by definition of satisfiability, I satisfies $f(\bar{x}) \neq y$.

□

Lemma 4. Let Π be a P-log program not necessarily containing all general axioms. An interpretation W of Π is a possible world of Π if and only if $\tau(W)$ is an answer set of $\tau(\Pi)$

Proof.

$\Rightarrow$ Let W be a possible world of Π. We prove that $\tau(W)$ is an answer set of $\tau(\Pi)$.

1) We show that $\tau(W)$ is a consistent set of ASP literals. We prove by contradiction. Suppose $\tau(W)$ is inconsistent. Thus, there exists an ASP atom $f(\bar{x}, y)$ such that $\tau(W)$ contains both $f(\bar{x}, y)$ and $\neg f(\bar{x}, y)$. By definition of $\tau(W)$, it implies that $I(f(\bar{x})) = y$ and there exists $y_1 \neq y$ such that $I(f(\bar{x})) = y_1$. Thus, since I is a mapping by definition, we have a contradiction.

2) We show that $\tau(W)$ satisfies the rules of the reduct $\tau(\Pi)^{\tau(W)}$. By R_a we denote the rules of $\tau(\Pi)$ described in 4.a) of the definition of τ; by R_b we denote rules described in 4.b) of the same definition. Clearly, $\tau(\Pi)^{\tau(W)} = R_a^{\tau(W)} \cup R_b^{\tau(W)}$.

   (a) We show that $\tau(W)$ satisfies the rules of $R_a^{\tau(W)}$. Let r be a rule in $R_a^{\tau(W)}$ such that $\tau(W)$ satisfies the body of r. We prove that $\tau(W)$ satisfies the head of r. From lemma 3, the definition of τ and the definition of R_a, we conclude that there is a rule r' in Π^W such that $r = \tau(r')$. By lemma 3 and the definition of $\tau(W)$, W satisfies the body of r'. Since W is a possible world of Π, W satisfies the head of r'. By lemma 3 and the definition of $\tau(r')$, $\tau(W)$ satisfies the head of r.

   (b) We show that $\tau(W)$ satisfies the rules of $R_b^{\tau(W)}$. Let r be a rule in $R_b^{\tau(W)}$ given below

$$\neg f(\bar{x}, y_1) \leftarrow f(\bar{x}, y_2)$$

   such that $\tau(W)$ satisfies $f(\bar{x}, y_2)$. We need to show that $\tau(W)$ satisfies $\neg f(\bar{x}, y_1)$.

   By lemma 3, since $\tau(W)$ satisfies $f(\bar{x}, y_2)$, W satisfies $f(\bar{x}) = y_2$, that, by definition of $\tau(W)$, implies that $\tau(W)$ satisfies $\neg f(\bar{x}, y)$.

3) We show that $\tau(W)$ is minimal, that is, there does not exist a set of literals A such that A satisfies the rules of $\tau(\Pi)^{\tau(W)}$ and A is a proper subset of $\tau(W)$. We prove by contradiction. Suppose there is A such that

$$A \text{ satisfies all the rules in } \tau(\Pi)^{\tau(W)} \tag{A.19}$$

and

$$A \text{ is a proper subset of } \tau(W) \tag{A.20}$$

Since $A \subsetneq \tau(W)$, by construction of $\tau(W)$ and the fact that W is an interpretation, A does not contain a pair of atoms $f(\bar{x}, y_1)$, $f(\bar{x}, y_2)$ for $y_1 \neq y_2$. Thus, we can construct interpretation I of Σ such that I maps $f(\bar{x})$ to y if and only if $f(\bar{x}, y)$ belongs to A.

In a) we show that I satisfies the rules of Π^W. In b) we show that $I \subsetneq W$, thus, obtaining a contradiction (by the definition of possible world, W should be a minimal interpretation satisfying Π^W).

(a) We prove that I satisfies the rules of Π^W. Let r be a rule of Π^W such that I satisfies the body of r. We need to show that I satisfies the head of r. First we prove that A satisfies the body of $\tau(r)$. Since r belongs to Π^W, r does not contain literals preceded by default negation.

- Let l be a literal of the form $f(\bar{x}, y)$ belonging to the body of $\tau(r)$. Since I satisfies the body of r, $I(f(\bar{x})) = y$. By construction of I, A satisfies $f(\bar{x}, y)$.
- Let l be a literal of the form $\neg f(\bar{x}, y)$ belonging to the body of $\tau(r)$. Since I satisfies the body of r, $I(f(\bar{x})) = y_1$, where $y_1 \neq y$. By construction of I, $f(\bar{x}, y_1)$ belongs to A. Since A satisfies the rules of $\tau(\Pi)^{\tau(W)}$, including the rule

$$\neg f(\bar{x}, y) \leftarrow f(\bar{x}, y_1)$$

Therefore, A satisfies $\neg f(\bar{x}, y)$.

By definition of reduct and from lemma 3 it follows that $\tau(r)$ belongs to $\tau(\Pi)^{\tau(W)}$. Therefore, since A satisfies the body of $\tau(r)$, and A satisfies the rules of $\tau(\Pi)^{\tau(W)}$, it follows that A satisfies the head of $\tau(r)$. Therefore, there exists an ASP literal $f(\bar{x}, y)$ in the head of $\tau(r)$ satisfied by A. By construction of I, I satisfies literal $f(\bar{x}) = y$. By definition of $\tau(r)$, the head of r contains the literal $f(\bar{x}) = y$. Therefore, I satisfies the head of r.

(b) We prove that $I \subsetneq W$. By construction of I, I contains a literal $f(\bar{x}) = y$

if an only if $f(\bar{x}, y) \in A$. By definition of $\tau(W)$, W contains a literal $f(\bar{x}) = y$ if and only if $f(\bar{x}, y) \in \tau(W)$. For a set of ASP literals S, by S^+ we denote the subset of S containing all positive literals of S and by S^- we denote the subset of S containing all negative literals in S (that is, literals of the form $\neg f(\bar{y})$). It is sufficient to show that $A^+ \subsetneq \tau(W)^+$. We prove by contradiction

i. Suppose A^+ is not a proper subset of $\tau(W)^+$

ii. Since A is a proper subset of $\tau(W)$, A^+ is a subset of A and $\tau(W)^+$ is a subset of $\tau(W)$, from i. we have

$$|\tau(W)^+| = |A^+| \tag{A.21}$$

(and, even more precisely, $\tau(W)^+ = A^+$)

iii. By definition of $\tau(W)$,

$$|\tau(W)^-| = \sum_{f(\bar{x}) \in \{f(\bar{x}) | \exists y : W(f(\bar{x})) = y\}} (|range(f(\bar{x}))| - 1) \tag{A.22}$$

iv. For each positive ASP literal $f(\bar{x}, y_2)$ in A and for each y_1 in $range(f)$ such that $y_1 \neq y_2$, there is rule (A.18) in $\tau(\Pi)$. Since A satisfies the rules of $\tau(W)$, in particular, those of the form (A.18), from (A.21) and the construction of $\tau(W)$, it follows that the number of negative literals in A is bounded below as follows:

$$|A^-| \geq \sum_{f(\bar{x}) \in \{f(\bar{x}) | \exists y : W(f(\bar{x})) = y\}} (|range(f(\bar{x}))| - 1) \tag{A.23}$$

v. By combining equation (A.22) and inequality (A.23), we get

$$|A^-| \geq |\tau(W)^-| \tag{A.24}$$

vi. From equation (A.21) and inequality (A.24) we have

$$\begin{aligned}
|A| &= |A^+| + |A^-| \\
&= |\tau(W)^+| + |A^-| \\
&\geq |\tau(W)^+| + |\tau(W)^-| \\
&= |\tau(W)| \qquad \text{(A.25)}
\end{aligned}$$

that contradicts our original assumption (A.20) stating that A is a proper subset $\tau(W)$.

4) From 1)- 3) it follows that $\tau(W)$ is an answer set of $\tau(\Pi)$.

$\Leftarrow$ Let A be an answer set of $\tau(\Pi)$ and I be an interpretation of Π such that $A = \tau(I)$. We prove that I is a possible world of Π. In 5) we show that I satisfies the rules of Π^I and in 6) we show the minimality of I.

5) We prove that I satisfies the rules of Π^I. Let r be a rule of Π^I such that I satisfies the body of r. From lemma 3 and the definitions of reduct in P-log and ASP it follows that the rule $\tau(r)$ belongs to $\tau(\Pi)^A$, and moreover, A satisfies the body of $\tau(r)$. Since A is an answer set of $\tau(\Pi)$, A satisfies the head of r. By definitions of $\tau(r)$ and $\tau(I)$ and lemma 3, this means that I satisfies the head of r and, therefore r itself.

6) We prove that I is minimal, that is, there does not exist an interpretation I' such that I' satisfies the rules of Π^I and $I' \subsetneq I$. We prove by contradiction. Suppose such I' exists. In a) we show that $\tau(I')$ satisfies the rules of $\tau(\Pi)^A$ and in b) we show that $\tau(I')$ is a proper subset of A, thus, obtaining a contradiction to the fact that A is an answer set of $\tau(\Pi)$.

(a) We prove that $\tau(I')$ satisfies the rules of $\tau(\Pi)^A$. Let r be a rule of $\tau(\Pi)^A$ such that $\tau(I')$ satisfies the body of r. We show that $\tau(I')$ satisfies the head of r. From lemma 3 and the definition of τ and the definition of reducts in ASP and P-log it follows that Π^I contains a rule r' such that $r = \tau(r')$. From 3 we have that I' satisfies the body of r'. Since I' satisfies the rules of Π^I, I' satisfies the head of r'. Therefore, from lemma 3 it follows that $\tau(I')$ satisfies the head of $r = \tau(r')$, and, therefore, the rule r itself.

(b) We prove that $\tau(I')$ is a proper subset of $A = \tau(I)$. By definition of I',

$$I' \subsetneq I \tag{A.26}$$

Thus, by definition of τ,

$$\tau(I')^+ \text{ is a proper subset of } \tau(I)^+ \tag{A.27}$$

From (A.27) it follows immediately

$$|\tau(I')^+| < |\tau(I)^+| \tag{A.28}$$

By definition of $\tau(I)$ and $\tau(I')$, we have

$$\tau(I)^- = \bigcup_{f(\bar{x})=y \in I} \{\neg f(\bar{x}) = y_1) | y_1 \in range(f(\bar{x})) \wedge y_1 \neq y\} \tag{A.29}$$

$$\tau(I')^- = \bigcup_{f(\bar{x})=y \in I'} \{\neg f(\bar{x}) = y_1) | y_1 \in range(f(\bar{x})) \wedge y_1 \neq y\} \tag{A.30}$$

From (A.26), (A.29) and (A.30) it follows that

$$\tau(I')^- \subseteq \tau(I)^- \tag{A.31}$$

From (A.27) and (A.31) and the fact that $\tau(I) = \tau(I)^+ \cup \tau(I)^-$ and $\tau(I') =$

$\tau(I')^+ \cup \tau(I')^-$ we get

$$\tau(I') \text{ is a proper subset of } \tau(I) \tag{A.32}$$

□

Proposition 13. Let Π be a P-log program and W_1, W_2 be two possible worlds of Π. It is not true that $W_1 \subsetneq W_2$.

Proof. We prove by contradiction. Let W_1 and W_2 be two possible world of a program Π such that

$$W_1 \subsetneq W_2 \tag{A.33}$$

By Lemma 4,

$$\tau(W_1) \text{ and } \tau(W_2) \text{ are answer sets of } \tau(\Pi). \tag{A.34}$$

By definition of τ,

$$\tau(W_1) = W_1 \cup \bigcup_{f(\bar{x})=y \in W_1} \{f(\bar{x}) \neq y_1 | y_1 \in range(f(\bar{x})) \wedge y_1 \neq y\} \tag{A.35}$$

$$\tau(W_2) = W_2 \cup \bigcup_{f(\bar{x})=y \in W_2} \{f(\bar{x}) \neq y_1 | y_1 \in range(f(\bar{x})) \wedge y_1 \neq y\} \tag{A.36}$$

From (A.33), (A.35) and (A.36) it follows that

$$\tau(W_1) \subsetneq \tau(W_2) \tag{A.37}$$

(A.37) and (A.34) contradict the theorem about minimality of answer sets of ASP programs (see Lemma 1 in [Gelfond & Lifschitz, 1991b]).

□

A.2.2 Splitting Set Theorem For P-log

In this section we present the P-log version of the original Splitting Set Theorem from [Lifschitz & Turner, 1994]. The adoption requires change in the definition of splitting set (see Definition 61). Other definitions follow [Lifschitz & Turner, 1994] and are presented for completeness.

Let Π be a program with signature Σ, and X and U be sets of literals of Σ. As in [Lifschitz & Turner, 1994], we will define the *bottom* and the *top* of a program Π with respect to X and U, denoted by $b_U(\Pi)$ and $e_U(\Pi \setminus b_U(P), X)$ correspondingly.

For a rule r of the form

$$l \leftarrow l_1, \ldots, l_k, not\ l_{k+1}, \ldots, not\ l_m \tag{A.38}$$

where $l_1, \ldots, l_m$ are literals, we will introduce notations:

$$pos(r) = \{l_1, \ldots, l_k\}$$

$$neg(r) = \{l_{l+1}, \ldots, l_m\}$$

$$head(r) = l$$

$$lit(r) = head(r) \cup pos(r) \cup neg(r)$$

Definition 58 (Bottom w.r.t. U)**.**
The bottom of Π w.r.t U, denoted by $b_U(\Pi)$, is a program such that:

1. $b_U(\Pi) = \{r \mid r \in \Pi \text{ and } lit(r) \subseteq U\}$
2. the signature of $b_U(\Pi)$ consists of all attribute terms of Σ which form literals from U

□

We next define Top. For rule r such that $pos(r) \cap U$ is satisfied by X and every literal from $(neg(r) \cap U)$ is not satisfied by X, we define $R_U(r)$ to be the rule such that:

$$head(R_U(r)) = head(r), pos(R_U(r)) = pos(r) \setminus U, neg(R_U(r)) = neg(r) \setminus U$$

Definition 59 (Top w.r.t. X and U)**.**
The top of Π w.r.t X and U, denoted by $e_U(\Pi, X)$, is a program such that:

1. the rules of $e_U(\Pi, X)$ are

$$\{R_U(r) \mid r \in \Pi \text{ and } pos(r) \cap U \text{ is satisfied by } X \text{ and every e-literal from } \{not\ l \mid l \in neg(r) \cap U\} \text{ is not satisfied by } X\}$$

2. the signature of $e_U(\Pi, X)$ consists of all attribute terms of Σ which do not form a literal in U.

□

We borrow the definition of a solution to Π w.r.t U:

Definition 60 (Solution to Π w.r.t U)**.**
Let Π be a P-log program. A solution to Π w.r.t U is a pair $\langle X, Y\rangle$ of sets of literals such that:

1. X is possible world of $b_U(\Pi)$
2. Y is a possible world of $e_U(\Pi \setminus b_U(\Pi), X)$
3. $X \cup Y$ is consistent

□

We will next define a splitting set for P-log programs:

Definition 61 (Splitting set)**.**
A *splitting set* for a P-log program Π is any set U of literals of Π's signature such that,

1. for every rule r of Π if $head(r) \in U$, then $pos(r) \cup neg(r) \subseteq U$,
2. if a literal formed by attribute term $f(\bar{x})$ belongs to U, then all the literals of Σ formed by $f(\bar{x})$ belong to U.

□

Finally, we state the splitting set theorem for P-log:

Theorem 2. *[Splitting Set Theorem]*
Let U be a splitting set for a program Π. A set A of literals is a possible world of Π if and only if $A = X \cup Y$ for some solution $\langle X, Y \rangle$ to Π with respect to U.

□

Note that the condition 2 from Definition 61, absent from the original definition of splitting set, is necessary for the correctness of the theorem. For instance, consider the program:

```
#s: {1,2}.
p: #boolean.
f:   #s.
p:- f != 1.
f = 2.
```

If condition 2 is not used, $U = \{p, f \neq 1\}$ would be a splitting set of Π. However, the theorem then wouldn't hold for U. The program has only one possible world $\{f = 2\}$. However, there does not exists a solution $\langle X, Y \rangle$ with respect to U, such that $A = X \cup U$, because, the program $b_U(\Pi)$, which contains only one rule

```
p:- not f != 1.
```

has exactly one possible world $\{p\}$, and, therefore, $X \cup Y$ must contain p for any solution $\langle X, Y \rangle$ of Π with respect to U.

Proof for theorem 2.
In 1 we show that $\tau(U)$ is a splitting set (as defined in [Lifschitz & Turner, 1994]) for the ASP program $\tau(\Pi)$. In 2 we show $\tau(b_U(\Pi)) = b_{\tau(U)}(\tau(\Pi))$. In 3 we show $e_{\tau(U)}(\tau(\Pi) \setminus b_{\tau(U)}(\tau(\Pi)), X') = \tau(e_U(\Pi \setminus b_U(\Pi), X))$. In 4 we use the results from 1-3 to prove that if A a possible world of Π, then $A = X \cup Y$ for some solution $\langle X, Y \rangle$ to Π with respect to U. In 5 we use the results from 1-3 to prove that if $A = X \cup Y$ for some solution $\langle X, Y \rangle$ to Π with respect to U, then A is a possible world of Π.

1. We show that $\tau(U)$ is a splitting set (as defined in [Lifschitz & Turner, 1994]) for the program $\tau(\Pi)$. Let r be a rule of $\tau(\Pi)$ such that:

$$\text{the head of } r \text{ is included into } \tau(U) \tag{A.39}$$

   We need to show that all the literals occurring in the body of r are included into $\tau(U)$. We consider two possible cases:

   (a) there exists r' of Π such that $r = \tau(r')$. In this case, by construction of $\tau(r')$ and clause 1 of Definition 61 we have that every literal occurring in $pos(r) \cup neg(r)$ is included into U. Thus, by definition of $\tau(U)$ and from $r = \tau(r')$, every literal from the body of r is included into $\tau(U)$.

   (b) r is of the form $\neg f(x, y_1) \leftarrow f(x, y)$ where $y_1 \neq y$. In this case, by construction of $\tau(U)$ from (A.39) we have that $f(x) \neq y_1 \in U$. Therefore, by clause 2) of definition 61 we have that $f(x) = y \in U$. By definition of $\tau(f(x) = y)$ and $\tau(U)$ we have that $f(x, y) \in \tau(U)$. Therefore, every literal in the body of r belongs to $\tau(U)$.

2. We prove:

$$\tau(b_U(\Pi)) = b_{\tau(U)}(\tau(\Pi)) \tag{A.40}$$

We prove (A.40) in two directions:

$$\tau(b_U(\Pi)) \subseteq b_{\tau(U)}(\tau(\Pi)) \tag{A.41}$$

$$b_{\tau(U)}(\tau(\Pi)) \subseteq \tau(b_U(\Pi)) \tag{A.42}$$

We start from (A.41). Suppose r is a a rule such that

$$r \in \tau(b_U(\Pi)) \tag{A.43}$$

There are two possible cases:

- r is a translation of a rule of $b_U(\Pi)$. In this case, since $b_U(\Pi) \subseteq \Pi$, r is a translation of a rule in Π. Therefore,

$$r \text{ belongs to } \tau(\Pi) \tag{A.44}$$

  Also, since every literal occurring in every rule in $b_U(\Pi)$ is from U, and r is a translation of a rule in $b_U(\Pi)$, we have that:

$$\text{every literal occurring in } r \text{ is from } \tau(U) \tag{A.45}$$

  From (A.45) and (A.44) we have

$$r \in b_{\tau(U)}(\tau(\Pi)) \tag{A.46}$$

Therefore, (A.41) holds.

- r is of the form $\neg f(\bar{x}, y_1) \leftarrow f(\bar{x}, y_2)$ In this case, by definition of τ from (A.43) we have $f(\bar{x}) = y_2$ and $f(\bar{x}) = y_1$ are atoms of $b_U(\Pi)$, which means U contains literals l_1 and l_2 formed by $f(\bar{x}) = y_1$ and $f(\bar{x}) = y_2$ respectively. Since U is a splitting set, by condition 2 we have that U contains atoms $f(\bar{x}) = y_1$ and $f(\bar{x}) = y_2$. Therefore, $\tau(U)$ contains literals $\neg f(\bar{x}, y_1)$ and $f(\bar{x}, y_2)$, $r \in b_{\tau(U)}(\tau(\Pi))$, and (A.41) holds.

We next show (A.42). Suppose r is a a rule such that

$$r \in b_{\tau(U)}(\tau(\Pi)) \tag{A.47}$$

There are two possible cases:

- r is a translation of some rule r' of Π. In this case, from (A.47) we have that all literals from r are from $\tau(U)$. Therefore, $lit(r') \subseteq U$, $r' \in b_U(\Pi)$ and $r \in \tau(b_U(\Pi))$. Therefore, (A.42) holds.
- r is of the form $\neg f(\bar{x}, y_1) \leftarrow f(\bar{x}, y_2)$. By construction of $b_{\tau(U)}(\tau(\Pi))$ we have that:

$$\neg f(\bar{x}, y_1) \in \tau(U) \tag{A.48}$$

$$f(\bar{x}, y_2) \in \tau(U) \tag{A.49}$$

and

$$r \in \tau(\Pi) \tag{A.50}$$

From (A.48) and (A.49) we have:

$$\text{all the literals of } \Sigma \text{ formed by } f(\bar{x}) \text{ belong to } U \tag{A.51}$$

From (A.51) we have that

$$f(\bar{x}) \text{ belongs to the signature of } b_U(\Pi) \tag{A.52}$$

Therefore, by definition of $\tau(\Pi)$, we have $r \in \tau(b_U(\Pi)$. Therefore, (A.42) holds.

From (A.41) and (A.42) we have (A.40).

3. We show that for every possible world X of $b_U(\Pi)$:

$$e_{\tau(U)}(\tau(\Pi) \setminus b_{\tau(U)}(\tau(\Pi)), \tau(X)) = \tau(e_U(\Pi \setminus b_U(\Pi), X)) \tag{A.53}$$

We prove (A.53) in two directions:

$$e_{\tau(U)}(\tau(\Pi) \setminus b_{\tau(U)}(\tau(\Pi)), \tau(X)) \subseteq \tau(e_U(\Pi \setminus b_U(\Pi), X)) \tag{A.54}$$

$$\tau(e_U(\Pi \setminus b_U(\Pi), X)) \subseteq e_{\tau(U)}(\tau(\Pi) \setminus b_{\tau(U)}(\tau(\Pi)), \tau(X)) \tag{A.55}$$

We start from (A.54). Let r be a rule s.t.

$$r \in e_{\tau(U)}(\tau(\Pi) \setminus b_{\tau(U)}(\tau(\Pi)), \tau(X)) \tag{A.56}$$

By construction of $e_{\tau(U)}(\tau(\Pi) \setminus b_{\tau(U)}(\tau(\Pi)), \tau(X))$ we have that $r = R_{\tau(U)}(r')$, where

$$r' \in \tau(\Pi) \tag{A.57}$$

$$r' \notin b_{\tau(U)}(\tau(\Pi)) \tag{A.58}$$

$$pos(r') \cap \tau(U) \subseteq \tau(X) \tag{A.59}$$

$$neg(r') \cap \tau(U) \cap \tau(X) = \emptyset \tag{A.60}$$

From (A.58) we have:

$$r' \text{ contains a literal which is not from } \tau(U) \tag{A.61}$$

There are only two possibilities:

- $r' = \tau(r'')$ for some rule r'' from Π. From (A.59) and (c) by lemma 3 we have:

$$pos(r'') \cap U \text{ is satisfied by } X \tag{A.62}$$

From (A.60) and (c) by lemma 3 we have:

$$\text{every literal in } \{not\ l \mid l \in neg(r'') \cap U\} \text{ is not satisfied by } X \tag{A.63}$$

From (A.61) we have:

$$lit(r'') \text{ contains a literal which is not in } U \tag{A.64}$$

Therefore,

$$r'' \notin b_U(\Pi) \tag{A.65}$$

From (A.62),(A.63) and (A.65) we have:

$$R_U(r'') \in e_U(\Pi \setminus b_U(\Pi), X) \tag{A.66}$$

Therefore,

$$\tau(R_U(r'')) \in \tau(e_U(\Pi \setminus b_U(\Pi), X)) \tag{A.67}$$

Since $r' = \tau(r'')$, by construction of R and by Lemma 3 we have:

$$\tau(R_U(r'')) = R_{\tau(U)}(r') \tag{A.68}$$

From (A.67) and (A.68) we have:

$$R_{\tau(U)}(r') \in \tau(e_U(\Pi \setminus b_U(\Pi), X)) \tag{A.69}$$

Therefore, since $r = R_{\tau(U)}(r')$, we have:

$$r \in \tau(e_U(\Pi \setminus b_U(\Pi), X)) \tag{A.70}$$

Therefore, in this case, (A.54) holds.

- r' is of the form

  $$\neg f(\bar{x}, y_1) \leftarrow f(\bar{x}, y_2)$$

  where $y_1 \neq y_2$ and

  $$f(\bar{x}) \text{ is an attribute term of } \Sigma \tag{A.71}$$

  From (A.61) by clause 2 of the definition of the splitting set we have:

  $$\text{no literal formed by } f(\bar{x}) \text{ belongs to U} \tag{A.72}$$

From (A.71) and (A.72) by Definition 59 we have:

$$f(\bar{x}) \text{ belongs to the signature of } e_U(\Pi \setminus b_U(\Pi), X) \tag{A.73}$$

From (A.72) we have:

$$R_{\tau(U)}(r') = r' = r \tag{A.74}$$

From (A.73) we have:

$$r' \in \tau(e_U(\Pi \setminus b_U(\Pi), X)) \tag{A.75}$$

From (A.75) and (A.74) we have:

$$r \in \tau(e_U(\Pi \setminus b_U(\Pi), X)) \tag{A.76}$$

Therefore, in this case, (A.54) holds.

We next prove (A.55).

Let r be a rule s.t.

$$r \in \tau(e_U(\Pi \setminus b_U(\Pi), X)) \tag{A.77}$$

There are only two possibilities:

- $r = \tau(r')$ for some $r' \in e_U(\Pi \setminus b_U(\Pi), X)$. By construction of $e_U(\Pi \setminus b_U(\Pi), X)$, we have that there is r'' such that:

$$r'' \in \Pi \tag{A.78}$$

$$r' = R_U(r'') \tag{A.79}$$

and:

$$r'' \notin b_U(\Pi) \tag{A.80}$$

$$pos(r'') \cap U \text{ is satisfied by } X \tag{A.81}$$

$$\text{every literal from } neg(r'') \cap U \text{ is not satisfied by } X \tag{A.82}$$

From (A.80) we have:

$$lit(r'') \text{ contains a literal not from } U \tag{A.83}$$

From (A.83), clause 2 of the splitting set definition and the construction of τ we have:

$$lit(\tau(r'')) \text{ contains a literal not from } \tau(U) \tag{A.84}$$

Therefore,

$$\tau(r'') \notin b_{\tau(U)}(\tau(\Pi)) \tag{A.85}$$

From (A.81) and Lemma 3 we have:

$$pos(\tau(r'')) \cap \tau(U) \text{ is satisfied by } \tau(X) \tag{A.86}$$

From (A.82) and Lemma 3 we have:

$$\text{every literal from } neg(\tau(r'')) \cap \tau(U) \text{ is not satisfied by } \tau(X) \tag{A.87}$$

From (A.78) we have:

$$\tau(r'') \in \tau(\Pi) \tag{A.88}$$

From (A.85), (A.86), (A.87) and (A.88) we have:

$$R_{\tau(U)}(\tau(r'')) \in e_{\tau(U)}(\tau(\Pi) \setminus b_{\tau(U)}(\tau(\Pi)), \tau(X)) \tag{A.89}$$

By construction of R and by Lemma 3 we have:

$$\tau(R_U(r'')) = R_{\tau(U)}(\tau(r'')) \tag{A.90}$$

From (A.89) and (A.90) we have:

$$\tau(R_U(r'')) \in e_{\tau(U)}(\tau(\Pi) \setminus b_{\tau(U)}(\tau(\Pi)), \tau(X)) \tag{A.91}$$

From (A.79) and (A.91) we have:

$$\tau(r') \in e_{\tau(U)}(\tau(\Pi) \setminus b_{\tau(U)}(\tau(\Pi)), \tau(X)) \tag{A.92}$$

From the fact that $r = \tau(r')$ and (A.92) we have:

$$r \in e_{\tau(U)}(\tau(\Pi) \setminus b_{\tau(U)}(\tau(\Pi)), \tau(X)) \tag{A.93}$$

Therefore, in this case, (A.55) holds.

- r is of the form

  $$\neg f(\bar{x}, y_1) \leftarrow f(\bar{x}, y_2)$$

  for some $y_1 \neq y_2$.

From (A.77) by construction of $\tau(e_U(\Pi \setminus b_U(\Pi), X))$ we have:

$$f(\bar{x}) = y_1 \text{ and } f(\bar{x}) = y_2 \text{ are atoms of the signature of } e_U(\Pi \setminus b_U(\Pi), X) \tag{A.94}$$

Therefore, by clause 2 of Definition 59, we have:

$$f(\bar{x}) = y_1 \text{ and } f(\bar{x}) = y_2 \text{ are atoms of } \Sigma \tag{A.95}$$

and

$$\text{no literals of } U \text{ are formed by } f(\bar{x}) \tag{A.96}$$

Therefore,

$$\tau(U) \text{ contains no literals of the forms } f(\bar{x}, y) \text{ or } \neg f(\bar{x}, y) \tag{A.97}$$

Since $neg(r) = \{\}$, we have:

$$\text{every literal in } \{not\ l \mid l \in neg(r) \cap \tau(U)\} \text{ is not satisfied by } \tau(X) \tag{A.98}$$

From (A.97) we have $pos(r) \cap \tau(U) = \emptyset$, therefore:

$$\text{every literal in } pos(r) \cap \tau(U) \text{ is satisfied by } \tau(X) \tag{A.99}$$

From (A.95) we have:

$$r \in \tau(\Pi) \tag{A.100}$$

From (A.97) we have:

$$r \notin b_{\tau(U)}(\tau(\Pi)) \tag{A.101}$$

From (A.100), (A.101), (A.99), (A.98) we have:

$$R_{\tau(U)}(r) \in e_{\tau(U)}(\tau(\Pi) \setminus b_{\tau(U)}(\tau(\Pi)), \tau(X)) \quad \text{(A.102)}$$

From (A.97) we have:

$$R_{\tau(U)}(r) = r \quad \text{(A.103)}$$

From (A.103) and (A.102) we have:

$$r \in e_{\tau(U)}(\tau(\Pi) \setminus b_{\tau(U)}(\tau(\Pi)), \tau(X)) \quad \text{(A.104)}$$

Therefore, in this case, (A.55) holds.

From (A.54) and (A.55) we have (A.53).

4. $\rightarrow$ we show that if A a possible world of Π, then $A = X \cup Y$ for some solution $\langle X, Y \rangle$ to Π with respect to U.

   Let A be a possible world of Π. By lemma 4 we have that $\tau(A)$ is an answer set of $\tau(\Pi)$. By splitting set theorem from [Lifschitz & Turner, 1994] we have that $\tau(A) = X' \cup Y'$ for some solution $\langle X', Y' \rangle$ to $\tau(\Pi)$ with respect to $\tau(U)$. Let Σ be the signature of Π. We will construct two sets of atoms of Σ, X and Y, such that

   - $X' = \tau(X)$
   - $Y' = \tau(Y)$
   - $\langle X, Y \rangle$ is a solution to Π with respect to U
   - $X \cup Y = A$.

   In (a) we construct X. In (b) we construct Y. In (c) we show $X' = \tau(X)$. In (d) we show $Y' = \tau(Y)$. In (e) we show $\langle X, Y \rangle$ is a solution to Π with respect to U. In (f) we show $X \cup Y = A$.

(a) We construct X. First, from construction of $\tau(\Pi)$ if follows that

$$\text{if } f(\bar{x}, y) \text{ occurs as an atom in } \tau(\Pi) \text{ then } f(\bar{x}) = y \text{ is an atom of } \Sigma \tag{A.105}$$

Since X' is an answer set of the program $b_{\tau(U)}(\tau(\Pi))$, it can only contain literals which occur in the head of the rules of the program $b_{\tau(U)}(\tau(\Pi))$, and, by definition of $b_{\tau(U)}(\tau(\Pi))$,

$$\text{all literals in } X' \text{ occur in } \tau(\Pi) \tag{A.106}$$

Let X be the set defined as follows:

$$X = \{f(\bar{x}) = y \mid f(\bar{x}, y) \in X'\} \tag{A.107}$$

From (A.105) and (A.106) we have that X is a set of atoms of Σ.

(b) We construct Y. Using arguments similar to the ones in (a), we can show that if $f(x, y) \in Y'$, then $f(x) = y$ is an atom for Σ. Then the set Y, defined as follows

$$Y = \{f(\bar{x}) = y \mid f(\bar{x}, y) \in Y'\}, \tag{A.108}$$

is a set of atoms of Σ.

(c) We show that $X' = \tau(X)$. Since X' is an answer set of the program $b_{\tau(U)}(\tau(\Pi))$, X' is consistent. Therefore, it is sufficient to show that for every atom $f(\bar{x}) = y$ such that

$$f(\bar{x}) = y \in X \tag{A.109}$$

in X,

$$\{\neg f(\bar{x}, y_1) \mid y_1 \neq y\} \subseteq X' \tag{A.110}$$

By construction of $\tau(\Pi)$, for every literal $f(\bar{x}) \neq y_1$ where $y_1 \neq y$, $\tau(\Pi)$ contains a rule r: $\neg f(\bar{x}, y_1) \leftarrow f(\bar{x}, y)$

We prove that

$$r \text{ belongs to } b_{\tau(U)}(\tau(\Pi)) \tag{A.111}$$

Since X' is an answer set of $b_{\tau(U)}\tau(\Pi)$, and X' contains $f(\bar{x}, y)$, $f(\bar{x}, y) \in \tau(U)$. By definition of $\tau(U)$, $\tau(U)$ should also include $\neg f(\bar{x}, y_1)$. Thus, $\tau(U)$ contains both $\tau(f(\bar{x}) = y)$ and $\tau(f(\bar{x}) \neq y_1)$ and, by construction of $b_{\tau(U)}(\tau(\Pi))$, we have (A.111)

Since X' is an answer set of $b_{\tau(U)}\tau(\Pi)$, from (A.111) we have:

$$X' \text{ satisfies } r \tag{A.112}$$

From (A.109) and (A.107) we have that X' satisfies the body of r. Therefore, from (A.112) we have X' also satisfies the head of r, which is $\neg f(\bar{x}, y_1)$. Therefore, (A.110) holds.

(d) We show that $Y' = \tau(Y)$. Since Y' is an answer set of $e_{\tau(U)}(\tau(\Pi) \setminus b_{\tau(U)}(\tau(\Pi)), X')$, Y' is consistent. Therefore, it is is sufficient to show that for every atom $f(\bar{x}) = y$ such that

$$f(\bar{x}) = y \in Y \tag{A.113}$$

we have

$$\{\neg f(\bar{x}, y_1) \mid y_1 \neq y\} \subseteq Y' \tag{A.114}$$

By construction of $\tau(\Pi)$, for every literal $f(\bar{x}) \neq y_1$ of Σ where $y_1 \neq y$, there is a rule r

$$\neg f(\bar{x}, y_1) \leftarrow f(\bar{x}, y)$$

such that

$$r \in \tau(\Pi) \tag{A.115}$$

We prove that

$$r \in e_{\tau(U)}(\tau(\Pi) \setminus b_{\tau(U)}(\tau(\Pi)), X') \tag{A.116}$$

From the results on page 5 of [Lifschitz & Turner, 1994], we have:

$$Y' \cap \tau(U) = \emptyset \tag{A.117}$$

From (A.117), (A.113) and (A.108) we have:

$$f(\bar{x}, y) \notin \tau(U) \tag{A.118}$$

From (A.118) by construction of $b_{\tau(U)}(\tau(\Pi))$ we have:

$$r \notin b_{\tau(U)}(\tau(\Pi)) \tag{A.119}$$

From (A.119) and (A.115) we have:

$$r \in \tau(\Pi) \setminus b_{\tau(U)}(\tau(\Pi)) \tag{A.120}$$

From (A.120) and (A.118) by definition of top we have (A.116).

Since Y' is an answer set of $e_{\tau(U)}(\tau(\Pi) \setminus b_{\tau(U)}(\tau(\Pi)), X')$, from (A.116) we have:

$$Y' \text{ satisfies } r \tag{A.121}$$

From (A.113) and (A.108) we have that Y' satisfies the body of r. Therefore, from (A.121) we have Y' also satisfies the head of the rule, which is $\neg f(\bar{x}, y_1)$. Therefore, (A.114) holds.

(e) We show that $\langle X, Y \rangle$ is a solution to Π with respect to U. We prove the clauses 1-3 of Definition 60 in i - iii respectively.

i. We show that X is a possible world of $b_U(\Pi)$. By construction,

$$X' \text{ is an answer set of } b_{\tau(U)}(\tau(\Pi)) \tag{A.122}$$

From (c), the fact that X' is an answer set of $b_{\tau(U)}(\tau(\Pi))$, A.40 by lemma 4 we have that X is a possible world of $b_U(\Pi)$.

ii. We show that Y is a possible world of $e_U(\Pi \setminus b_U(\Pi), X)$.
Recall that:

$$Y' \text{ is an answer set of } e_{\tau(U)}(\tau(\Pi) \setminus b_{\tau(U)}(\tau(\Pi)), X') \tag{A.123}$$

From (A.123), (d), and (A.436) by Lemma 4 we have that Y is a possible world of $e_U(\Pi \setminus b_U(\Pi), X)$.

iii. We prove $X \cup Y$ is consistent. Recall that $\tau(A) = X' \cup Y'$ for a possible world A of Π. From (c) and (d) we have that $X \cup Y$ consists of all positive literals which are members of $X' \cup Y'$, which is precisely A. Since A is a possible world, A is consistent, as well as $X \cup Y$.

(f) In 2.e).iii we have already shown that $A = X \cup Y$.

By theorem 4 and the fact that if Y contains an atom $f(x, y)$, it also contains atoms $f(x, y_1)$ for every $y_1 \neq y$, there exists Y' such that $Y = \tau(Y')$ and Y' is a possible world of $e_U(\Pi \setminus b_U(P), X)$.

5. $\leftarrow$ we show that if $A = X \cup Y$ for some solution $\langle X, Y \rangle$ to Π with respect to U, then A is a possible world of Π.

By 1, $\tau(U)$ is a splitting set of $\tau(\Pi)$. We prove that $\langle\tau(X), \tau(Y)\rangle$ is a solution to $\tau(\Pi)$ with respect to $\tau(U)$. In (a) we show that $\tau(X)$ is a possible world of $b_{\tau(U)}(\tau(\Pi)$. In (b) we show $\tau(Y)$ is a possible world of $e_{\tau(U)}(\tau(\Pi) \setminus b_{\tau(U)}(\tau(\Pi)), X')$. In (c) we show $\tau(X) \cup \tau(Y)$ is consistent.

(a) We show that

$$\tau(X) \text{ is a possible world of } b_{\tau(U)}(\tau(\Pi)) \tag{A.124}$$

Since $\langle X, Y\rangle$ is a solution of Π w.r.t. U, we have

$$X \text{ is a possible world of } b_U(\Pi) \tag{A.125}$$

Therefore by Lemma 4 we have:

$$\tau(X) \text{ is a possible world of } \tau(b_U(\Pi)) \tag{A.126}$$

From (A.40) and (A.126) we have (A.124)

(b) We show that

$$\tau(Y) \text{ is a possible world of } e_{\tau(U)}(\tau(\Pi) \setminus b_{\tau(U)}(\tau(\Pi)), \tau(X)) \tag{A.127}$$

Since $\langle X, Y\rangle$ is a solution of Π w.r.t. U, we have:

$$Y \text{ is a possible world of } e_U(\tau(\Pi) \setminus b_U(\tau(\Pi)), X) \tag{A.128}$$

Therefore by Lemma 4 we have:

$$\tau(Y) \text{ is a possible world of } \tau(e_U(\tau(\Pi) \setminus b_U(\tau(\Pi)), X)) \tag{A.129}$$

From (A.436) and (A.129) we have (A.127)

(c) Since $\langle X, Y \rangle$ is a solution of Π w.r.t. U, $X \cup Y$ is consistent. Therefore, by construction, $\tau(X) \cup \tau(Y)$ is consistent (it consists of atoms from X and Y and all negative literals of the form $f = y$ for every atom $f = y_1$ where $y \neq y_1$).

□

The original paper also contains an analogy of the following claim that we will use in the proof of Theorem 1.

Lemma 5. Let U be a splitting set for a program Π. If A is a possible world of Π such that $A = X \cup Y$ for some solution $\langle X, Y \rangle$ to Π with respect to U, then $Y \cap U = \emptyset$.

□

Proof. Since Y is a possible world of $e_U(\Pi \setminus b_U(\Pi), X)$, and, by clause 2 of Definition 59, the signature of $e_U(\Pi \setminus b_U(\Pi), X)$ does not contain literals from U, Y does not contain literals from U. Therefore, $Y \cap U = \emptyset$. □

A.2.3 Proof of Theorem 1

In this section we will prove the theorem:

Theorem 1.

Every program from $\mathcal{B}$ is coherent.

□

The outline of the proof is the same as that of Theorem 1 from [Baral et al., 2009], which says that a program from a different class introduced there is coherent. First, [Baral et al., 2009] introduces the notion of a tableau representing a program and shows that programs considered there can be represented by such tableaux. The second part of the proof consists of the theorem which states that every program which can be represented by a tableau is coherent. The definition of a tableaux and the corresponding theorem about coherency in our proof is very close to that in [Baral et al., 2009]. The only changes are those related to our refinement of the semantics of the original P-log. However, proof of the first part requires a substantial amount of work and new insights, given in lemmas 8 - 27 below.

Definition 62 (Unitary Tree)**.**
Let T be a tree in which every arc is labeled with a real number in [0,1]. We say T is *unitary* if the labels of the arcs leaving each node add up to 1.

□

Figure A.1 gives an example of a unitary tree.

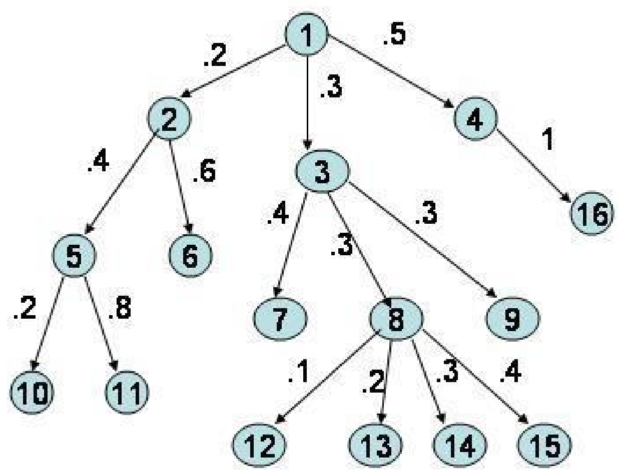

Figure A.1: Unitary tree T

Definition 63 ($p_T(n)$)**.**

Let T be a tree with labeled nodes and n be a node of T. By $p_T(n)$ we denote the set of labels of nodes lying on the path from the root of T to n, including the label of n and the label of the root.

□

Example 18. Consider the tree T from Figure A.1. If n is the node labeled (13), then $p_T(n) = \{1, 3, 8, 13\}$.

□

Definition 64 (Path value)**.**
Let T be a tree in which every arc is labeled with a number in [0,1]. The *path value* of a node n of T, denoted by $pv_T(n)$, is defined as the product of the labels of the arcs in the path to n from the root. (Note that the path value of the root of T is 1.)

□

When the tree T is obvious from the context we will simply write $pv(n)$.

Example 19. Consider the tree T from Figure A.1. If n is the node labeled with 8, then $pv(n) = 0.3 \times 0.3 = 0.09$.

□

Lemma 6. *[Property of Unitary Trees]*
Let T be a unitary tree and n be a node of T. Then the sum of the path values of all the leaf nodes descended from n (including n if n is a leaf) is the path value of n.

□

The proof of Lemma 6 can be found in [Baral et al., 2009].

Definition 65 (A set of literals compatible with an e-literal)**.**
A set S of literals of Π is Π-*compatible* with an e-literal l of Π if there exists a possible world of Π satisfying $S \cup \{l\}$. Otherwise S is Π-*incompatible* with l. S is Π-*compatible* with a set B of e-literals of Σ if there exists a possible world of Π satisfying $S \cup B$; otherwise S is Π-*incompatible* with B. □

Definition 66 (A set of literals guaranteeing an e-literal)**.**
A set S of literals is said to Π-*guarantee* an e-literal l if S and l are Π-compatible and every possible world of Π satisfying S also satisfies l; S Π-*guarantees* a set B of e-literals if S Π-guarantees every member of B. □

Definition 67 (Ready to branch)**.**

Let T be a tree whose nodes are labeled with atoms of Σ and r be a rule of Π of the form

$$random(a(\bar{t}) : \{X : p(X)\}) \leftarrow K.$$

where K can be empty. A node n of T is *ready to branch on* $a(\bar{t})$ *via* r *relative to* Π if

1. $p_T(n)$ contains no literal of the form $a(\bar{t}) = y$ for any y,

2. $p_T(n)$ Π-guarantees K,

3. for every pr-atom of the form $pr(a(\bar{t}) = y \mid B) = v$ in Π, either $p_T(n)$ Π-guarantees B or is Π-incompatible with B, and

4. for every $y \in range(a)$, $p_T(n)$ either Π-guarantees $p(y)$ or is Π-incompatible with $p(y)$ and moreover there is at least one $y \in range(a)$ such that $p_T(n)$ Π-guarantees $p(y)$.

If Π is obvious from the context we may simply say that n is ready to branch on $a(\bar{t})$ via r.

□

Proposition 14. Suppose n is ready to branch on $a(\bar{t})$ via some rule r of Π, and $a(\bar{t}) = y$ is Π-compatible with $p_T(n)$; and let W_1 and W_2 be possible worlds of Π-compatible with $p_T(n)$. Then $P(W_1, a(\bar{t}) = y) = P(W_2, a(\bar{t}) = y)$.

□

Proof. Suppose n is ready to branch on $a(\bar{t})$ via some rule r of Π, and $a(\bar{t}) = y$ is Π-compatible with $p_T(n)$; and let W_1 and W_2 be possible worlds of Π compatible with $p_T(n)$.

Case 1: Suppose $a(\bar{t}) = y$ has an assigned probability in W_1. Then there is a pr-atom $pr(a(\bar{t}) = y \mid B) = v$ of Π such that W_1 satisfies B. Since W_1 also satisfies

$p_T(n)$, B is Π-compatible with $p_T(n)$. It follows from the definition of ready-to-branch that $p_T(n)$ Π-guarantees B. Since W_2 satisfies $p_T(n)$ it must also satisfy B and so $P(W_2, a(\bar{t}) = y) = v$.

Case 2: Suppose $a(\bar{t}) = y$ does not have an assigned probability in W_1. Case 1 shows that the assigned probabilities for values of $a(\bar{t})$ in W_1 and W_2 are precisely the same; so $a(\bar{t}) = y$ has a default probability in both worlds. We need only show that the possible values of $a(\bar{t})$ are the same in W_1 and W_2. Suppose then that for some z, $a(\bar{t}) = z$ is possible in W_1. Then W_1 satisfies $p(z)$. Hence since W_1 satisfies $p_T(n)$, we have that $p_T(n)$ is Π-compatible with $p(z)$. By definition of ready-to-branch, it follows that $p_T(n)$ Π-guarantees $p(z)$. Now since W_2 satisfies $p_T(n)$ it must also satisfy $p(z)$ and hence $a(\bar{t}) = z$ is possible in W_2. The other direction is the same. □

Suppose n is ready to branch on $a(\bar{t})$ via some rule r of Π, and $a(\bar{t}) = y$ is Π-compatible with $p_T(n)$, and W is a possible world of Π compatible $p_T(n)$. We may refer to the $P(W, a(\bar{t}) = y)$ as $v(n, a(\bar{t}), y)$. Though the latter notation does not mention W, it is well defined by proposition 14.

Example 20. *[Ready to branch]*

Consider the following version of the dice example. Lets refer to it as Π_{16}

```
#dice = {d1,d2}.
#score = 1..6.
#person = {mike, john}.
roll: #dice -> #score.
owner: #dice -> #person.
owner(d1) = mike.
owner(d2) = john.
even(D) :- roll(D)= Y, Y mod 2 = 0.
-even(D) :- not even(D).
random(roll(D)).
```

```
pr(roll(D)=Y | owner(D) = john) = 1/6.
pr(roll(D)=6 | owner(D) = mike) = 1/4.
pr(roll(D)=Y | Y != 6, owner(D) = mike) = 3/20.
```

Now consider a tree T_2 of Figure A.2 [1]. Let us refer to the root of this tree as n_1,

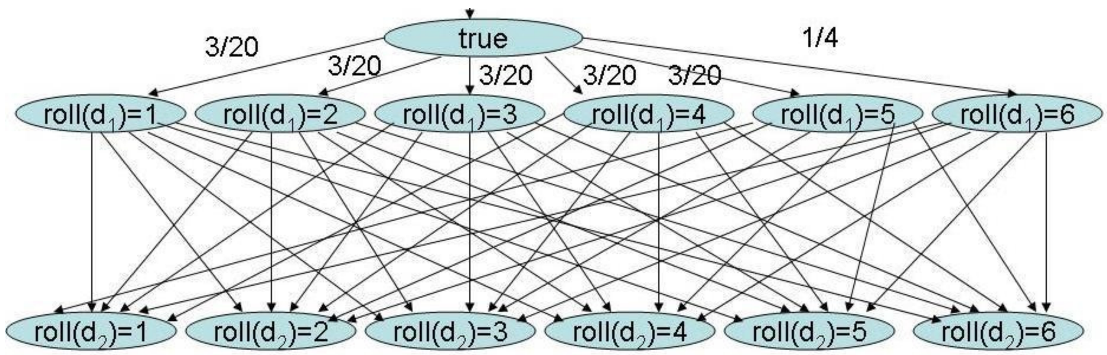

Figure A.2: T_2: The tree corresponding to the dice P-log program Π_{16}

the node $roll(d_1) = 1$ as n_2, and the node $roll(d_2) = 2$ connected to n_2 as n_3. Then $p_{T_2}(n_1) = \{true\}$, $p_{T_2}(n_2) = \{true, roll(d_1) = 1\}$, and $p_{T_2}(n_3) = \{true, roll(d_1) = 1, roll(d_2) = 2\}$. The set $\{true\}$ of literals Π_{16}-guarantees $\{owner(d_1) = mike$, $owner(d_2) = john\}$ and is Π_{16}-incompatible with $\{owner(d_1) = john, owner(d_2) = mike\}$. Hence n_1 and the attribute $roll(d_1)$ satisfy condition 3 of definition 67. Similarly for $roll(d_2)$. Other conditions of the definition hold vacuously and therefore n_1 is ready to branch on $roll(D)$ via $random(roll(D))$ relative to Π_{16} for $D \in \{d_1, d_2\}$. It is also easy to see that n_2 is ready to branch on $roll(d_2)$ via $random(roll(d_2))$, and that n_3 is not ready to branch on any attribute of Π_{16}.

□

Definition 68 (Expanding a node)**.**

In case n is ready to branch on $a(\bar{t})$ via some rule of Π, the Π-*expansion* of T at n by $a(\bar{t})$ is a tree obtained from T as follows: for each y such that $p_T(n)$ is Π-compatible

[1] The root is labeled with *true*, which can be viewed, for instance, as a true arithmetic atom $1 = 1$

with $a(\bar{t}) = y$, add an arc leaving n, labeled with $v(n, a(\bar{t}), y)$, and terminating in a node labeled with $a(\bar{t}) = y$. We say that n *branches on* $a(\bar{t})$.

□

Definition 69 (Expansions of a tree)**.**
A *zero-step* Π-expansion of T is T. A *one-step* Π-*expansion of* T is an expansion of T at one of its leaves by some attribute term $a(\bar{t})$. For $n > 1$, an *n-step* Π-*expansion* of T is a one-step Π-expansion of an $(n-1)$-step Π-expansion of T. A Π-*expansion* of T is an n-step Π-expansion of T for some non-negative integer n.

□

For instance, the tree consisting of the top two layers of tree T_2 from Figure A.2 is a Π_{16}-expansion of one node tree n_1 by $roll(d_1)$.

Definition 70 (Seed)**.**
A *seed* is a tree with a single node labeled *true*.

□

Definition 71 (Tableau)**.**
A *tableau* of Π is a Π-expansion of a seed which is maximal with respect to the subtree relation.

□

For instance, a tree T_2 of Figure A.2 is a tableau of Π_{16}.

Definition 72 (Node representing a possible world)**.**
Suppose T is a tableau of Π. A possible world W of Π is *represented* by a leaf node n of T if W is the set of atoms Π-guaranteed by $p_T(n)$.

□

For instance, a node n_3 of T_2 represents a possible world
$\{owner(d_1, mike), owner(d_2, john), roll(d_1, 1), roll(d_2, 2), \neg even(d_1), even(d_2)\}$.

Definition 73 (Tree representing a program)**.**
If every possible world of Π is represented by exactly one leaf node of T, and every leaf node of T represents exactly one possible world of Π, then we say T *represents* Π. □

It is easy to check that the tree T_2 represents Π_2.

Definition 74 (Probabilistic soundness)**.**
Suppose Π is a P-log program and T is a tableau representing Π, such that R is a mapping from the possible worlds of Π to the leaf nodes of T which represent them. If for every possible world W of Π we have

$$pv_T(R(W)) = \mu(W)$$

i.e. the path value in T of $R(W)$ is equal to the normalized measure of W, then we say that the representation of Π by T is *probabilistically sound*. □

The following lemma gives conditions sufficient for the coherency of P-log programs. It will later be shown that all unitary, dynamically causally ordered programs satisfy the hypothesis of this theorem, establishing Theorem 1.

Lemma 7 (Coherency Condition)**.**
Let Π be a program, and Π' be a program obtained from Π by removing activity records. If there exists a unitary tableau T representing Π', and this representation is probabilistically sound, then $P_{\Pi'}$ is defined, and for every pair of rules

$$random(a(\bar{t}) : \{X : p(X)\}) \leftarrow K. \tag{A.130}$$

and

$$pr(a(\bar{t}) = y \mid B) = v. \tag{A.131}$$

of Π' such that $P_{\Pi'}(B \cup K) > 0$ we have

$$P_{\Pi' \cup o(B) \cup o(K)}(a(\bar{t}) = y) = v$$

Hence Π is coherent.

$\square$

Proof. Since there exists a unitary tableau representing Π', by lemma 6 and Definition 74 we have that there exists at least one possible world with a non-zero measure. Therefore, $P_{\Pi'}$ is defined.

For any set S of literals, let $lgar(S)$ (pronounced "L-gar" for "leaves guaranteeing") be the set of leaves n of T such that $p_T(n)$ Π'-guarantees S.

Let μ denote the normalized measure on possible worlds induced by Π'.

Let Ω be the set of possible worlds of $\Pi' \cup o(B) \cup o(K)$. Since $P_{\Pi'}(B \cup K) > 0$ we have

$$P_{\Pi' \cup o(B) \cup o(K)}(a(\bar{t}) = y) = \frac{\sum_{\{W \,:\, W \in \Omega \,\wedge\, a(\bar{t}) = y \,\in\, W\}} \mu(W)}{\sum_{\{W \,:\, W \in \Omega\}} \mu(W)} \tag{A.132}$$

Now, let

$$\alpha = \sum_{n \in lgar(B \cup K \cup \{a(\bar{t}) = y)\}} pv(n)$$

$$\beta = \sum_{n \in lgar(B \cup K)} pv(n)$$

Since T is a probabilistically sound representation of Π', the right-hand side of (A.132) can be written as α/β. So we will be done if we can show that $\alpha/\beta = v$.

We first claim

$$\text{Every } n \in lgar(B \cup K) \text{ has a unique ancestor } ga(n) \text{ which branches on } a(\bar{t}) \text{ via } rule \text{ (A.130).} \tag{A.133}$$

If existence failed for some leaf n then n would be ready to branch on $a(\bar{t})$ which contradicts maximality of the tree. Uniqueness follows from Condition 1 of Definition 67.

Next, we claim the following:

$$\text{For every } n \in lgar(B \cup K),\ p_T(ga(n)) \ \Pi'\text{-guarantees } B \cup K. \tag{A.134}$$

Let $n \in lgar(B \cup K)$. Since $ga(n)$ branches on $a(\bar{t})$, $ga(n)$ must be ready to branch on $a(\bar{t})$ via a rule of Π'. So by clause 3 of Definition 67, $ga(n)$ either Π'-guarantees B or is Π'-incompatible with B. But $p_T(ga(n)) \subset p_T(n)$, and $p_T(n)$ Π'-guarantees B, so $p_T(ga(n))$ cannot be Π'-incompatible with B. Hence $p_T(ga(n))$ Π'-guarantees B. It also follows from clause 2 of Definition 67 that $p_T(ga(n))$ Π'-guarantees K.

From (A.134), it follows easily that

$$\text{If } n \in lgar(B \cup K),\ \text{every leaf descended from of } ga(n) \text{ belongs to } lgar(B \cup K). \tag{A.135}$$

Let

$$A = \{ga(n) : n \in lgar(B \cup K)\}$$

In light of (A.133) and (A.135), we have

$$lgar(B \cup K) \text{ is precisely the set of leaves descended from nodes in } A. \tag{A.136}$$

Therefore,

$$\beta = \sum_{n \text{ is a leaf descended from some } a \in A} pv(n)$$

Moreover, by construction of T, no leaf may have more than one ancestor in A, and hence

$$\beta = \sum_{a \in A} \sum_{n \text{ is a leaf descended from } a} pv(n)$$

Now, by Lemma 6 on unitary trees, since T is unitary,

$$\beta = \sum_{a \in A} pv(a)$$

This way of writing β will help us complete the proof. Now for α. Recall the definition of α:

$$\alpha = \sum_{n \in lgar(B \cup K \cup \{a(\bar{t})=y\})} pv(n)$$

Denote the index set of this sum by $lgar(B, K, y)$. Let

$$A_y = \{n : parent(n) \in A, \text{ the label of } n \text{ is } a(\bar{t}) = y\}$$

Since $lgar(B, K, y)$ is a subset of $lgar(B \cup K)$, (A.136) implies that $lgar(B, K, y)$ is precisely the set of nodes descended from nodes in A_y. Hence

$$\alpha = \sum_{n' \text{ is a leaf descended from some } n \in A_y} pv(n')$$

Again, no leaf may descend from more than one node of A_y, and so by the lemma on unitary trees,

$$\alpha = \sum_{n \in A_y} \sum_{n' \text{ is a leaf descended from } n} pv(n') = \sum_{n \in A_y} pv(n) \tag{A.137}$$

Finally, we claim that every node n in A has a unique child in A_y, which we will label $ychild(n)$. The existence and uniqueness follow from (A.134), along with Condition 3 of Section 2.2.3, and the fact that every node in A branches on $a(\bar{t})$ via rule A.130. Thus from (A.137) we obtain

$$\alpha = \sum_{n \in A} pv(ychild(n))$$

Note that if $n \in A$, the arc from n to $ychild(n)$ is labeled with v. Now we have:

$$P_{\Pi' \cup obs(B) \cup obs(K)}(a(\bar{t}) = y)$$

$$= \alpha/\beta$$

$$= \sum_{n \in A} pv(ychild(n)) / \sum_{n \in A} pv(n)$$

$$= \sum_{n \in A} pv(n) * v / \sum_{n \in A} pv(n)$$

$$= v.$$

□

Lemma 8. *[Tableau for programs from $\mathcal{B}$]*
Suppose Π is a program from $\mathcal{B}$ and U is the set of activity records in Π; then there exists a tableau T of $\Pi \setminus U$ which represents $\Pi \setminus U$ such that the representation of $\Pi \setminus U$ by T is probabilistically sound.

□

Proof. Let $\alpha = a_1 \ldots, , a_k$ be a probabilistic leveling of Π s.t. Π is dynamically causally ordered via α and $\Pi_0, \ldots, \Pi_k$ be the dynamic structure of $\Pi \setminus U$ induced by α. Let W_0 be the possible world of Π_0.

Consider a sequence $T_0, \ldots, T_k$ of trees where T_0 is a tree with one node, n_0, labeled by *true*, and T_i is obtained from T_{i-1} by expanding every leaf of T_{i-1} which is ready to branch on $a_i(\bar{t}_i)$ via any rule relative to Π_i by this term. Let $T = T_k$. We will show that T_m is a tableau of Π which represents Π and the representation is probabilistically sound.

Our proof will unfold as a sequence of lemmas: Let Σ_i be the signature of Π_i for every $i \in \{0..k\}$, and L_i be the set of all e-literals that can be formed by attribute terms from the signature of Π_i. □

Lemma 9. Let Π be a P-log program with signature Σ, A be a set of attribute terms of Σ, L be the set of all e-literals of Σ formed by attribute terms from A. Suppose there exists a set of atoms $W_L \subseteq L$ such that every possible world W of Π, $W_L \subseteq W$ and $(W \setminus W_L) \cap L = \emptyset$. Let $R \subseteq \Pi$ be a subset of rules of Π such that for every $r \in R$, the body of r contains an e-literal from L which is not satisfied by W_L. We have:

$$\Omega_\Pi = \Omega_{\Pi \setminus R} \tag{A.138}$$

□

Proof. In 1 we will prove $\Omega_\Pi \subseteq \Omega_{\Pi \setminus R}$. In 2 we will prove $\Omega_{\Pi \setminus R} \subseteq \Omega_\Pi$. (A.138) follows immediately from 1 and 2.

1. We prove

$$\Omega_\Pi \subseteq \Omega_{\Pi \setminus R} \tag{A.139}$$

Let $W \in \Omega_\Pi$ be a possible world of Π. We will show

$$W \in \Omega_{\Pi \setminus R} \tag{A.140}$$

Consider the reduct $\Pi' = (\Pi \setminus R)^W$. To show (A.140), in 1.1 we will show W satisfies the rules of $(\Pi \setminus R)^W$. In 1.2 we will prove W is minimal such set.

1.1 We show W satisfies the rules of $(\Pi \setminus R)^W$. Since $W \in \Omega_\Pi$, W satisfies Π^W. Since $(\Pi \setminus R)^W \subseteq \Pi^W$, W satisfied the rules of $(\Pi \setminus R)^W$.

1.2 For the sake of contradiction, suppose there exists $W' \subsetneq W$ such that W' satisfies $(\Pi \setminus R)^W$. We will show that

$$W' \text{ satisfies } \Pi^W \tag{A.141}$$

W' satisfies the subset $(\Pi \setminus R)^W$ of Π^W. Now suppose $r \in R^W$. Let r^* be the rule of Π which produced r in Π^W. By construction of R, the body of

r^* contains an e-literal l formed by an attribute term from A not satisfied by W_L. Since $W_L \subseteq W$ and $(W \setminus W_L) \cap L = \emptyset$, L is the set of all e-literals formed by attribute terms from A, the body of r^* contains l which is not satisfied by W. l cannot have default negation (or else, the rule r shouldn't belong to the reduct R^W). Therefore, l belongs to the body of r. Since $W' \subsetneq W$, and all the literals in the body of r do not contain default negation, l is not satisfied by W'. Therefore, W' satisfies r.

2. We prove

$$\Omega_{\Pi \setminus R} \subseteq \Omega_{\Pi} \tag{A.142}$$

Let $W \in \Omega_{\Pi \setminus R}$ be a possible world of $\Pi \setminus R$. We will show

$$W \in \Omega_{\Pi} \tag{A.143}$$

Consider the reduct $\Pi' = (\Pi)^W$. To show (A.143), in 2.1 we will show W satisfies the rules of $(\Pi)^W$. In 2.2 we will prove W is minimal such set.

2.1 Since $W \in \Omega_{\Pi \setminus R}$, it satisfies the rules of $(\Pi \setminus R)^W$. The further reasoning is similar to 1.2.

2.2 For the sake of contradiction suppose there exists $W' \subsetneq W$ which satisfies $(\Pi)^W$. Since $(\Pi \setminus R)^W \subseteq (\Pi)^W$, W' also satisfies $(\Pi \setminus R)^W$, which is a contradiction to the fact that W is a possible world of $(\Pi \setminus R)$.

□

Lemma 10. Let Π be a program with signature Σ such that the base of Π has a unique possible world. We have $\Omega_{red(\Pi)} = \Omega_{\Pi}$. □

Proof. Let L_0' be the set of literals of Σ each of which does not depend on an a random attribute term of Π. L_0' is a splitting set of Π. therefore, by splitting set theorem, for every possible world W of Π we have $W'_0 \subseteq W$ and $(W \setminus W'_0) \cap L'_0 = \emptyset$. By construction of $red(\Pi)$, $\Pi = red(\Pi) \cup R$, where the body of every rule in R contains a e-literal from L'_0 not satisfied by W'_0. Then the lemma follows trivially from lemma 9. □

Lemma 11. Let $0 \leq i \leq k$ be an integer. $\Omega_{red(\Pi_i)} = \Omega_{\Pi_i}$. □

Proof. Since $b_{L'_0}(\Pi_i) = b_{L'_0}(\Pi)$, $b_{L'_0}(\Pi_i)$ is the base of Π which has a unique possible world W'_0. Then the lemma follows immediately from Lemma 10. □

Lemma 12. Let $0 \leq i \leq k$ be an integer. Let W_i be a possible world of Π_i. We have:

1. $W_0 \subseteq W_i$
2. $(W_i \setminus W_0) \cap L_0 = \emptyset$

Proof. Let L'_0 be the set of literals from Π's base signature, and W'_0 be the answer set of Π's base. By Lemma 11

$$\Omega_{red(\Pi_i)} = \Omega_{\Pi_i} \tag{A.144}$$

Therefore, $W_i \in \Omega_{red(\Pi)}$. The lemma follows from the fact that L_0 is a splitting set of $red(\Pi_i)$, and $red(\Pi_0) = b_{L_0}(red(\Pi_i))$, and Lemma 11.

□

Lemma 13. Consider integers n, m such that $0 \leq n \leq m \leq k$. If W_m is a possible world of Π_m, then there exists a unique possible world W_n of Π_n such that $W_n \subseteq W_m$, and $(W_m \setminus W_n) \cap L_n = \emptyset$.

□

Proof. In the first part of the proof we show the existence of W_n. We start from two special cases.

- *Case 1.* $n = m$. The claim clearly holds, we can have $W_n = W_m$.

- *Case 2.* $n = 0$. That is, we prove that if W_m is a possible world of the program Π_m, then there exists a unique possible world W_0 of the program Π_0 such that $(W_m \setminus W_0) \cap L_0 = \emptyset$.

  From the definition of a dynamically causally ordered program, Π_0 has a unique possible world W_0. Therefore, from Lemma (12) every possible world A_m of the program Π_m can be written as $W_0 \cup Y$, for some Y such that $Y \cap L_0 = \emptyset$.

The proof is by double induction on n, m.

1. (*Base case* $n = m = 0$) The case follows immediately from *Case* 1.

2. (*Inductive Hypothesis*) Let h and j be two integers in the range $\{0..k\}$ such that $h \geq j > 0$. Let d and g be a pair of integers such that

$$d \leq j,$$

$$g \leq h,$$

$$d \leq g$$

   and

$$d + g < h + j.$$

   For every possible world W_d of the program Π_d there exists a possible world W_g of the program Π_g such that $W_d = W_g \cup U_g$, where $U_g \cap L_g = \emptyset$

3. (*Inductive Step*) We prove that for every possible world W_h of the program Π_h there exists a possible world W_j of the program Π_j such that $W_h = W_j \cup U_j$ where $U_j \cap L_j = \emptyset$. Let W_j be the set $W_h|_{L_j}$ we prove that W_j is a possible world of Π_j. In a) we show that W_j satisfies the rules of $\Pi_j^{W_j}$ and in b) we show that W_j is minimal.

   (a) We show that W_j satisfies the rules of $\Pi_j^{W_j}$. Let r be a rule of $\Pi_j^{W_j}$ such that the body of r is satisfied by W_j. We prove that the head of r is satisfied by W_j. Let r' be the rule of Π_j from which r was obtained during the computation of $\Pi_j^{W_j}$. Since $W_h \setminus W_j$ does not contain literals from L_j, and the rules of the program Π_j is a subset of the rules of the program Π_h, r' belongs to the rules of Π_h, and the reduct Π^{W_h} will contain the rule r. Since $W_j \subset W_h$, W_h satisfies the body of r. Therefore, Since W_h is an answer set of Π_h, the head of r is included into W_h. Since r belongs to $\Pi_j^{W_j}$, its head must belong to L_j. Since $W_j = W_h|_{L_j}$, the head of r also belongs to W_j. This means that W_j satisfies the head of r.

(b) We show that W_j is minimal. That is, there does not exist an interpretation W_j' of Π_j such that

$$W_j' \subsetneq W_j \tag{A.145}$$

and W_j' satisfies the rules of $\Pi_j^{W_j}$. We prove by contradiction. Suppose such an interpretation exists. Let's define U_j and W_h' as follows:

$$U_j = W_h \setminus W_j \tag{A.146}$$

$$W_h' = W_j' \cup U_j \tag{A.147}$$

From (A.146) we have:

$$U_j \cap W_j = \emptyset \tag{A.148}$$

From (A.148) and (A.145) we have:

$$U_j \cap W_j' = \emptyset \tag{A.149}$$

From (A.145) - (A.149) we have:

$$W_h' \subsetneq W_h \tag{A.150}$$

We show that W'_h satisfies the rules of $\Pi_h^{W_h}$, thus, obtaining a contradiction to the fact that W_h is a possible world of Π_h. Let r be a rule of $\Pi_h^{W_h}$ such that W'_h satisfies the body of r. We prove that W'_h satisfies the head of r. Let r' be the rule of Π_h from which r was obtained during the computation of $\Pi_h^{W_h}$. We consider two possible cases.

i. r' is a rule of Π_j. In this case r must belong to $\Pi_j^{W_j}$. Since W'_j satisfies the rules of $\Pi_j^{W_j}$, and it satisfies the body of r, it must satisfy the head of r.

ii. r' is not a rule of Π_j. We show that W_h satisfies the body of r'. First, since r belongs to the reduct $\Pi_h^{W_h}$, $\{not\ l \mid l \in neg(r)\}$ must be satisfied by W_h. Since W'_h satisfies the body of r, which is precisely $pos(r)$, and $W'_h \subsetneq W_h$, W_h satisfies $pos(r)$ too. This means

$$W_h \text{ satisfies the body of } r' \tag{A.151}$$

Let us denote the head of r' by l_0. Since W_h is a possible world of Π_h, from (A.151) we have

$$W_h \text{ satisfies } l_0 \tag{A.152}$$

We consider two cases:

A. l_0 is a member of L_j. We first prove that l_0 must be of one of the forms $random(a_q, p)$ or $a_q = y$ for some random attribute term a_q, where $q \leq j$. We prove by contradiction. Suppose l_0 is either formed by a random attribute term a_r, where $r > j$, or it is formed by a non-random attribute term $random(a_r, p)$, where $r > j$, or it is formed by a non-random attribute term b whose attribute is not $random$. The first two cases are clearly impossible, because L_j contains only attribute terms $a_0, \ldots, a_j$ and $random(a_s, p)$ for $0 \leq s \leq r$, and l_0 belongs to L_j. Consider the latter case, where l_0

is formed by non-random attribute term b whose attribute is not *random*. We show that, in this case, the level of attribute term b in Π must exceed j, thus, obtaining a contradiction to the fact that l_0 is a member of L_j.

- We show by contradiction that the rule r' belongs to $red(\Pi)$. Suppose that r' does not belong to $red(\Pi)$. This implies that there is an extended literal el in the body of r' formed by an atom in the signature of Π_0 such that W_0 does not satisfy el. By case #2, we get that the possible world W_h can be written as $W_0 \cup U'$, where $U' \cap L_0 = \emptyset$. Therefore, W_h does not satisfy the literal el, which contradicts A.151.
- We show that the level of $body(r')$ in Π must exceed j. Suppose the level of $body(r')$ does not exceed j. In this case, if both b and $body(r')$ have level $\leq j$, the rule r' must belong to Π_j, which contradicts our previous assumption. Thus, since r' belongs to $red(\Pi)$, b must have a level $> j$.

Thus, we are left with the two cases when

$$l_0 \text{ is either formed by } a_q \text{ or is of the form } random(a_q, p) \tag{A.153}$$

for random attribute term $a_q \in \{a_1 \ldots, a_j\}$. By inductive hypothesis, there exists a possible world W_{q-1} of the program Π_{q-1} such that

$$W_h = W_{q-1} \cup U_{q-1}, \tag{A.154}$$

and

$$U_{q-1} \cap L_{q-1} = \emptyset \tag{A.155}$$

Since r' does not belong to Π_j,

$$r' \text{ contains at least one literal which does not belong to } L_j. \tag{A.156}$$

Since $q \leq j$,

$$L_q \subseteq L_j. \tag{A.157}$$

From (A.156) and (A.157) it follows that

$$r' \text{ contains at least one literal which does not belong to } L_q \tag{A.158}$$

Since the head of r' is formed by a_q, from (A.158) it follows that

$$\text{the body of } r' \text{ contains a literal which does not belong to } L_q \tag{A.159}$$

From (A.159) it follows that

$$W_{q-1} \text{ does not satisfy the body of } r'. \tag{A.160}$$

Therefore, by clause 1 of Definition 20 from (A.160) it follows that we have only of the two cases:

- W_{q-1} falsifies the body of r'
  which means that the body of r' contains an extended literal el_{q-1} from the signature of Π_{q-1} such that W_{q-1} does not satisfy it. From (A.154) and (A.155) it follows that W_h does not satisfy el_{q-1}, and, therefore, it does not satisfy the *body*(r'). Therefore, we have a contradiction to (A.151).
- r' is a general axiom, which is, given that it's head is either

$random(a_q, p)$ or a_q, must be of the form:

$$a_q = y \leftarrow random(a_q, p), not\ a_q = y_1, \ldots, not\ a_q = y_k \quad \text{(A.161)}$$

In this case, the level of all attribute terms in r' is $q \leq j$, and, therefore, the rule r' must belong to Π_j. Contradiction to the main assumption in ii.

B. l_0 is not a member of L_j. In this case, since, by (A.152), W_h satisfies l_0 and $W_h = W_j \cup U_j$, and all the literals in W_j belong to L_j, l_0 belongs to U_j. Since $W'_h = W'_j \cup U_j$, W'_h satisfies l_0.

In the second part of the proof we show the uniqueness of W_j. Suppose there exist two different possible worlds W_j^1 and W_j^2 of Π_j such that

$$W_h = W_j^1 \cup U_j^1 \quad \text{(A.162)}$$

$$W_h = W_j^2 \cup U_j^2 \quad \text{(A.163)}$$

$$U_j^1 \cap L_j = \emptyset \quad \text{(A.164)}$$

$$U_j^2 \cap L_j = \emptyset \quad \text{(A.165)}$$

Since L_j contains all the literals that can be constructed from the signature of Π_j, from equations (A.162) and (A.164) it follows that

$$W_j^1 = W_h|_{L_j} \quad \text{(A.166)}$$

Similarly, from equations (A.163) and (A.165) it follows that

$$W_j^2 = W_h|_{L_j} \quad \text{(A.167)}$$

From equations (A.166) and (A.167) it follows that $W_j^1 = W_j^2$. This contradicts our original assumption that W_j^1 and W_j^2 are two **different** possible world of Π_j.

□

Lemma 14. Let Π be a program with a possible world W. The program $\Pi \cup W$ has a unique possible world W.

Proof. Since W is a possible world of Π, W satisfies Π^W. Therefore, W satisfies $(\Pi \cup W)^W = \Pi^W \cup W^W = \Pi^W \cup W$. W is minimal, because, by Proposition 1, every possible world of $\Pi^W \cup W$ must include W.

W is the only possible world of $\Pi \cup W$, because, by Proposition 1, every possible world of $\Pi \cup W$ must include W, and, by Proposition 13, no possible world which includes W and is different from W can exist.

□

Lemma 15. Let $i \in \{0..k-1\}$ be an integer and V be a possible world of Π_i. Let Π' be a program from the set $\{\Pi_{i+1} \cup V \cup \{\leftarrow \textit{not } a_{i+1} = y\}, \Pi_{i+1} \cup V \cup \{a_{i+1} = y\}, \Pi_{i+1} \cup V\}$. For every possible world W of Π', $W_0 \subseteq W$ and $W \setminus W_0 \cap L_0 = \emptyset$.

□

Proof. Let L'_0 be the set of literals from the base S of Π', and W'_0 be the answer set of S. We first show

$$\Omega_{red(\Pi')} = \Omega_{\Pi'} \tag{A.168}$$

Since $b_{L'_0}(\Pi'_i) = bot_{L'_0}(\Pi_i) \cup W'_0$, $b_{L'_0}(\Pi'_i)$ has a unique possible world W'_0. Therefore, by splitting set theorem, for every possible world W of Π'_i we have $W'_0 \subseteq W$ and $(W \setminus W'_0) \cap L'_0 = \emptyset$. By construction of $red(\Pi'_i)$, $\Pi = red(\Pi'_i) \cup R$, where the body of every rule in R contains a e-literal from L'_0 not satisfied by W'_0. (A.168) follows immediately from lemma 9.

Clearly, L_0 is a splitting set of $red(\Pi'_i)$, and $b_{L_0}(red(\Pi'_i)) = red(\Pi_0) \cup W_0$, and Lemma 11. By lemma (11), W_0 is a possible world of $red(\Pi_0)$. By Lemma 14, W_0 is a possible world of $red(\Pi_0) \cup W_0$. Therefore, by splitting set theorem, for every possible W of $red(\Pi'_i)$ we have $W_0 \subseteq W$ and $W \setminus W_0 \cap L_0$. Therefore, by (A.168), the lemma holds. □

Lemma 16. [2] Let $i \in \{0..k-1\}$ be an integer and V be a possible world of Π_i. Let Π' be a program from the set $\{\Pi_{i+1} \cup V \cup \{\leftarrow \textit{not } a_{i+1} = y\}, \Pi_{i+1} \cup V \cup \{a_{i+1} = y\}, \Pi_{i+1} \cup V\}$. For every possible world W of Π', $W \setminus V$ does not contain literals from L_i.

□

Proof. We define X as follows:

$$X = (W \setminus V)|_{L_i} \tag{A.169}$$

We show:

$$W \setminus X \text{ satisfies the rules of } \Pi^W_{i+1} \tag{A.170}$$

Let r be a rule of Π^W_{i+1}. We consider 2 cases:

1. Suppose the head of r is not a literal of X. If the body of r is satisfied by $W \setminus X$, it is also satisfied by W, thus, since W satisfies the rules of Π^W_{i+1}, it contains the head of r. Since the head of r does not belong to X, $W \setminus X$ satisfies head of r.

2. Suppose the head of r is formed by a literal from X. We need to show that, if the body of r is satisfied by $W \setminus X$, the head of r is also satisfied by $W \setminus X$. We consider two cases:

   (a) the head of r is of the form $a_j = y$, or of the form $random(a_j, p)$, for a random attribute term a_j, where $j \leq i$. By Lemma 13, there must exists a possible world V_{j-1} of Π_{j-1} such that

$$V_{j-1} \subseteq V \tag{A.171}$$

[2]the lemma is more general than it is required for this proof of coherency, however we will use it in section A.3 for another proof

and

$$(V \setminus V_{j-1}) \cap L_{j-1} = \emptyset \tag{A.172}$$

Let r' be the rule of Π_{i+1} from which r was obtained during the computation of the reduct Π_{i+1}^{W} By definition of dynamically causally ordered program, there are three cases:

i.
$$V_{j-1} \text{ satisfies the body of } r' \tag{A.173}$$

and

$$\text{all the literals occurring in } r' \text{ are in } L_{j-1} \tag{A.174}$$

Since $j \leq i$, $L_{j-1} \subseteq L_i$, we have

$$r' \text{ belongs to } \Pi_i \tag{A.175}$$

From (A.171), (A.172), (A.173) and (A.174) we have that V satisfies the body of r'.

Since V is a possible world of Π_i from (A.175) we have V contains the head of r' (and r, since the heads of r and r' are the same). Since $V \subseteq (W \setminus X)$, $(W \setminus X)$ also contains the head of r

ii. V_{j-1} falsifies the body of r'. Because $(V \setminus V_{j-1}) \cap L_{j-1} = \emptyset$, V falsifies the body of r'. That is, there exists an e-literal l belonging to the body of r' such that $l \in L_{j-1}$ and V does not satisfy l. Because $((W \setminus X) \setminus V) \cap L_i = \emptyset$, and $L_{j-1} \subseteq L_i$, we have that $((W \setminus X) \setminus V) \cap L_{j-1} = \emptyset$. Therefore, since V falsifies $body(r')$, $W \setminus X$ falsifies $body(r')$. If l does not contain default negation, this contradicts our original assumption that the body of r is satisfied by $W \setminus X$. Suppose now $l = not\ l'$, where $l' \in L_{j-1}$. Since V does not satisfy l, V satisfies l'. Since $V \subseteq W$ (in all 3 cases) , W satisfies l'. Therefore, W does not satisfy l. This contradicts the fact that r from the reduct Π^W_{i+1} is obtained from r'.

iii. r' is a general axiom of the form

$$a_j = y \leftarrow random(a_j, p), not\ a_j = y_1, \ldots, not\ a_j = y_k$$

and r is of the form

$$a_j = y \leftarrow random(a_j, p)$$

Since $W \setminus X$ satisfies the body of r, and all the literals of L_i from $W \setminus X$ are contained in V, we have

$$random(a_j, p) \in V \tag{A.176}$$

Since $V \subseteq W$, $random(a_j, p) \in W$. Since r belongs to the reduct Π^W_{i+1}, $\{a_j = y_1, \ldots, a_j = y_k\} \cap W = \emptyset$. Since $V \subseteq W$,

$$\{a_j = y_1, \ldots, a_j = y_k\} \cap V = \emptyset \tag{A.177}$$

Since $r' \in \Pi_i$ (all the literals are clearly in $L_j \subseteq L_i$), and V is a possible world of Π_i, from (A.176) and (A.177) we have $a_j = y \in V$. Since X does not contain literals from V, and $V \subseteq W$, $a_j = y \in W \setminus X$.

(b) the head of r is formed by a non-random attribute term nr, whose attribute is not $random$, and whose level in Π is $\leq i$. Let r' be the rule of Π^W_{i+1} from which r was obtained. We consider two cases:

i. r' does not belong to $red(\Pi)$. In this case the body of r' contains an e-literal $l \in L_0$ such that $|a| = 0$ and W_0 does not satisfy l.
By Lemma 15 we have:

$$W_0 \subseteq W \tag{A.178}$$

$$(W \setminus W_0) \cap L_0 = \emptyset \tag{A.179}$$

From (A.179) we have $((W \setminus X) \setminus W_0) \cap L_0 = \emptyset$. Therefore, since W_0 does not satisfy l and $l \in L_0$, $W \setminus X$ also does not satisfy l, which contradicts the fact that $W \setminus X$ satisfies the body of r.

ii. r' belongs to $red(\Pi)$ The body of r' must consist of literals in L_i (otherwise, the head of r will not belong to L_i, which contradicts the fact $l \in X$). Since $(W \setminus X) \setminus V$ does not contain literals from L_i, and $(W \setminus X)$ satisfies the body of r, V satisfies the body of r. Since r belongs to the reduct Π^W_{i+1}, W satisfies all extended literals of the body of r' preceded by default negation. Since $V \subseteq W$, V also satisfies all extended literals of the body of r' preceded by default negation. This means that V satisfies the body of r'. Since all the literals in r' are members of L_i, r' must belong to Π_i. Since V is a possible world of Π_i, it must satisfy r', thus, the head of r', which is the same as the head of r, must belong to V. Since $V \subseteq W \setminus X$, and V satisfies the head of r, $W \setminus X$ satisfies the head of r.

To conclude the proof, we consider the 3 possible values of Π' from the lemma separately and show $X = \emptyset$:

1. Suppose W is the possible world of $\Pi_{i+1} \cup V$. We need to show that $X = \emptyset$. For the sake of contradiction suppose $X \neq \emptyset$. We have previously shown that $W \setminus X$ satisfies the rules of Π^W_{i+1}. Since $V \subseteq W \setminus X$, $W \setminus X$ satisfies V. Therefore, $W \setminus X$ satisfies the rules of $\Pi^W_{i+1} \cup V^W$, which contradicts the fact that W is a possible world of $\Pi_{i+1} \cup V$.

2. Suppose W is the possible world of $\Pi_{i+1} \cup V \cup \{\leftarrow not\ a_{i+1} = y\}$. We show that $X = \emptyset$. For the sake of contradiction suppose $X \neq \emptyset$. We previously showed that $W \setminus X$ satisfies the rules of Π^W_{i+1}. Since $V \subseteq W \setminus X$, $W \setminus X$ satisfies

V. Since W is a possible world of $\Pi_{i+1} \cup V \cup \{\leftarrow not\ a_{i+1} = y\}$, W contains $a_{i+1} = y$. Therefore, $(\Pi_{i+1} \cup V \cup \{\leftarrow not\ a_{i+1} = y\})^W = (\Pi_{i+1} \cup V)^W$.Therefore, $(W \setminus X) \subsetneq W$ satisfies all the rules of $(\Pi_{i+1} \cup V \cup \{\leftarrow not\ a_{i+1} = y\})^W$, which contradicts the fact that W is a possible world of $(\Pi_{i+1} \cup V \cup \{\leftarrow not\ a_{i+1} = y\})$.

3. Suppose W is the possible world of $\Pi_{i+1} \cup V \cup \{a_{i+1} = y\}$. We show that $X = \emptyset$. For the sake of contradiction suppose $X \neq \emptyset$. We previously showed that $W \setminus X$ satisfies the rules of Π_{i+1}^W. Since $V \subseteq W \setminus X$, $W \setminus X$ satisfies V. Since W is a possible world of $\Pi_{i+1} \cup V \cup \{a_{i+1} = y\}$, W satisfies $a_{i+1} = y$. Since $a_{i+1} \in L_{i+1} \setminus L_i$, and X consists of literals from L_i, $W \setminus X$ satisfies $a_{i+1} = y$. Therefore, $W \setminus X$ satisfies $(\Pi_{i+1} \cup V \cup \{a_{i+1} = y\})^W$, which contradicts the fact that W is a possible world of $(\Pi_{i+1} \cup V \cup \{a_{i+1} = y\})$.

□

Lemma 17. Let i be an integer in the range $\{0..k-1\}$ and V be a possible world of Π_i.

1. if no random selection rule with a_{i+1} is active in V, then every possible world of $\Pi_{i+1} \cup V$ is a possible world of Π_{i+1},

2. if there is a random selection rule of the form

$$random(a_{i+1}, p) \leftarrow B \tag{A.180}$$

active in V, and $p(y) \in V$, then every possible world of $\Pi_{i+1} \cup V \cup \{\leftarrow not\ a_{i+1} = y\}$, is a possible world of Π_{i+1}.

□

Proof. Let Π_{ext} denote the program $\Pi_{i+1} \cup V$ in case 1 and and the program $\Pi_{i+1} \cup V \cup \{\leftarrow not\ a_{i+1} = y\}$ in case 2. Let W be a possible world of Π_{ext}. We show that W is a possible world of Π_{i+1}. We first show that W satisfies the rules of Π_{i+1}^W, and then we show that W is minimal.

1. We show that W satisfies the rules of Π^W_{i+1}. Let r be a rule of Π^W_{i+1} such that W satisfies the body of r. We need to show that W satisfies the head of r. Since W is a possible world of Π_{ext}, it satisfies the rules of Π^W_{ext}, which include the rules in Π^W_{i+1}.

2. We show that W is minimal. That is, there does not exist an interpretation W' such that W' satisfies the rules of Π^W_{i+1} and $W' \subsetneq W$. We prove by contradiction. Suppose such W' exists. We show that W' satisfies the rules of Π^W_{ext}, obtaining a contradiction to the fact that W is a possible world of Π_{ext}. By definition, W' satisfies the rules of Π^W_{i+1}. Therefore, since, in the second case of the Lemma $\{\leftarrow \mathit{not}\ a_{i+1} = y\}^W = \emptyset$, we just need to show that

$$W' \text{ satisfies } V^W \tag{A.181}$$

Since V is a collection of facts, we just need to show that $V \subset W'$. We prove by contradiction.

Suppose there is an atom $a = y$ of Π such that

$$a = y \in V \tag{A.182}$$

and

$$a = y \notin W' \tag{A.183}$$

Let us define V' to be:

$$V' = W'|_{L_i} \tag{A.184}$$

From (A.183) and (A.184) we have:

$$a = y \notin V' \tag{A.185}$$

By lemma 16 we have

$$W \setminus V \text{ does not contain literals from } L_i \tag{A.186}$$

Therefore, since V is a possible world of Π_i, we have:

$$V = W|_{L_i} \tag{A.187}$$

Since $W' \subsetneq W$, from (A.187) and (A.184) we have:

$$V' \subseteq V \tag{A.188}$$

From (A.188), (A.182) and (A.185) we have:

$$V' \subsetneq V \tag{A.189}$$

We next show

$$V' \text{ satisfies the rules of } \Pi^V \tag{A.190}$$

From (A.186) we have:

$$\Pi_i^V \subseteq \Pi_{i+1}^W \tag{A.191}$$

Let r be a member of Π_i^V such that V' satisfies the body of r. We show that V' satisfies the head of r. From (A.184) we have that the body of r is satisfied by W'.

Since W' satisfies the rules of Π_{i+1}^{W}, from (A.191) W' satisfies the head of r. Since r is a member of Π_i^V, $head(r) \in L_i$. Therefore, from (A.184), V' satisfies $head(r)$. Therefore, (A.190) holds, and, considering (A.188), we have a contradiction.

□

Lemma 18. For every $i \in \{0, \ldots, k\}$ and every leaf node n of T_i program Π_i has a unique possible world W satisfying $p_{T_i}(n)$. □

Proof. We use induction on i. The case where $i = 0$ follows from Condition 1 of Definition 17 of dynamically causally ordered program. Assume that the lemma holds for $i - 1$ and consider a leaf node n of T_i. By construction of T, there exists a leaf node m of T_{i-1} which is either the parent of n or equal to n. By inductive hypothesis there is a unique possible world V of Π_{i-1} containing $p_{T_{i-1}}(m) \setminus \{true\}$.

(i) First we will show that every possible world W of Π_i containing $p_{T_{i-1}}(m)$ also contains V. By lemma 13, set $V' = W|_{L_{i-1}}$ is a possible world of Π_{i-1}. Obviously, $p_{T_{i-1}}(m) \setminus \{true\} \subseteq V'$. By inductive hypothesis, $V' = V$, and hence $V \subseteq W$.

Now let us consider two cases.

(ii) For every random selection r rule of Π of the form

$$random(a_i : \{X : p(X)\}) \leftarrow K \tag{A.192}$$

V falsifies K. We will show that in this case m is not ready to branch on a_i w.r.t Π_i. It is sufficient to show that for every random selection rule of the form (A.192), V is not Π_i-compatible with K. Since V falsifies K, there exists an e-literal $l \in K$ such that:

$$V \text{ does not satisfy } l \tag{A.193}$$

$$l \in L_i \tag{A.194}$$

Let us show by contradiction. Suppose there exists a possible world W of Π_i such that

$$W \text{ satisfies } V \cup K \tag{A.195}$$

By lemma 13 we have:

$$(W \setminus V) \cap L_i = \emptyset \tag{A.196}$$

From (A.193) , (A.194) and (A.196) we have:

$$W \text{ does not satisfy } l \tag{A.197}$$

Therefore, since $l \in K$, we have a contradiction to (A.195). Therefore, m is not ready to branch on a_i w.r.t Π_i, and, by construction of T_i, $m = n$. By condition 3 of Definition 20, we have $V \cup \Pi_{i-1}$ has exactly one possible world, W. By lemma 17, W is a possible world of Π_i. Obviously, W contains V and hence $p_{T_{i-1}}(m)$. Since $n = m$ this implies that W contains $p_{T_i}(n)$.

Uniqueness follows immediately from (i) and Condition (3) of Definition 20.

(iii) There is a random selection rule r of the form (A.192) active in V.

We will show that m is ready to branch on a_i via rule r relative to Π.

Condition (1) of the definition of "ready to branch" (Definition 67) follows immediately from construction of T_{i-1}.

To prove Condition (2) we need to show that $p_{T_{i-1}}(m)$ Π_i-guarantees K. Since r is active in V, by Condition 1 of Definition 19 and Condition 2 of Definition 20 we have that there exists y_0 such that $p(y_0) \in V$ and $V \cup \Pi_{i+1}$ has a possible world containing $a = y_0$, say, W_0. Since r is active in V,

$$V \text{ satisfies } K \tag{A.198}$$

Therefore, by lemma 16, W_0 satisfies K. Since V contains $p_{T_{i-1}}(m)$, W_0 also contains $p_{T_{i-1}}(m)$. Therefore,

$$\text{V is } \Pi_i\text{-compatible with } K \tag{A.199}$$

Now consider a possible world W of Π_i which contains $p_{T_{i-1}}(m)$. By (i) we have that $V \subseteq W$. Since V satisfies K so does W (by lemma 13, $(W \setminus V) \cap L_{i-1} = \emptyset$). Condition (2) of the definition of ready to branch is satisfied.

To prove condition (3) consider $pr(a_i = y \mid B) = v$ from Π_i such that B is Π_i-compatible with $p_{T_{i-1}}(m)$. Π_i-compatibility implies that there is a possible world W_0 of Π_i which contains both, $p_{T_{i-1}}(m)$ and B. By (i) we have that $V \subseteq W_0$. By Lemma 13 we have that $(W \setminus V) \cap L_{i-1} = \emptyset$. From condition (1) of 20 it follows that either V satisfies B or V falsifies B. If V falsifies B, then W_0 does not satisfy B. Hence, V satisfies B. Since for every possible world W' of Π_i containing $p_{T_{i-1}}(m)$ we have that W' contains V and, by Lemma 13 $(W \setminus V) \cap L_{i-1} = \emptyset$, we have that W' satisfies B which proves condition (3) of the definition.

To prove Condition (4) we consider $y_0 \in range(a_i)$ such that $p(y_0) \in V$ (The existence of such y_0 is proven at the beginning of (iii)). We show that $p_{T_{i-1}}(m)$ Π_i-guarantees $p(y_0)$. Condition (2) of Definition 20 guarantees that Π_i has possible world, say W, containing V. By construction, $p(y_0) \in V$ and hence $p(y_0)$ and $p_{T_{i-1}}(m)$ are Π_i compatible. From (i) we have that $p_{T_{i-1}}(m)$ Π_i-guarantees $p(y_0)$. Similar argument shows that if $p_{T_{i-1}}(m)$ is Π_i-compatible with $p(y)$ then $p(y)$ is also Π_i-guaranteed by $p_{T_{i-1}}(m)$.

We can now conclude that m is ready to branch on a_i via rule r relative to Π_{i+1}. This implies that a leaf node n of T_i is obtained from m by expanding it by an atom $a_i = y$.

By Condition (2) of Definition 20, program $V \cup \Pi_i \cup \{\leftarrow not\ a_i = y_0\}$ has exactly one possible world, W. By lemma 17 we have that W is a possible world of Π_i. Clearly W contains $p_{T_i}(n)$. Uniqueness follows immediately from (i) and Condition

(2) of Definition 20.

□

Lemma 19. For all $i \in \{0..k\}$, every possible world of Π_i satisfies $p_{T_i}(n)$ for some unique leaf node n of T_i. □

Proof. We use induction on i. The case where $i = 0$ is immediate. Assume that the lemma holds for $i-1$, and consider a possible world W of Π_i. By Lemma 13, Π_{i-1} has a possible world V such that:

$$V \subseteq W \tag{A.200}$$

$$(W \setminus V) \cap L_{i-1} = \emptyset \tag{A.201}$$

By the inductive hypothesis there is a unique leaf node m of T_{i-1} such that V contains $p_{T_{i-1}}(m)$. Consider two cases.

(a) For every random selection rule

$$random(a_i : \{X : p(X)\}) \leftarrow K \tag{A.202}$$

K is falsified by V. In part ii of the proof of Lemma 18 we have shown that in this case m is not ready to branch on a_i. This means that m is a leaf of T_i and $p_{T_{i-1}}(m) = p_{T_i}(m)$. Let $n = m$. Since $V \subseteq W$ we have that $p_{T_i}(n) \subseteq W$. To show uniqueness suppose n' is a leaf node of T_i such that $p_{T_i}(n') \subseteq W$, and n' is not equal to n. By construction of T_i there is some j and some $y_1 \neq y_2$ such that $a_j = y_1 \in p_{T_i}(n')$ and $a_j = y_2 \in p_{T_i}(n)$. Since W is an interpretation, it is impossible.

(b) There is a random selection rule r of the form

$$random(a_i : \{X : p(X)\}) \leftarrow K \tag{A.203}$$

active in V. Since r is active in V, we have

$$V \text{ satisfies } K \tag{A.204}$$

and, therefore

$$\text{every literal from } K \text{ is in } L_{i-1} \tag{A.205}$$

From (A.204), (A.205), (A.200) and (A.201) we have:

$$W \text{ satisfies } K \tag{A.206}$$

From clause 2 of Definition 19 and the fact that $a_i \in L_i$ we have:

$$r \in \Pi_i \tag{A.207}$$

Since W is a possible world of Π_i, it must satisfies r together with a general axiom from Π_i:

$$a_i = y_1 \mid \ \dots \ \mid a_i = y_m \leftarrow random(a_i : \{X : p(X)\})$$
$$\leftarrow a_i = Y, not\ p(Y)$$

Therefore, since W satisfies the body of r, there exists $y \in range(a)$ such that

$$a_i = y \in W \tag{A.208}$$

and

$$p(y) \in W \tag{A.209}$$

Since r is active in V, by clause 2 of Definition 19 we must have $p(y) \in L_{i-1}$. From (A.209) and (A.201) we have:

$$p(y) \in V \tag{A.210}$$

Repeating the argument from part (iii) of the proof of Lemma 18 we can show that m is ready to branch on a_i via r relative to Π_i. Since $p_{T_{i-1}}(m) \subseteq V \subseteq W$, $p_{T_{i-1}}(m)$ is Π_i-compatible with $p(y)$. Thus, there is a leaf node n of T_i which is a son of m labeled with $a_i = y$. It is easy to see that W contains $p_{T_k}(n)$. The proof of uniqueness is similar to that used in (a).

□

Lemma 20. Let i, j be integers s.t. $0 < i \leq j \leq k$. For every leaf node n of T_{i-1}, every set B of extended literals of L_i, and we have $p_{T_{i-1}}(n)$ is Π_i-compatible with B iff $p_{T_{i-1}}(n)$ is Π_j-compatible with B.

□

Proof. $\rightarrow$

Suppose that $p_{T_{i-1}}(n)$ is Π_i-compatible with B. This means that there is a possible world V of Π_i which satisfies $p_{T_{i-1}}(n)$ and B. By Lemma 19, there exists a unique leaf node n' of T_i such that $p_{T_i}(n') \setminus \{true\} \subseteq V$. Consider a leaf node m of T_j belonging to a path containing node n' of T_i. By Lemma 18, Π_j has a unique possible world W containing $p_{T_j}(m)$. By lemma 13 $W = V' \cup U$ where V' is a possible world of Π_i and $U \cap L_i = \emptyset$. This implies that V' contains $p_{T_i}(n')$, and hence, by Lemma 18 $V' = V$. Since V satisfies B and $U \cap L_i = \emptyset$ we have that W also satisfies B. Since $p_{T_{i-1}}(n) \subseteq V \subseteq W$, we have $p_{T_{i-1}}(n)$ is Π_j-compatible with B.

$\leftarrow$

Let W be a possible world of Π_j satisfying $p_{T_{i-1}}(n)$ and B. By Lemma 13, we have that $W = V \cup U$ where V is a possible world of Π_i and $U \cap L_i = \emptyset$. Since B and $p_{T_{i-1}}(n)$ belong to the language of L_i we have that B and $p_{T_{i-1}}(n)$ are satisfied by V and hence $p_{T_{i-1}}(n)$ is Π_i-compatible with B.

□

Lemma 21. Let i, j be integers such that $0 < i \leq j \leq k$. For every leaf node n of T_{i-1}, every set B of extended literals of L_i, we have $p_{T_{i-1}}(n)$ Π_i-guarantees B iff

$p_{T_{i-1}}(n)$ Π_j-guarantees B. □

Proof. $\rightarrow$

Let us assume that $p_{T_{i-1}}(n)$ Π_i-guarantees B. This implies that $p_{T_{i-1}}(n)$ is Π_i-compatible with B, and hence, by Lemma 20 $p_{T_{i-1}}(n)$ is Π_j-compatible with B. Now let W be a possible world of Π_j satisfying $p_{T_i}(n)$. By Lemma 13 $W = V \cup U$ where V is a possible world of Π_i and $U \cap L_i = \emptyset$. This implies that V satisfies $p_{T_{i-1}}(n)$. Since $p_{T_{i-1}}(n)$ Π_i-guarantees B we also have that V satisfies B. Finally, since $U \cap L_i = \emptyset$ we can conclude that W satisfies B.

$\leftarrow$

Suppose now that $p_{T_{i-1}}(n)$ Π_j-guarantees B. This implies that $p_{T_{i-1}}(n)$ is Π_i-compatible with B. Now let V be a possible world of Π_i containing $p_{T_{i-1}}(n)$. By Lemma 19, there exists a unique leaf node n' of T_i such that

$$V \text{ satisfies } p_{T_i}(n') \tag{A.211}$$

To show that V satisfies B let us consider a leaf node m of a path of T_j containing n'. By Lemma 18 Π_j has a unique possible world W satisfying $p_{T_j}(m)$. By construction,

$$W \text{ satisfies } p_{T_i}(n') \tag{A.212}$$

By Lemma 13, $W = V' \cup U$ where V' is a possible world of Π_i and $U \cap L_i = \emptyset$. Since $p_{T_i}(n')$ is in L_i, we have:

$$V' \text{ satisfies } p_{T_i}(n') \tag{A.213}$$

From (A.211) and (A.213) we by Lemma 18 we have:

$$V = V' \tag{A.214}$$

Since $p_{T_{i-1}}(n) \setminus \{true\} \subseteq V = V' \subseteq W$, we have W satisfies $p_{T_{i-1}}(n)$. Therefore, since

$p_{T_{i-1}}(n)$ Π_j-guarantees B, W satisfies B. Since B belongs to the language of L_i it is satisfied by V'. Therefore, from $V' = V$ we have that V satisfies B and we conclude $p_{T_{i-1}}(n)$ Π_i-guarantees B.

□

Lemma 22. Let i, j be integers such that $0 < i \leq j \leq k$. Every leaf node n of T_{i-1}, n is ready to branch on term a_i relative to Π_i iff n is ready to branch on a_i relative to Π_j. □

Proof. $\rightarrow$

Suppose n is ready to branch on a_i via rule r

$$random(a_i : \{X : p(X)\}) \leftarrow K \tag{A.215}$$

relative to Π_i. We show that n is ready to branch on a_i via r relative to Π_j. We prove the conditions 1-4 of the definition:

1. Condition 1 follows immediately from the fact that n is ready to branch on a_i relative to Π_i.

2. We prove condition 2:

$$p_{T_{i-1}}(n) \text{ } \Pi_j\text{-guarantees } K \tag{A.216}$$

   By lemma 18, there is a unique possible world W_{i-1} of Π_{i-1} such that

$$W_{i-1} \text{ satisfies } p_{T_{i-1}}(n) \tag{A.217}$$

   We prove that

$$r \text{ is active in } W_{i-1} \tag{A.218}$$

   We prove by contradiction. Suppose r is not active in W_{i-1}. By condition 1 of 20 we have W_{i-1} falsifies K. That is, there is a literal $l \in L_{i-1}$ such that W_{i-1} does not satisfy l. Then, by conditions 2-3 of 20 and Lemma 17 we have that Π_i has a possible world W_i containing W_{i-1}. By Lemma 13, $W_i \setminus W_{i-1} \cap L_{i-1} = \emptyset$. Therefore, W_i does not satisfy l, and, therefore, K. This, given that $p_{T_{i-1}}(n) \setminus \{true\} \subseteq W_{i-1} \subseteq W_i$ contradicts the fact n is ready to branch on a_i via r relative to Π_i. Therefore, (A.218) holds. Therefore, the literals occurring in the body of r are from L_{i-1}, and by lemma (21) we conclude (A.216).

3. We prove condition 3. Let $pr(a_i = y \mid B) = v$ be a pr-atom from Π_j. We show that

$$p_{T_{i-1}}(n) \text{ either } \Pi_j\text{-guarantees } B \text{ or is } \Pi_j\text{-incompatible with } B \tag{A.219}$$

Since n is ready to branch on a_i via rule r relative to Π_i, we have 3 cases:

(a) $pr(a_i = y \mid B) = v$ is a pr-atom from Π_i, and B is Π_i-guaranteed by $p_{T_{i-1}}(n)$. Using the arguments similar to the ones from 2, we can obtain $B \in L_{i-1}$, and conclude by lemma (21) that $p_{T_{i-1}}(n)$ Π_j-guarantees B

(b) $pr(a_i = y \mid B) = v$ is a pr-atom from Π_i, and B is Π_i-incompatible with $p_{T_{i-1}}(n)$. That is,

$$\text{every possible world } W_i \text{ or } \Pi_i \text{ satisfying } p_{T_{i-1}}(n) \text{ does not satisfy } B \tag{A.220}$$

By lemma 18, there is a unique possible world W_{i-1} of Π_{i-1} such that

$$W_{i-1} \text{ satisfies } p_{T_{i-1}}(n) \tag{A.221}$$

We prove

$$W_{i-1} \text{ falsifies } B \tag{A.222}$$

For the sake of contradiction, suppose (A.222) is false. By condition 1 of definition (20) we have:

$$W_{i-1} \text{ satisfies } B \tag{A.223}$$

By 20 and lemma (17) we have that Π_i has a possible world W_i containing W_{i-1}. Since B is in L_i, by lemma (13) we have

$$W_i \text{ satisfies } B \tag{A.224}$$

Since $p_{T_{i-1}}(n) \setminus \{true\} \subseteq W_{i-1} \subseteq W_i$, we have a contradiction from (A.224) and (A.220).

Therefore, (A.222) holds. Now let W_j be a possible world of Π_j satisfying $p_{T_{i-1}}(n)$. By lemma (13), There is a possible W'_{i-1} of Π_{i-1} such that $W'_{i-1} \subseteq W_j$ and

$$(W'_{i-1} \cap W_j) \cap L_{i-1} = \emptyset \tag{A.225}$$

Since $p_{T_{i-1}}(n)$ is in L_{i-1}, we have $p_{T_{i-1}}(n) \subseteq W'_{i-1}$. Therefore, By lemma 18

$$W'_{i-1} = W_{i-1} \tag{A.226}$$

From (A.226), (A.222), (A.225) we have that W_j does not satisfy B. Therefore, $p_{T_{i-1}}(n)$ is Π_j-incompatible with B

(c) $pr(a_i = y \mid B) = v$ does not belong to Π_i. That is,

B contains an e-literal l

$$l \notin L_i \tag{A.227}$$

By lemma 18, there is a unique possible world W_{i-1} of Π_{i-1} such that

$$W_{i-1} \text{ satisfies } p_{T_{i-1}}(n) \tag{A.228}$$

Since B has l s.t. (A.227), W_{i-1} cannot satisfy B. Therefore by condition 1 of definition (20),

$$W_{i-1} \text{ falsifies } B \tag{A.229}$$

Similarly to 2, given (A.229), we can show that every possible world W_j satisfying $p_{T_{i-1}}(n)$ does not satisfy B, which implies $p_{T_{i-1}}(n)$ is Π_j-incompatible with B

4. We prove condition 4. By lemma 18, there is a unique possible world W_{i-1} of Π_{i-1} such that:

$$W_{i-1} \text{ satisfies } p_{T_{i-1}}(n) \tag{A.230}$$

As in 1, we can show that r is active in W_{i-1}. Therefore, by condition 2 of 19, we have that every atom $p(y)$ s.t. $y \in range(a_i)$ belongs to L_{i-1}. Therefore, condition 4 immediately follows from the fact n is ready to branch on a_i via rule r relative to Π_i and lemmas (20), (21).

$\leftarrow$

Now suppose n is ready to branch on a_i via rule r

$$random(a_i : \{X : p(X)\}) \leftarrow K \tag{A.231}$$

relative to Π_j. We show that n is ready to branch on a_i via r relative to Π_i. We prove the conditions 1-4 of the definition:

1. Condition 1 follows immediately from the fact that n is ready to branch on a_i relative to Π_i.

2. We prove Condition 2:

$$p_{T_{i-1}}(n)\ \Pi_i\text{-guarantees } K \tag{A.232}$$

By lemma 18, there is a unique possible world W_{i-1} of Π_{i-1} such that

$$W_{i-1} \text{ satisfies } p_{T_{i-1}}(n) \tag{A.233}$$

We prove that

$$r \text{ is active in } W_{i-1} \tag{A.234}$$

We prove by contradiction. Suppose r is not active in W_{i-1}. By condition 1 of 20 we have W_{i-1} falsifies K. That is, there is a literal $l \in L_{i-1}$ such that W_{i-1} does not satisfy l. Then, by conditions 2-3 of 20 and Lemma 17 we have that Π_j has a possible world W_j containing W_{i-1}. By Lemma 13, $W_j \setminus W_{i-1} \cap L_{i-1} = \emptyset$. Therefore, W_j does not satisfy l, and, therefore, K. This, given that $p_{T_{i-1}}(n) \setminus \{true\} \subseteq W_{i-1} \subseteq W_j$ contradicts the fact n is ready to branch on a_i via r relative to Π_j. Therefore, (A.234) holds. Therefore, the literals occurring in the body of r are from L_{i-1}, and by lemma (21) we conclude (A.232).

3. We prove condition 3. Let $pr(a_i = y \mid B) = v$ be a pr-atom from Π_i. We show that

$$p_{T_{i-1}}(n) \text{ either } \Pi_i\text{-guarantees } B \text{ or is } \Pi_i\text{-incompatible with } B \tag{A.235}$$

Since n is ready to branch on a_i via rule r relative to Π_j, we have 2 cases:

(a) B is Π_j-guaranteed by $p_{T_{i-1}}(n)$. Using the arguments similar to the ones from 2, we can obtain $B \in L_{i-1}$, and conclude by Lemma 21 that $p_{T_{i-1}}(n)$ Π_j-guarantees B

(b) B is Π_j-incompatible with $p_{T_{i-1}}(n)$. That is,

$$\text{every possible world } W_j \text{ or } \Pi_j \text{ satisfying } p_{T_{i-1}}(n) \text{ does not satisfy } B \tag{A.236}$$

By lemma 18, there is a unique possible world W_{i-1} of Π_{i-1} such that

$$W_{i-1} \text{ satisfies } p_{T_{i-1}}(n) \tag{A.237}$$

We prove

$$W_{i-1} \text{ falsifies } B \tag{A.238}$$

For the sake of contradiction, suppose (A.222) is false. By condition 1 of definition (20) we have:

$$W_{i-1} \text{ satisfies } B \tag{A.239}$$

By 20 and lemma (17) we have that Π_j has a possible world W_j containing W_{i-1}. Since B is in L_i, by lemma (13) we have

$$W_j \text{ satisfies } B \tag{A.240}$$

Since $p_{T_{i-1}}(n) \setminus \{true\} \subseteq W_{i-1} \subseteq W_i$, we have a contradiction from (A.240) and (A.236).

Therefore, (A.238) holds. Now let W_i be a possible world of Π_i satisfying $p_{T_{i-1}}(n)$. By lemma (13), There is a possible W'_{i-1} of Π_{i-1} such that $W'_{i-1} \subseteq W_j$ and

$$(W'_{i-1} \cap W_i) \cap L_{i-1} = \emptyset \tag{A.241}$$

Since $p_{T_{i-1}}(n)$ is in L_{i-1}, we have $p_{T_{i-1}}(n) \subseteq W'_{i-1}$. Therefore, By lemma

18

$$W'_{i-1} = W_{i-1} \tag{A.242}$$

From (A.242), (A.238), (A.2251) we have that W_i does not satisfy B. Therefore, $p_{T_{i-1}}(n)$ is Π_i-incompatible with B

4. As in 1, we can show that r is active in W_{i-1}. Therefore, by condition 2 of 19, we have that every atom $p(y)$ s.t. $y \in range(a_i)$ belongs to L_{i-1}. Therefore, condition 4 immediately follows from the fact n is ready to branch on a_i via rule r relative to Π_j and lemmas (20), (21).

□

Lemma 23. $T = T_k$ is a tableau for $\Pi \setminus U = \Pi_k$.

□

Proof. Follows immediately from the construction of the T's and Π's, the definition of a tableau, and Lemmas 22 and 20. □

□

Lemma 24. $T = T_k$ represents $\Pi \setminus U = \Pi_k$.

□

Proof. Let W be a possible world of Π. By Lemma 19 W contains $p_T(n)$ for some unique leaf node n of T. By Lemma 18, W is the set of literals Π-guaranteed by $p_T(n)$, and hence W is represented by n. Suppose now that n' is a node of T representing W. Then $p_T(n')$ Π-guarantees W which implies that W contains $p_{T_m}(n')$. By Lemma 19 this means that $n = n'$, and hence we proved that every answer set of Π is represented by exactly one leaf node of T.

Now let n be a leaf node of T. By Lemma 18 Π has a unique possible world W containing $p_T(n)$. It is easy to see that W is the set of literals represented by n. □ □

Lemma 25. Suppose T is a tableau representing Π. If n is a node of T which is ready to branch on $a(\bar{t})$ via r, then all possible worlds of Π compatible with $p_T(n)$ are probabilistically equivalent with respect to r.

□

Proof. This is immediate from Conditions (3) and (4) of the definition of ready-to-branch.

□

Notation: If n is a node of T which is ready to branch on $a(\bar{t})$ via r, the Lemma 25 guarantees that there is a unique scenario for r containing all possible worlds compatible with $p_T(n)$. We will refer to this scenario as *the scenario determined by* n.

Lemma 26. $T = T_m$ is unitary

□

Proof. We need to show that for every node n of T, the sum of the labels of the arcs leaving n is 1. Let n be a node and let s be the scenario determined by n. s satisfies (1) or (2) of the Definition 25. In case (1) is satisfied, the definition of $v(n, a(\bar{t}), y)$, along with the construction of the labels of arcs of T, guarantee that the sum of the labels of the arcs leaving n is 1. In case (2) is satisfied, the conclusion follows from the same considerations, along with the definition of $PD(W, a(\bar{t}) = y)$. □

Lemma 27. $T = T_m$ is a probabilistically sound representation of $\Pi \setminus U$.

Proof. Let R be a mapping from the possible worlds of $\Pi \setminus U$ to the leaf nodes of T which represent them. We need to show that for every possible world W of $\Pi \setminus U$ we have

$$v_T(R(W)) = \mu(W). \tag{A.243}$$

By definition of μ, we have:

$$\mu(W) = \frac{\hat{\mu}(W)}{\sum_{W_i \in \Omega(\Pi \setminus U)} \hat{\mu}(W_i)} \tag{A.244}$$

where

$$\hat{\mu}(W) = \prod_{W(a)=y} P(W, a = y) \tag{A.245}$$

where the product is taken over atoms for which $P(W, a = y)$ is defined.

By lemma 26, T is a unitary tree. Therefore, by lemma 6 we have that the sum of path values of it's leaves is 1. Therefore, it is sufficient to show that for every possible world W of $\Pi\backslash$

$$v_T(R(W)) = \hat{\mu}(W). \tag{A.246}$$

To prove A.246, it is sufficient to show that for every possible world W of $\Pi \setminus U$ (1) $p_T(R(W))$ contains an atom $a = y$ if and only if $a = y \in W$ and $P(W, a = y)$ is defined, (2) if n is a node in the path P of T from its root to $R(W)$ which branches on a, then the probability assigned to the arc which goes from n to its child in P, $v(n, a, y)$ is equal to $P(W, a = y)$.

1) $\Rightarrow$ We first show that if $p_T(R(W))$ contains an atom $a = y$, then $P(W, a = y)$ is defined and $W(a) = y$.

   By definition of $P(W, a = y)$, it is defined if and only if there exists a rule of Π of the form

$$random(a : \{X : p(X)\}) \leftarrow K \tag{A.247}$$

   such that W satisfies K, $truly\_random(a)$ and $p(y)$.

   By definition of T, if $a = y$ belongs to $p_T(R(W))$, there must exist a node n in the path from the root of T to $R(W)$ such that n branches on a via some rule r of the form (A.247) of Π. This means that $p_T(n)$ $\Pi \setminus U$-guarantees K and $p(y)$. By construction $p_T(R(W))$ contains $p_T(N)$, thus, $p_T(R(W))$ $\Pi \setminus U$-guarantees the body of r and $p(y)$. Since $R(W)$ represents W, W must contain all positive literals in $p_T(R(W))$. Therefore, W satisfies both K and $p(y)$. From rule A.247 it follows that W satisfies $random(a : \{X : p(X)\})$, and, since $\Pi \setminus U$ does not contain activity records, W satisfies $truly\_random(a)$.

By definition of a tableau representing a program, $p_T(R(W))$ $\Pi \setminus U$-guarantees W. By lemma 19 and minimality of possible worlds, W contains $p_T(R(W))$. This, $W(a) = y$.

$\Leftarrow$ We show that if $P(W, a = y)$ is defined, and $W(a) = y$, then $p_T(R(W))$ contains an atom $a = y$. We prove by contradiction. Suppose $p_T(R(W))$ does not contain an atom $a = y$. There are two possible cases:

(a) $p_T(R(W))$ contains an atom $a = y_1$, where $y_1 \neq y$. By definition of a tree representing a possible world, $p_T(R(W))$ $\Pi \setminus U$-guarantees W. By lemma 19 and minimality of possible worlds (proposition 13), we have that

$$W \text{ satisfies } p_T(R(W)) \tag{A.248}$$

Therefore, $W(a)$ contains both $a = y_1$ and $a = y$, which is impossible by definition of an interpretation.

(b) $p_T(R(W))$ contains no literal of the form $a = y_1$ for any y_1. In this case, using minimality of possible worlds, it is easy to see that $R(W)$ is ready to branch on a, which contradicts the definition of a tableau.

2) We show that if n is a node in the path P of T from its root to $R(W)$ which branches on a, then the probability assigned to the arc which goes from n to its child in P, $v(n, a, y)$, is equal to $P(W, a = y)$. By definition of $v(n, a, y)$, we only need to show that W is $\Pi \setminus U$-compatible with $p_T(n)$. Since W is a possible world of $\Pi \setminus U$, it is sufficient to show that W contains $p_T(n) \setminus \{true\}$. From A.248 we have that W contains $p_T(R(W)) \setminus \{true\}$. Since n is a node on the path P from the root of T to $R(W)$, $p_T(R(W))$ contains $p_T(n) \setminus \{true\}$.

Therefore, W contains $p_T(n) \setminus \{true\}$.

□

Therefore, as shown by Lemmas 24, 27, and 26, T is a unitary probabilistically sound representation of $\Pi \setminus U$, that concludes the proof of Lemma 8.

We are now ready to prove the main theorem.

Theorem 1

Every dynamically causally ordered, unitary program is coherent.

Proof. Suppose Π is dynamically causally ordered and U be the set of activity records of Π. Proposition 8 tells us that $\Pi \setminus U$ is represented by some tableau T. Lemmas 26 and 27 tells us that the tree is unitary and that the representation is probabilistically sound correspondingly. Thus, by Lemma 7 Π is coherent. □

A.3 Algorithm Correctness Proof

A.3.1 Proof of Proposition 3

Lemma 28. Let Π be a program with signature Σ. Suppose Π contains a rule $a = y \leftarrow B$ such that a is a random attribute and Π contains an action $do(a, y')$ belongs to Π. Let Π' be a program with signature of $\gamma(\Pi)$, obtained from Π by adding the rules:

$$f_{do}(a) \leftarrow B$$

$$\neg f_{do}(a) \leftarrow not\ f_{do}(a)$$

of $\gamma(\Pi)$. Let ψ be a mapping from the possible worlds of Π such that for each possible world W of Π:

$$\psi(W) = W \cup \{f_{do}(a) \mid B \text{ is satisfied by } W\} \cup \{\neg f_{do}(a) \mid B \text{ is not satisfied by } W\}.$$

ψ is a bijection from Ω_Π to $\Omega_{\Pi'}$.

□

Proof. In 1. we will prove that ψ is a function from Ω_Π to $\Omega_{\Pi'}$. In 2 we will show that ψ is surjective. In 3 we will prove that ψ is injective. 1-3 together imply that ψ is a bijection from Ω_Π to $\Omega_{\Pi'}$.

1. Let W be a possible world of Π. We will prove that $\psi(W)$ is a possible world of Π'. Let L be the set of literals from Σ. L is a splitting set of Π'. $bot_L(\Pi')$ is Π, thus W is a possible world of $bot_L(\Pi')$. $Y = \{f_{do}(a) \mid B \text{ is satisfied by } W\} \cup \{\neg f_{do}(a) \mid B \text{ is not satisfied by } W\}$ is a possible world of $e_L(\Pi', W)$. Therefore, $\psi(W) = W \cup Y$ is a possible world of Π'.

2. Let W' be a possible world of Π'. Let L be the set of literals from Σ. L is a splitting set of Π'. By splitting set theorem, $W' = W \cup Y$, where W is a possible world of Π. It is easy to see that $\psi(W) = W'$. Thus, ψ is surjective.

3. Let W_1 and W_2 be two distinct possible worlds of Π. By definition of ψ, $\psi(W) \setminus W$ does not include atoms from Σ. Therefore, since W_1 and W_2 are different, $\psi(W_1)$ and $\psi(W_2)$ differ on at least one atom from Σ.

□

Lemma 29. Let Π be a program not containing activity records with signature Σ. Suppose Π contains a rule $a = y \leftarrow B$ such that a is a random attribute and Π. Let Π_2 be a program with signature of $\gamma(\Pi)$, obtained from Π by replacing $a = y \leftarrow B$ with:

$$random(a, p_r) \leftarrow B$$

$$p_r(y)$$

of $\gamma(\Pi)$. Let ψ be a mapping from the possible worlds of Π such that for each possible world W of Π

$$\psi(W) = W \cup \{random(a, p_r) \mid B \text{ is satisfied by } W\}$$
$$\cup \{p_r(y)\}$$
$$\cup \{truly\_random(a) \mid B \text{ is satisfied by } W\}$$

ψ is a bijection from Ω_Π to Ω_{Π_2}.

□

Proof. In 1. we will prove that ψ is a function from Ω_Π to $\Omega_{\Pi'}$. In 2 we will show that ψ is surjective. In 3 we will prove that ψ is injective. 1-3 together imply that ψ is a bijection from Ω_Π to $\Omega_{\Pi'}$.

1. Let W be a possible world of Π we will show that $\psi(W)$ is a possible world of Π_2. In 1.1 we will show that $\psi(W)$ satisfies the rules of $\Pi_2^{\psi(W)}$ In 1.2 we will prove that $\psi(W)$ is minimal.

   1.1 Since no e-literals with default negation in the bodies of rules of Π_2 are formed by $random(a, p_r)$, $p_r(y)$, or $truly\_random(a)$, We have:

$$\Pi_2^{\psi(W)} = (\Pi \setminus \{a = y \leftarrow B\})^W$$
$$\cup \{random(a, p_r) \leftarrow B\}^W$$
$$\cup \{p_r(y)\}$$
$$\cup \{truly\_random(a) \leftarrow random(a, p_r)\}$$
$$\cup \{a = y_1 \leftarrow random(a, p_r), not\ a = y_2, \dots, not\ a = y_k\}$$
$$\dots$$
$$\cup \{a = y_k \leftarrow random(a, p_r), not\ a = y_2, \dots, not\ a = y_{k-1}\}$$
$$\cup \{\leftarrow a = Y, random(a, p_r), not\ p_r(Y)\} \quad \text{(A.249)}$$

where $range(a) = \{y_1, \ldots, y_k\}$.

Since $random(a, p_r)$ is in $\psi(W)$ if W satisfies B:

$$\{random(a, p_r) \leftarrow B\}^W \text{ is satisfies by } \psi(W) \tag{A.250}$$

By construction of $\psi(W)$:

$$p_r(y) \in \psi(W) \tag{A.251}$$

Since the bodies of rules of Π_2, except possibly the rule

$$truly\_random(a) \leftarrow random(a, p_r)$$

satisfied by $\psi(W)$, do not contain e-literals formed by $random(a, p_r)$, $p_r(y)$ and $truly\_random(a)$, and W satisfies Π^W, we have:

$$\psi(W) \text{ satisfies } (\Pi \setminus \{a = y \leftarrow B\})^W \tag{A.252}$$

If $random(a, p_r) \notin \psi(W)$, then the other rules are satisfied. Otherwise, B is satisfied by W, and, therefore, $a = y \in W$, which also implies that the other rules are satisfied. Therefore, from (A.249), (A.250), (A.251) we have that $\psi(W)$ satisfies $\Pi_2^{\psi(W)}$.

1.2 We show $\psi(W)$ is minimal. For the sake of contradiction, suppose there is W' such that:

$$W' \subsetneq \psi(W) \tag{A.253}$$

and W' satisfies $\Pi_2^{\psi(W)}$. Consider the set W'' defined as follows:

$$\begin{aligned} W'' = W' &\setminus \{p_r(y)\} \\ &\setminus \{random(a, p_r) \mid B \text{ is satisfied by } W\} \\ &\setminus \{truly\_random(a) \mid B \text{ is satisfied by } W, \\ &\qquad truly\_random(a) \notin W\} \end{aligned} \tag{A.254}$$

In 1.2.1 we will show

$$W'' \subsetneq W \tag{A.255}$$

In 1.2.2 we will prove W'' satisfies the rules of Π^W, obtaining a contradiction to the fact that W is a possible world of Π^W.

1.2.1 By construction, $W'' \subseteq W$. For the sake of contradiction, suppose

$$W'' = W \tag{A.256}$$

There are two possibilities:

(a) B is satisfied by W. In this case W' and $\psi(W)$ coincide on the atoms different from $p_r(y)$, $random(a, p_r)$ and $truly\_random(a)$. Therefore, since W' satisfies the rules of $\Pi_2^{\psi(W)}$, it must contain $p_r(y)$, $random(a, p_r)$ and $truly\_random(a)$. Therefore, we have $W' = \psi(W)$, which is a contradiction to (A.253).

(b) B is not satisfied by W. In this case $\psi(W) = W \cup \{p_r(y)\}$. Since W' satisfies the rules of $\Pi_2^{\psi(W)}$, it contains $p_r(y)$. Therefore, from (A.253) there exists an atom a different from $p_r(y)$ such that:

$$a \notin W' \tag{A.257}$$

and

$$a \in \psi(W) \tag{A.258}$$

By construction of W'':

$$a \notin W'' \tag{A.259}$$

By construction of ψ:

$$a \in W \tag{A.260}$$

From (A.260) and (A.259) we have:

$$W \neq W'' \tag{A.261}$$

which clearly contradicts (A.256).

1.2.2 Since W' satisfies $\Pi_2^{\psi(W)}$, by (A.249) we have W' satisfies all the rules of Π^W except possibly $\{a = y \leftarrow B\}^W$. By construction of W'', W'' also satisfies all those rules. We now prove W'' satisfies $\{a = y \leftarrow B\}^W$. If $\{a = y \leftarrow B\}^W$ is empty, the case is trivially true. Therefore, we can consider:

$$\{a = y \leftarrow B\}^W \text{ is } \{a = y \leftarrow B'\} \text{ for some set of literals } B' \tag{A.262}$$

If B' is not satisfied by W'', then the rule $a = y \leftarrow B'$ is satisfied by W''.

Suppose now that

$$B' \text{ is satisfied by } W'' \tag{A.263}$$

Since B' does not contain literals formed by any of the attributes from

$\{p_r, random, truly\_random\}$, from (A.254) we have:

$$W' \text{ satisfies } B \tag{A.264}$$

Since W' satisfies $\Pi_2^{\psi(W)}$, from (A.249) we have:

$$\{random(a, p_r) \leftarrow B\}^W \text{ is satisfied by } W' \tag{A.265}$$

From (A.262) and (A.265) we have:

$$random(a, p_r) \leftarrow B' \text{ is satisfied by } W' \tag{A.266}$$

From (A.266) and (A.263):

$$random(a, p_r) \in W' \tag{A.267}$$

Therefore, since W' satisfies the rules $\Pi_2^{\psi(W)}$, we must have $a = y \in W'$. Therefore,since a is a random attribute term, W'' satisfies $a = y$ and the rule $a = y \leftarrow B'$.

2. We prove that ψ is surjective. Let W be a possible world of Π_2. We can show that the set of atoms:

$$\begin{aligned} W' = W &\setminus \{p_r(y)\} \\ &\setminus \{random(a, p_r)\} \\ &\setminus \{truly\_random(a) \mid random(a, p_r) \\ &\quad \text{is the only atom of the form } random(a, p') \text{ in } W\} \end{aligned} \tag{A.268}$$

is a possible world of of Π.

3. We prove that ψ is injective. Consider two distinct possible worlds W_1 and W_2 of

Π. Since $random(a, p_r)$ and $p_r(y)$ are not in Σ, we have that if $\psi(W_1) \neq \psi(W_2)$, then $W_1 = W_2 \cup \{truly\_random(a)\}$ (up to symmetry), which is impossible by minimality of possible worlds.

□

Lemma 30. Let Π be a program with signature Σ from $\mathcal{B}$. Let AR be the set of activity records in $\gamma(\Pi)$. $red(\gamma(\Pi) \setminus AR)$ is defined.

Proof. $red(\gamma(\Pi) \setminus AR)$ is defined iff $red(\gamma(\Pi) \setminus AR)_{base}$ has a unique possible world. By construction, $red(\gamma(\Pi) \setminus AR)_{base}$ consists of $\Pi_{base} \cup R$, where R a collection of the pairs rules of the forms:

$$f_{do}(a) \leftarrow B$$

$$\neg f_{do}(a) \leftarrow not\ f_{do}(a)$$

We have that $lit(\Sigma)$ is a splitting set of $(\gamma(\Pi) \setminus AR)_{base}$, with $bot_{lit(\Sigma)}((\gamma(\Pi) \setminus AR)_{base}) = \Pi_{base}$, having a unique possible world W and $e_{lit(\Sigma)}(\Pi, W)$ consists of the pairs of rules of the form:

$$f_{do}(a)$$

$$\neg f_{do}(a) \leftarrow not\ f_{do}(a)$$

Clearly, $e_{lit(\Sigma)}(\Pi, W)$ V has a unique possible world, and, therefore, by splitting set theorem $W \cup V$ is the unique possible world of $(\gamma(\Pi) \setminus AR)$. □

Lemma 31. Let Π be a program, U be the set of activity records of Π, and Π' be obtained from Π by removing U. Let W' be a possible world of Π' such that:

1. for every action $do(a = y)$ of Π, W satisfies $a = y$ and $random(a, p)$ for some p.

2. for every observation $obs(l)$ of Π, W satisfies l

Let W be the set of atoms defined as follows:

$$W = W' \setminus \{truly\_random(a) \mid do(a, y) \in \Pi \text{ for some } y\} \cup U \quad \text{(A.269)}$$

W is a possible world of Π.

□

Proof. In 1 we show W satisfies the rules of Π^W and in 2 we show W is a minimal such set.

1. Since $truly\_random$ does not occur in the bodies of rules, we have

$$\Pi^W = (\Pi'^{W'} \setminus R) \cup U$$

Where R is the collection of the rules of the form

$$truly\_random(a) \leftarrow random(a)$$

such that $do(a, y) \in \Pi$ for some y.

Since U is in W by construction, we only need to show that

$$W \text{ satisfies the rules from } \Pi'^{W'} \setminus R. \quad \text{(A.270)}$$

Clearly, from (A.269), W satisfies all rules of $\Pi'^{W'} \setminus R$ that do not contain occurrences of *do*, *obs* and *truly_random*. The possible forms of the remaining rules are considered below:

(a) $\leftarrow obs(l)$ such that $l \notin W'$. These rules are satisfied by W by condition 2 from the lemma, and the construction of W.

(b) $truly\_random(a) \leftarrow random(a, p)$ such that $do(a, y) \notin \Pi$ for every y. Suppose $random(a, p)$ belongs to W. We need to show $truly\_random(a)$

belongs to W. We have $random(a, p) \in W'$, and hence, since Π' does not contain actions or observations, and W' is a possible world of Π', $truly\_random(a) \in W'$. Since $do(a, y) \notin \Pi$ for every y, $truly\_random(a)$ belongs to W by construction.

(c) $\leftarrow do(a, y)$ such that $a = y \notin W'$ These rules are satisfied by W by condition 1 from the lemma, and the construction of W.

(d) $\leftarrow do(a, y)$, where there is no p such that $random(a, p)$ belongs to W. These rules are satisfied by condition 1 from the lemma, and the construction of W.

Therefore, (A.270) holds and W satisfies the rules of Π^W

2. We prove that W is minimal. Suppose there exists $V \subsetneq W$ such that V satisfies the rules of Π^W. Consider the set

$$V' = (V \setminus U) \cup \{truly\_random(a) \mid random(a, p) \in W' \text{ for some } p\} \quad \text{(A.271)}$$

On the other hand, from (A.269) we have:

$$W' = (W \setminus U) \cup \{truly\_random(a) \mid random(a, p) \in W' \text{ for some } p\} \quad \text{(A.272)}$$

We show

$$V' \subsetneq W' \quad \text{(A.273)}$$

Since $U \subseteq \Pi^W$, and both W and V satisfy the rules of $\Pi^W \setminus R$, we have $U \subseteq V$ and $U \subseteq W$. Therefore,

$$(V \setminus U) \subsetneq (W \setminus U) \quad \text{(A.274)}$$

We need to show that there exists atom l such that (A.275) - (A.278) below hold.

$$l \in W \tag{A.275}$$

$$l \notin V \tag{A.276}$$

$$l \notin U \tag{A.277}$$

$$l \notin \{truly\_random(a) \mid random(a, p) \in W' \text{ for some } p\} \tag{A.278}$$

From (A.274) we have that there exits an l' such that $l = l'$ satisfies (A.275) - (A.277). If l' also satisfies (A.278) – we found l. Suppose

$$l' = truly\_random(a) \tag{A.279}$$

and for some p

$$random(a, p) \in W' \tag{A.280}$$

Since $l' \in W$, from (A.269), Π does not contain actions of the form $do(a, y)$. Therefore, the rule

$$truly\_random(a) \leftarrow random(a, p)$$

belongs to Π^W. Since V satisfies the rule of Π^W, and (A.276) holds for $l = l'$, we have: $random(a, p) \notin V$. From (A.280) we have $random(a, p) \in W$. Thus, $l = random(a, p)$ satisfies conditions (A.275) - (A.278) and (A.273) holds.

We will prove that

$$V' \text{ satisfies the rules of } \Pi'^{W'} \tag{A.281}$$

We have:

$$\Pi'^{W'} = \Pi^W \setminus U \cup R \tag{A.282}$$

where R consists of rules of the form

$$truly\_random(a) \leftarrow random(a, p) \tag{A.283}$$

such that $do(a = y) \in \Pi$ for some y.

Since V satisfies the rules of Π^W, V' satisfies every rule of $\Pi^W \setminus U$ which do not contain literals formed by obs, do and $truly\_random$. We now show that V' satisfies the remaining rules of $\Pi^W \setminus U$. We consider their possible forms:

(a) $\leftarrow obs(l)$. Since W' is a possible world of Π, it does not contain atoms formed by obs. Thus, by (A.273) we have $obs(l) \notin V'$.

(b) $truly\_random(a) \leftarrow random(a, p)$ such that $do(a, y) \notin \Pi$ for every y. Suppose

$$random(a, p) \in V' \tag{A.284}$$

We need to show

$$truly\_random(a) \in V' \tag{A.285}$$

From (A.284), (A.271), (A.272) we have:

$$random(a, p) \in W' \tag{A.286}$$

Therefore, by (A.271) we have (A.285).

(c) $\leftarrow do(a, y)$. Since W' is a possible world of Π, it does not contain atoms formed by do. Thus, by (A.273) we have $do(a, y) \notin V'$.

Therefore, V' satisfies the rules of $\Pi^W \setminus U$. We now show that V' satisfies the rules of R. Suppose r is a rule of the form (A.283) such that $do(a = y) \in \Pi$. If V' satisfies $random(a, p)$, then, by construction $random(a, p) \in W'$, and by (A.271), $truly\_random(a) \in V'$. Therefore, (A.281) holds, and from (A.273) we have a contradiction to the fact that W' is a possible world of Π'. Therefore, W is a minimal set satisfying the rules of Π^W.

□

Proposition 3. Let Π be a program from $\mathcal{B}$. We have:

1. $\gamma(\Pi)$ is from $\mathcal{B}$
2. there is a bijection ϕ from the possible world of Π to the possible worlds of $\gamma(\Pi)$ such that for every possible world W of Π:
   (a) $\mu_\Pi(W) = \mu_{\gamma(\Pi)}(\phi(W))$, and
   (b) W and $\phi(W)$ coincide on the atoms of Π.

Proof. We first prove:

$$\gamma(\Pi) \text{ is from } \mathcal{B} \tag{A.287}$$

Let AR_1 be the set of activity records in Π and AR_2 be the set of activity records in $\gamma(\Pi)$. Let $a_1, \ldots, a_n$ be a probabilistic leveling of $\Pi \setminus AR_1$ satisfying conditions 1-3 from Definition 20. Clearly, $a_1, \ldots, a_n$ is a probabilistic leveling of $\gamma(\Pi) \setminus AR_2$. Let $\Pi_0, \ldots, \Pi_n$ be the dynamic structure of $\Pi \setminus AR_1$, and $\Pi^2_0, \ldots, \Pi^2_n$ be the dynamic structure of $\gamma(\Pi) \setminus AR_2$. From lemmas 28 and 29 it follows that there is a bijection

ψ_i from the possible worlds of Π_i and the possible worlds of Π_i^2:

$$\begin{aligned}
\psi_i(W) = W \cup \{f_{do}(a) \mid & \text{ there exists a rule } a = y \leftarrow B \text{ s.t. } W \models B, a \\
& \text{ is a random attribute term, and } do(a, y') \in \Pi \text{ for some } y'\} \\
\cup \{\neg f_{do}(a) \mid & \text{ there is a rule } a = y \leftarrow B \text{ s.t } a \\
& \text{ is a random attribute term } do(a, y) \in \Pi \text{ for some } y' \text{ and } W \not\models B\} \\
\cup \{random(a, p_r) \mid & \text{ there exists a rule } a = y \leftarrow B \text{ s.t. } W \models B \text{ and } a \\
& \text{ is a random attribute term}\} \\
\cup \{p_r(y) \mid & \text{ there exists a rule } a = y \leftarrow B \text{ s.t. } a \\
& \text{ is a random attribute term}\}
\end{aligned}$$

Since for every possible world W of Π_i, W and $\psi(W)$ do not differ on the atoms of Σ, conditions 1-3 of definition 21 are satisfied, so $\gamma(\Pi)$ is from $\mathcal{B}$.

Let us define ϕ as follows:

$$\begin{aligned}
\phi(W) = W \cup \{\neg f_{do}(a) \mid & \text{ there is a rule } a = y \leftarrow B \text{ s.t. } a \\
& \text{ is a random attribute term, and } do(a, y') \in \Pi \text{ for some } y' \in range(a)\} \\
\cup \{random(a, p_r) \mid & \text{ there exists a rule } a = y \leftarrow B \text{ s.t. } W \models B \text{ and } a \\
& \text{ is a random attribute term}\} \\
\cup \{p_r(y) \mid & \text{ there exists a rule } a = y \leftarrow B \text{ s.t. } a \\
& \text{ is a random attribute term}\} \\
\cup \{obs(\neg f_{do}(a))\} &
\end{aligned}$$

It follows from Lemma 31 and the definition of ψ_n that ϕ is a bijection from Ω_Π to $\Omega_{\gamma(\Pi)}$. Clearly, for every $W \in \Omega_\Pi$, $\phi(W)$ coincides with W on the atoms of Σ, $\mu_\Pi(W) = \mu_{\gamma(\Pi)}(\phi(W))$.

□

A.3.2 Proof of Proposition 4

Proposition 4 Let Π be a program with signature Σ. Let I be an e-interpretation of Σ and W be a possible world of Π compatible with I. We have:

- if I satisfies an e-literal l of Σ, then W satisfies l,
- if I falsifies a literal l of Σ, then W does not satisfy l

□

Proof. The first part of the claim: if I satisfies l, then W satisfies l follows immediately from the definition of a compatible possible world.

We next show that if I falsifies W, then W does not satisfy l. We next consider all 4 possible forms of l:

1. l is $a = y$. I contains a literal contrary to l, which can be of one of the forms:
   (a) $a \neq y$. Then, W satisfies $a \neq y$, $a = y_1 \in W$ for $y_1 \neq y$, therefore, since W cannot assign two values to a, W does not satisfy l.
   (b) $a = y_1$ for some $y_1 \neq y$. Then, W satisfies $a = y_1$, $a = y_1 \in W$. Therefore, since W cannot assign two values to a, W does not satisfy l.
   (c) *not* $a = y$. Since W satisfies *not* $a = y$, $a = y \notin W$.
2. l is $a \neq y$. I contains a literal contrary to l, which can be of one of the forms:
   (a) $a = y$. We have W satisfies $a = y$. Since W is an interpretation, for no $y_1 \neq y$ $a = y_1 \in W$. Therefore, W does not satisfy $a \neq y$.
   (b) *not* $a \neq y$. Since W satisfies *not* $a \neq y$, W does not satisfy $a \neq y$.
3. l is *not* $a = y$. Since I falsifies l, I contains $a = y$. By Definition 35, W satisfies $a = y$. Therefore, W does not satisfy l.

4. l is *not* $a \neq y$. . Since I falsifies l, I contains $a \neq y$. Therefore, W satisfies $a \neq y$, and W does not satisfy l.

□

A.3.3 Proof of Proposition 5

Before proving Proposition 5, we prove some auxiliary lemmas.

Lemma 32. Let Π be a program with signature Σ and I be an e-interpretation of Σ. which satisfies (falsifies) a set of extended literals B. Every possible world W of Π compatible with I satisfies (does not satisfy) B.

□

Proof. This follows immediately from Lemma 4 and Definitions 32 and 34.

□

Lemma 33. Let Π be a program with signature Σ. Let I be an e-interpretation of Σ. We have:

1. If random attribute term a of Σ is active in I via a rule r of Π, then every possible world of Π compatible with I assigns a value to Π from the set of possible values of a in I;

2. If random attribute term a if Σ is disabled in I, then every possible world of Π compatible with I does not assign a value to a.

□

Proof. 1. Let a be a random attribute term of Π which is active in I via a rule

$$random(a : \{X : p(X)\}) \leftarrow B \tag{A.288}$$

of Π.

Let Y be the set of possible values of a in I. Let W be a possible world of Π compatible with I. We will show

$$\exists y \in Y. a = y \in W \tag{A.289}$$

By clause 2.a) of Definition 10 we have:

$$B \text{ is satisfied by } I \tag{A.290}$$

By Lemma 32, from (A.290) and the fact that W is compatible with I we have

$$B \text{ is satisfied by } W \tag{A.291}$$

Since the rule (A.288) belongs to Π, from (A.291) by Proposition 1 we have:

$$random(a : \{X : p(X)\}) \text{ is satisfied by } W \tag{A.292}$$

From (A.292), the fact that axiom 2.6 belongs to Π by proposition 1 we have:

$$W \text{ assigns some value } y \text{ to } a \tag{A.293}$$

Since Π contains axiom (1.3), from (A.292) and (A.293) we have:

$$p(y) \in W \tag{A.294}$$

Therefore, we just need to show that $p(y) \in I$ (which would imply that y is a possible value of a in I). Since a is active via r, by clause 2 (c) of Definition 39 we have that I either falsifies or satisfies $p(y)$ from (A.294) and Proposition (4) we have $p(y) \in I$. Therefore, (A.289) holds.

2. Let a be a random attribute term of Π which is disabled in I. By definition of a disabled attribute term we have: that for every rule of the form

$$random(a : \{X : p(X)\}) \leftarrow B$$

B is falsified by I.

Let W be a possible world of Π compatible with I. We need to show:

$$\text{no atom of the form } a = y' \text{ belongs to } W \tag{A.295}$$

By lemma 32 from (a) and (b) we have for every rule of the form

$$random(a : \{X : p(X)\}) \leftarrow B$$

B is not satisfied by W.

From (c) by minimality of possible worlds we have:

$$W \text{ does not contain atoms of the form } random(a : \{X : p(X)\}) \tag{A.296}$$

Therefore, we have:

(e) the body of every axiom of the form:

$$a = y_1 \leftarrow random(a : \{X : p(X)\}), not\ a = y_2, \ldots, not\ a = y_k$$

is not satisfied by W

From (c) - (e) by minimality of possible worlds we have (A.295).

□

Lemma 34. Let $T_\Pi\langle f\rangle$ be an AI-tree of Π and C be a cut of $T_\Pi\langle f\rangle$. For every possible world of Π, there exists a unique leaf L of I such that W is compatible with L

□

Proof. Let W be a possible world of Π. We prove by (strong) induction on the number of i-nodes of C that, for any positive n and for any cut of size n or less, there exists a unique leaf I of C such that W is compatible with I.

Base Case $n = 1$ In this case C consists of a single node $f(\{\})$. Since W is compatible with $\{\}$, and f is a consequence of Π, we have that W is compatible with $f(\{\})$.

Ind. Hyp. Let $k > 1$ be an integer. Suppose for any cut C of size less than k, there exists a unique leaf I of C such that W is compatible with I.

Ind. Step We show that for any cut C of size k there exists a unique leaf I of C such that W is compatible with I. Let L be a leaf node of C such that the distance from L to the root of C is maximal. Since $k > 1$, L is not a root. Let A be the parent of L, and M be the parent of A. Since L is a leaf node such that the distance from L to the root of C is maximal, we have:

$$\text{no child of } A \text{ has children} \tag{A.297}$$

Let C' be the cut obtained from C by removing A and all its children. By inductive hypothesis:

$$\text{there exists a unique leaf } N \text{ of } C' \text{ s.t. } W \text{ is compatible with } N \tag{A.298}$$

We consider two possibilities:

1. $N \neq M$. Then, since N is unique, W is not compatible with M. Therefore,

since all the children are supersets of M, W is not compatible with any of them. Thus, N is the unique leaf node of C such that W is compatible with N.

2. $N = M$. In this case W is not compatible with any of the leafs of C which are not descendants of A. We will show that W is compatible with exactly one child of A. Since, by construction of an AI-tree, A is ready in M, there are two cases:

   (a) A is disabled in M. By Lemma 33, W does not assign a value to A,

   $$W \text{ is compatible with } satr(a = u) \tag{A.299}$$

   By construction of $T_\Pi\langle f\rangle$, A has exactly one child $f(M \cup \{a = u\})$. From (A.298), W is compatible with M. Therefore, from (A.299) we have that W is compatible with $M \cup \{a = u\}$. Since f is a consequence function of Π, W is compatible with $f(M \cup \{a = u\})$.

   (b) A is active in M. By Lemma 33,

   $$A = y \in W \tag{A.300}$$

   for some y from the set of possible values of A in N. By construction $T_\Pi\langle f\rangle$, each child of A assigns a distinct value to a. Clearly, W is not compatible with all the children of A which assign a value to A different from y. Therefore, it is sufficient to show that W is compatible with $f(M \cup \{a = y\}$. From (A.298), W is compatible with M. Therefore, from (A.300) we have that W is compatible with $M \cup \{a = y\}$. Since f is a consequence function of Π, W is compatible with $f(M \cup \{a = y\})$.

□

Proposition 5 Given a program Π from $\mathcal{B}$, a query Q of Π and a solution tree S of

Π with respect to Q, let $\mathcal{L}$ be the set of compatible leaves of S, and $\mathcal{L}_Q$ be the subset of $\mathcal{L}$ such that each member of $\mathcal{L}_Q$ satisfies Q. We have:

1. if P_Π is defined, then

$$P_\Pi(Q) = \frac{\sum_{I \in \mathcal{L}_Q} \hat{\mu}_\Pi(I)}{\sum_{I \in \mathcal{L}} \hat{\mu}_\Pi(I)} \tag{A.301}$$

2. otherwise,

$$\sum_{I \in \mathcal{L}} \hat{\mu}_\Pi(I) = 0 \tag{A.302}$$

□

Proof. Let $\Omega = \{W_1, \ldots, W_n\}$ be the set of all possible worlds of Π, and Ω^Q be the set of all possible worlds of Π in each of which Q is true. And for an e-interpretation I, let Ω^I be the subset of possible worlds of Π compatible with I.

Recall that the probability $P_\Pi(Q)$ is defined iff

$$\sum_{W \in \Omega} \hat{\mu}(W) \neq 0 \tag{A.303}$$

Let L be a set of literals each of which is *decided* (falsified or satisfied) in every member of $\mathcal{L}$. By $\mathcal{L}_L$ we denote the subset of $\mathcal{L}$ consisting of all elements of $\mathcal{L}$ satisfying L. Also, by Ω_L we denote the subset Ω satisfying every member of L.

We start from showing:

$$\sum_{I \in \mathcal{L}_L} \hat{\mu}(I) = \sum_{W \in \Omega_L} \hat{\mu}(W) \tag{A.304}$$

By Definition 37 we have:

$$\sum_{I \in \mathcal{L}_L} \hat{\mu}(I) = \sum_{I \in \mathcal{L}_L} \left(\sum_{W \in \Omega^I} (\hat{\mu}(W)) \right) \tag{A.305}$$

We next show that for two distinct $I_1, I_2 \in \mathcal{L}$:

$$\Omega^{I_1} \cap \Omega^{I_2} = \emptyset \tag{A.306}$$

Let N be the lowest common ancestor of I_1 and I_2 in S. By construction, N is an a-node, and there exists two different terms $N = y_1 \in I_1$ and $N = y_2 \in I_2$. By definition of a compatible possible world (Def. 35), every possible world compatible with both I_1 and I_2 has to satisfy both $N = y_1$ and $N = y_2$, which is impossible. Therefore, (A.306) holds.

From (A.305) and (A.306) we have:

$$\sum_{I \in \mathcal{L}_L} \hat{\mu}(I) = \sum_{W \in (\bigcup_{I \in \mathcal{W}_L} \Omega^I)} \hat{\mu}(W) \tag{A.307}$$

We next show:

$$\bigcup_{I \in \mathcal{L}_L} \Omega^I = \Omega_L \tag{A.308}$$

Indeed, suppose $W \in \bigcup_{I \in \mathcal{W}_L} \Omega^I$. That is, W is compatible with I which satisfies L. This, clearly, means that W satisfies L, and, therefore $W \in \Omega_L$.

On the other hand, suppose $W \in \Omega_L$.That is, W is a possible world which satisfies L. By Lemma 34, there exists a unique member $\mathcal{V} \in \mathcal{L}$ such that W is compatible with $\mathcal{V}$. That is,

$$W \in \Omega^{\mathcal{V}} \tag{A.309}$$

Since L is satisfied by W , L is decided by every member of $\mathcal{L}$, and W is compatible with L, by Proposition 4, $\mathcal{V}$ satisfies L. That is,

$$\mathcal{V} \in \mathcal{L}_L \tag{A.310}$$

From (A.309) and (A.310) we have $W \in \bigcup_{I \in \mathcal{L}_L} \Omega^I$. Therefore, (A.308) holds.

From (A.308) and (A.307) we have (A.304).

We now consider both cases from the proposition.

1. P_Π is defined. In this case, let us define P_Q and P_N as follows:

$$P_Q = \sum_{I \in \mathcal{W}_{\{Q\}}} (\hat{\mu}(I)) \tag{A.311}$$

and

$$P_N = \sum_{I \in \mathcal{W}_{\{\}}} (\hat{\mu}(I)) \tag{A.312}$$

From (A.311) and (A.304) for $L = \{Q\}$ we have:

$$P_Q = \sum_{W \in \Omega_{\{Q\}}} \hat{\mu}(W) \tag{A.313}$$

From (A.312) and (A.304) for $L = \{\}$ we have:

$$P_N = \sum_{W \in \Omega} \hat{\mu}(W) \tag{A.314}$$

From (A.311) - (A.314) we have:

$$P_Q/P_N = \frac{\sum\limits_{I \in \mathcal{W}_{\{Q\}}} \hat{\mu}(I)}{\sum\limits_{I \in \mathcal{W}_{\{\}}} \hat{\mu}(I)} = \frac{\sum\limits_{W \in \Omega_{\{Q\}}} \hat{\mu}(W)}{\sum\limits_{W \in \Omega} \hat{\mu}(W)} = P_\Pi(Q) \tag{A.315}$$

2. P_Π is undefined. That is, In this case we have:

$$\sum_{W \in \Omega} \hat{\mu}(W) = 0 \tag{A.316}$$

By (A.304) for $L = \{\}$ we have:

$$\sum_{I \in \mathcal{L}} \hat{\mu}(I) = \sum_{W \in \Omega} \hat{\mu}(W) \tag{A.317}$$

From (A.316) and (A.317) we have:

$$\sum_{I \in \mathcal{L}} \hat{\mu}(I) = 0 \tag{A.318}$$

□

A.3.4 Proof of Proposition 6

Before proving the proposition, we prove some auxiliary lemmas.

Lemma 35. Let I be an interpretation of signature Σ, and L be the set of e-literals of Σ, and $l \in satr(L)$. We have:

1. if l is of the form $a = y$, then $l \in L$
2. if l is of the form $a \neq y$, then $l \in L$, or $a = y_1 \in satr(L)$ for some $y_1 \neq y$.
3. if l is of the form $not\ a = y$, then $l \in L$, or $a \neq y \in satr(L)$ or $not\ a \neq y_1 \in satr(L)$ for some $y_1 \neq y$
4. if l is of the form $not\ a \neq y$, then $l \in L$, or $a = y \in satr(L)$, or $\{not\ a = y_1 \mid y_1 \in range(a) \setminus \{y\}\} \subseteq satr(L)$

□

Proof. To prove 1, suppose l is $a = y$ and $a = y \notin L$. In this case the set $satr(L) \setminus l$ is saturated and contains L, which is a contradiction. To prove 2-4, suppose $l \notin L$. If none of the other conditions hold, we can, again, check easily that the set $satr(L) \setminus l$ is saturated and contains L, which is a contradiction.

□

Lemma 36. Let I be an interpretation of signature Σ and L be a set of e-literals of Σ. If I satisfies L, then I satisfies every literal from $satr(L)$

□

Proof. Suppose

$$I \text{ satisfies } L \tag{A.319}$$

Let l be an arbitrary e-literal from $satr(L)$. We need to show

$$I \text{ satisfies } l \tag{A.320}$$

We will show that I satisfies l. We consider all four possible forms of an e-literal in $satr(L)$:

1. l is $a = y$. In this case by clause 1 of Lemma 35 $l \in L$, and we have that I satisfies l.
2. l is $a \neq y$. The only possibilities are:
   (a) $l \in L$. From (A.319) we have (A.320)
   (b) $l \notin L$. By clause 2 of Lemma 35 we must have $a = y_1 \in satr(L)$ for some $y_1 \neq y$. By 1 we have $a = y_1 \in I$. Therefore, $a \neq y$ is satisfied by I.
3. l is *not* $a = y$. The only possibilities are:
   (a) $l \in L$. From (A.319) we have (A.320)
   (b) $l \notin L$. By clause 3 of Lemma 35, we have only two possibilities:
      i. $a \neq y \in satr(L)$. In this case by 2 we have that I satisfies $a \neq y$. Therefore, $a = y_1 \in I$ for some $y_1 \neq y$. Therefore, $a = y \notin I$ (otherwise, I would be inconsistent). Therefore, I satisfies l.
      ii. *not* $a \neq y_1 \in satr(L)$ for some $y_1 \neq y$. By clause 4 of Lemma 35, either $a = y_1 \in satr(L)$, or $\{not\ a = y_2 \mid y_2 \in range(a) \setminus \{y_1\}\} \subseteq satr(L)$

In the first case by 1 we have $a = y_1 \in I$ and l is satisfied by I. In the second we prove by contradiction. Suppose I does not satisfy $nota = y$. This means $a = y \in I$. Since I satisfies L, we must have

$$a \neq y \notin L \tag{A.321}$$

and

$$not\ a \neq y_1 \notin L \text{ for every } y_1 \in range(a) \setminus \{y\} \tag{A.322}$$

$$a = y_1 \notin L \text{ for every } y_1 \in range(a) \setminus \{y\} \tag{A.323}$$

$$not\ a = y \notin L \tag{A.324}$$

Let L_a be the set of e-literals formed by a in L, and S_a be the set of e-literals formed by a in $satr(L)$. Clearly, S_a is saturated. From (A.321) - (A.324) we have that all the e-literals formed by a in L are subset of $satr(a = y)$. But then $S'_a = S_a \cap satr(a = y)$ is a subset of S_a (since S_a contains $not\ a = y$). By lemma (38), S'_a is saturated. But then the set $satr(L) \setminus S_a \cup (satr(a = y) \cap S_a))$ is smaller than $satr(L)$ is saturated, and contains L, which is a contradiction.

4. l is $not\ a \neq y$.

   Again, there are two possibilities: The only possibilities are:

   (a) $l \in L$. From (A.319) we have (A.320)

   (b) $l \notin L$. We must have $a = y \in satr(I)$ or $\{not\ a = y_1 \mid y_1 \in range(a) \setminus \{y\}\} \subseteq satr(L)$ (otherwise, $satr(I) \setminus l$ is saturated). We consider both cases separately.

      i. $a = y \in satr(L)$ by 1 we have $a = y \in I$. Therefore, since I is an interpretation, $a \neq y$ is not satisfied by I. Thus, l is satisfied by I.

ii. $\{not\ a = y_1 \mid y_1 \in range(a) \setminus \{y\}\} \subseteq satr(L)$. By 3 we have that I satisfies $\{not\ a = y_1 \mid y_1 \in range(a) \setminus \{y\}\}$. Therefore, $I \cap \{a = y_1 \mid y_1 \in range(a) \setminus \{y\}\} = \emptyset$. Therefore, I does not satisfy $a \neq y$, and I satisfies l.

□

Lemma 37. Let Π be a program, U be the set of activity records of Π, and Π' be obtained from Π by removing U. Let W be a possible world of Π. Let W' be the set of atoms

$$W' = W \setminus U \cup \{truly\_random(a) \mid \exists p, y : random(a, p) \in W \text{ and } do(a, y) \in \Pi\} \tag{A.325}$$

We have W' is a possible world of Π'.

□

Proof. Since the bodies of rules of Π' do not contain occurrences of literals with default negation formed by *obs* and *truly_random*, we have:

$$\Pi'^{W'} = \Pi^{W} \cup R \setminus U \tag{A.326}$$

Where R is the collection of the rules of the form

$$truly\_random(a) \leftarrow random(a, p)$$

such that

$$do(a, y) \in \Pi \text{ for some } y \tag{A.327}$$

In 1 we show that W' satisfies the rules of $\Pi'^{W'}$. In 2 we show that no proper subset of W' satisfies the rules of $\Pi'^{W'}$.

1. We show that W' satisfies the rules of $\Pi'^{W'}$. From (A.326), it is sufficient to show that W' satisfies the rules of $\Pi^W \cup R \setminus U$. Let r be a rule from $\Pi^W \cup R \setminus U$. Suppose the body of r is satisfied by W'. We need to show that

$$\text{the head of } r \text{ is satisfied by } W' \tag{A.328}$$

Since the bodies of rules of $\Pi \cup R$ do not contain e-literals formed by attribute $truly\_random$, from (A.325) we have:

$$\text{the body of } r \text{ is satisfied by } W \tag{A.329}$$

Since r belongs to $\Pi^W \cup R \setminus U$, r also belongs to $\Pi^W \cup R$.

We know have two cases:

(a) r is a rule from Π^W. Since W is a possible world of Π, from (A.329) we have that

$$\text{the head of } r \text{ is satisfied by } W \tag{A.330}$$

Since $\Pi^W \setminus U$ does not contain literals formed by *do* and *obs* in the heads of rules and r belongs to $\Pi^W \setminus U$, from (A.330) and (A.325) we have (A.328).

(b) r is $truly\_random(a) \leftarrow random(a, p)$ and r belongs to R. From (A.327), (A.329) and (A.325) we have $truly\_random(a) \in W'$. Therefore, (A.328) holds.

2. We show that no proper subset of W' satisfies the rules of $\Pi'^{W'}$. For the sake of contradiction, suppose there is V' such that:

$$V' \subsetneq W' \tag{A.331}$$

and

$$V' \text{ satisfies the rules of } \Pi'^{W'} \tag{A.332}$$

Consider the set of atoms V defined as follows:

$$V = V' \setminus \{truly\_random(a) \mid \exists p, y:\ random(a,p) \in W \text{ and } do(a,y) \in \Pi\} \cup U \tag{A.333}$$

In (a) we will show:

$$V \subsetneq W \tag{A.334}$$

In (b) we will show:

$$V \text{ satisfies the rules of } \Pi^W \tag{A.335}$$

thus obtaining a contradiction to the fact that W is a possible world of Π.

(a) We show that (A.334) holds. From (A.325) we have:

$$W = W' \cup U \setminus \{truly\_random(a) \mid \exists p, y:\ random(a,p) \in W, do(a,y) \in \Pi\} \tag{A.336}$$

Since W' is a possible world of Π', $U \cap W' = \emptyset$. Therefore, it is sufficient to show:

$$V' \setminus \{truly\_random(a) \mid \exists p, y:\ random(a,p) \in W \text{ and } do(a,y) \in \Pi\} \tag{A.337}$$

$$\subsetneq W' \setminus \{truly\_random(a) \mid \exists p, y:\ random(a,p) \in W \text{ and } do(a,y) \in \Pi\}$$

From (A.331) we have:

$$V' \setminus \{truly\_random(a) \mid \exists p, y:\ random(a,p) \in W \text{ and } do(a,y) \in \Pi\} \tag{A.338}$$

$$\subseteq W' \setminus \{truly\_random(a) \mid \exists p, y:\ random(a,p) \in W \text{ and } do(a,y) \in \Pi\}$$

We need to show that there exists atom l such that (A.339) - (A.341) below hold.

$$l \in W' \tag{A.339}$$

$$l \notin V' \tag{A.340}$$

$$l \notin \{truly\_random(a) \mid \exists p, y : \ random(a,p) \in W \text{ and } do(a,y) \in \Pi\} \tag{A.341}$$

From (A.331) we have that there exists an l' such that $l = l'$ satisfies (A.339) - (A.340). If l' also satisfies (A.341) – we found l. Suppose

$$l' = truly\_random(a) \tag{A.342}$$

and there exists p such that

$$random(a,p) \in W \tag{A.343}$$

and

$$do(a,y) \in \Pi \text{ for some } y \tag{A.344}$$

Since W' contains no atoms formed by do, the rule

$$truly\_random(a) \leftarrow random(a,p)$$

belongs to $\Pi'^{W'}$. Therefore, from (A.332) and the fact that $l = l'$ satisfies (A.340), we have:

$$random(a,p) \notin V' \tag{A.345}$$

From (A.343) and (A.325) we have $random(a,p) \in W'$. Hence, $l = random(a,p)$ satisfies conditions (A.339) - (A.341) and (A.337) holds.

(b) We show that (A.335) holds. From (A.333) we have:

$$V \text{ satisfies } U \tag{A.346}$$

From (A.332) and (A.326) we have:

$$V' \text{ satisfies } \Pi^W \cup R \setminus U \tag{A.347}$$

Therefore, since Π does not contain e-literals formed by *truly_random* in the bodies of rules, from (A.333) we have:

$$V \text{ satisfies every rule from } \Pi^W \cup R \setminus U \quad \text{whose head is not formed by } truly\_random \tag{A.348}$$

Now let r be a rule from $\Pi^W \cup R \setminus U$ of the form

$$truly\_random(a) \leftarrow random(a, p') \tag{A.349}$$

Suppose

$$random(a, p') \in V \tag{A.350}$$

From (A.333) we have:

$$random(a, p') \in V' \tag{A.351}$$

Therefore, from the fact that $r \in \Pi^W \cup R \setminus U$ and (A.347) we have:

$$truly\_random(a) \in V' \tag{A.352}$$

The only two possible cases are:

i. r belongs to Π^W, by definition of the reduct we have:

$$\text{there is no } y \text{ such that } do(a, y) \in W \tag{A.353}$$

Since W is a possible world of Π, we have:

$$\text{there is no } y \text{ such that } do(a, y) \in \Pi \tag{A.354}$$

Therefore, from (A.352) by (A.333) we have:

$$truly\_random(a) \in V \tag{A.355}$$

Therefore, V satisfies r

ii. r belongs to R. By construction of R we have

$$do(a, y) \in \Pi \text{ for some y} \tag{A.356}$$

From (A.325), (A.351) and (A.331) we have

$$random(a, p') \in W \tag{A.357}$$

From (A.333), (A.357) and (A.356) we have we have:

$$truly\_random(a) \in V \tag{A.358}$$

Therefore, V satisfies r.

Therefore,

$$V \text{ satisfies every rule from } \Pi^W \setminus U \text{ whose head is of the form } truly\_random(a) \tag{A.359}$$

From (A.348) and (A.359) we have:

$$V \text{ satisfies every rule from } \Pi^W \setminus U \tag{A.360}$$

From (A.360) and (A.346) we have (A.335).

□

Lemma 38. Let S_1 and S_2 be two saturated sets of e-literals of Σ. The set $S_1 \cap S_2$ is saturated.

□

Proof. Let Y be the set $S_1 \cap S_2$. In 1, we prove condition 1. In 2, we prove condition 5 from the definition. The conditions 2-4 are very similar to 1 in their structure. So, the proofs will be similar as well, and we will omit them here.

1. Let $a = y$ be a member of Y. We need to show that

$$\forall y_1 \in range(a) \setminus \{y\},\ a \neq y_1 \text{ belongs to } Y \tag{A.361}$$

Since $Y = S_1 \cap S_2$ and $a = y \in Y$, we have that

$$a = y \in S_1 \tag{A.362}$$

and

$$a = y \in S_2 \tag{A.363}$$

From (A.423), (A.362) and (A.363) we have:

$$\forall y_1 \in range(a) \setminus \{y\},\ a \neq y_1 \text{ belongs to } S_1 \tag{A.364}$$

$$\forall y_1 \in range(a) \setminus \{y\},\ a \neq y_1 \text{ belongs to } S_2 \tag{A.365}$$

From (A.364) and (A.365) and the fact that $Y = S_1 \cap S_2$ we have (A.361).

2. We prove condition 5 for Y. Suppose a is an attribute with $range(a) = \{y_1, \ldots, y_k\}$ and there is $y \in \{y_1, \ldots, y_k\}$ such that

$$\{not\ a = y' \mid y' \in (\{y_1, \ldots, y_k\}\ \{y\})\} \subseteq Y \tag{A.366}$$

We need to show

$$not\ a \neq y \in Y \tag{A.367}$$

Since $Y = S_1 \cap S_2$, from (A.366) we have:

$$\{not\ a = y' \mid y' \in (\{y_1, \ldots, y_k\}\ \{y\})\} \subseteq S_1 \tag{A.368}$$

and

$$\{not\ a = y' \mid y' \in (\{y_1, \ldots, y_k\}\ \{y\})\} \subseteq S_2 \tag{A.369}$$

From (A.423), (A.368) and (A.369) we have:

$$not\ a \neq y \in S_1 \tag{A.370}$$

$$not\ a \neq y \in S_2 \tag{A.371}$$

From (A.370) (A.371), since $Y = S_1 \cap S_2$ we have (A.367).

□

Lemma 39. Let A and B be two sets of e-literals of Σ such that $A \subseteq B$. We have $satr(A) \subseteq satr(B)$.

□

Proof. For the sake of contradiction, suppose $satr(A) \not\subseteq satr(B)$. Consider the set $X = satr(A) \cap satr(B)$. We have $X \subsetneq satr(A)$ and $A \subseteq X$. By lemma 38, $X = satr(A) \cap satr(B)$ is saturated. Therefore, we have a contradiction to the fact that $satr(A)$ is the smallest superset of A which is saturated. □

Lemma 40. Let A and B be two sets of e-literals. We have $A \cup satr(B) \subseteq satr(A \cup B)$.

□

Proof. Let l be a literal from $A \cup satr(B)$. If $l \in A \cup B$, the truth of the lemma follows immediately. Suppose $l \notin A \cup B$. Then we have $l \in satr(B)$. By Lemma 39 we have $l \in satr(A \cup B)$.

□

Lemma 41. Let Π be a program. Let TU be the set of e-literals: Let TU be the set of e-literals:

$$TU = \{not\ l \mid l \text{ is formed by } truly\_random(a) \text{ and } do(a = y) \in \Pi \text{ for some } y\}$$

Let W be a possible world of Π. W satisfies TU.

□

Proof. Let l be an e-literal from TU formed by $truly\_random(a)$. Since W is a possible world of Π, the only rule with $truly\_random(a)$ in the head is:

$$truly\_random(a) \leftarrow random(a), not\ do(a, y_1), \ldots, not\ do(a, y_k)$$

and we have $do(a, y) \in \Pi$ for some $y \in \{y_1, \ldots, y_k\}$, we have that Π^W has no rules with $truly\_random(a)$ in the head. Therefore, by minimality of possible worlds, the atom $truly\_random(a)$ does not belong to W. Therefore, W satisfies l. □

For an e-literal l, by $atf(l)$ we will denote the attribute term used to form l. For a set of e-literals I, by $atf(I)$ we will denote the set of attribute terms:

$$atf(I) = \{atf(l) \mid l \in I\}$$

Lemma 42. Let A and B be two sets of e-literals of Σ, such that $atf(A) \cap atf(B) = \emptyset$. We have:

$$satr(A) \cup satr(B) = satr(A \cup B) \tag{A.372}$$

□

Proof. From Lemma 40 we have $satr(A) \cup B \subseteq satr(A \cup B)$ and $satr(B) \cup A \subseteq satr(A \cup B)$. Therefore $satr(A) \cup satr(B) \subseteq satr(A \cup B)$. We next show

$$satr(A \cup B) \subseteq satr(A) \cup satr(B) \tag{A.373}$$

For the sake of contradiction, suppose there exits an e-literal such that:

$$l \in satr(A \cup B) \tag{A.374}$$

but

$$l \notin satr(A) \cup satr(B) \tag{A.375}$$

Let $a = atf(l)$. We must have that $a \in atf(A)$ or $a \in atf(B)$ (otherwise, the set obtained from $satr(A \cup B)$ by removing all e-literals formed by a is saturated, is smaller than $satr(A \cup B)$, and contains both A and B). Without loss of generality,

$$a \in atf(A) \tag{A.376}$$

Let L_A be the set of e-literals in $satr(A)$ formed by a. Let L_{AB} be the set of e-literals in $satr(A \cup B)$ formed by a.

From (A.374) and (A.375) and the fact that l is formed by a we have that $L' = L_A \cap L_{AB} \subsetneq L_{AB}$. By lemma (38), L' is saturated. But then we have $satr(A \cup B) \setminus L_{AB} \cup L'$ is saturated and contains A and B, which is a contradiction.

□

Lemma 43. Let A be a set of atoms of Σ. $satr(A)$ is consistent.

Proof. It is easy to see that no interpretation can satisfy two inconsistent e-literals. Consider the interpretation I consisting of atoms in A. Clearly, I satisfies A. By lemma 36, I satisfies $satr(A)$, therefore it is consistent. □

Proposition 6. Let f be an admissible consequence function of program Π from $\mathcal{B}$. We have f is a consequence function of Π.

□

Proof. We prove both conditions of Definition (44) in 1 and 2 respectively. As in the definition (50), we define set of literals

$$L_{tr} = \{l \mid l \text{ is formed by } truly\_random(a) \text{ and } do(a = y) \in \Pi \text{ for some } y\}$$

set of e-literals:

$$TU = \{not\ l \mid l \in L_{lr}\}$$

AR, the set of activity records of Π and and AR_{NOT} be defined as in Definition 50.

1. We show that $f(\{\})$ is defined. Let f' be the consequence function of Π'. By condition 1 of definition (44) , we have that

$$f'(\{\}) \text{ is defined} \tag{A.377}$$

Since $I = \{\}$, we have:

$$I \cap L_{tr} = \{\} \tag{A.378}$$

and

$$I \cap AR_{NOT} = \{\} \tag{A.379}$$

$$I \text{ contains no literals formed by random attribute terms} \tag{A.380}$$

From (A.377) - (A.380) by definition (50) we have that $f(\{\})$ is defined.

2. We show that if $f(I)$ is defined, then $f(I)$ is a consequence of I w.r.t Π (Definition 36). In (a) we show that $f(I)$ is consistent (note that it is saturated by construction). In (b) and (c) we prove both conditions of Definition 36 respectively.

   (a) We show that $f(I)$ is consistent. Let X be the subset of set of e-literals from $f'(I \setminus (TU \cup satr(AR)))$ not containing attribute terms from TU and AR, and $Z = f'(I \setminus (TU \cup satr(AR))) \cap (TU \cup satr(AR))$.

We have:

$$f(I) = satr((f'(I \setminus (TU \cup satr(AR))) \setminus L_{tr} \setminus AR_{NOT}) \cup TU \cup AR) \tag{A.381}$$

$$= satr(X \cup Z \cup TU \cup AR)$$

$$\subseteq satr(X \cup TU \cup satr(AR)) \text{ (by Lemma (39))}$$

$$= satr(X) \cup satr(TU) \cup satr(AR) \text{ (by Lemma (42))}$$

$$= X \cup TU \cup satr(AR) (X \text{ and } TU \text{ are saturated by construction)}$$

It is easy to check see that $X \cup TU \cup satr(AR)$ is consistent: $atf(X)$, $atf(TU)$ and $atf(AR)$ are pairwise disjoint, X is a subset of an interpretation, TU contains only literals preceded by default negation, and $satr(AR)$ is consistent by lemma 43. Therefore, since, by (A.381), $f(I)$ is a subset of $X \cup TU \cup satr(AR)$, it is consistent.

(b) We show that $I \subseteq f(I)$. Since f' is a consequence function, we have

$$I \setminus (TU \cup satr(AR)) \subseteq f'(I \setminus (TU \cup satr(AR))) \tag{A.382}$$

Since $f(I)$ is defined, we have $I \cap L_{tr} = \emptyset$ and $I \cap AR_{NOT} = \emptyset$. Therefore,

$$I \setminus (TU \cup satr(AR)) \subseteq f'(I \setminus (TU \cup satr(AR))) \setminus L_{tr} \setminus AR_{NOT} \tag{A.383}$$

Therefore,

$$\begin{aligned}
I &= I \setminus (TU \cup satr(AR)) \cup TU \cup satr(AR) \\
&\subseteq f'((I \setminus (TU \cup satr(AR))) \setminus L_{tr} \setminus AR_{NOT}) \cup TU \cup satr(AR) \\
&\quad \text{(by (A.383))} \\
&\subseteq satr((f'(I \setminus (TU \cup satr(AR))) \setminus L_{tr} \setminus AR_{NOT}) \cup TU \cup AR) \\
&\quad \text{(by Lemma 40)} \\
&= f(I)
\end{aligned}$$

(c) We show that $f(I)$ is a consequence of I w.r.t Π. Let W be a possible world of Π. compatible with I. We need to show that W is compatible with $f(I)$. Let Π' be a program obtained from Π by removing AR. By Lemma (37), the set of atoms

$$W' = W \setminus U \cup \{truly\_random(a) \mid \exists p, y : \ random(a,p) \in W, do(a,y) \in \Pi\} \tag{A.384}$$

is a possible world of Π'.

We will show that

$$W' \text{ is compatible with } I \setminus (TU \cup satr(AR)) \tag{A.385}$$

By construction, W' satisfies every e-literal l of $I \setminus (TU \cup satr(AR))$ which is not formed by $do(a)$ such that $do(a) \in \Pi$, $obs(a)$ such that $obs(a) \in \Pi$ or one of the attribute terms from $\{truly\_random(a) \mid \exists p, y : \ random(a,p) \in W$ and $do(a,y) \in \Pi\}$. We consider the remaining forms of l in i-ii below.

i. for $f \in \{obs, do\}$, l is formed by $f(l)$ and $f(l) \in \Pi$. Since $l \in I$, and W is a possible world of Π compatible with I we have that W satisfies l. Therefore, l must belong to $satr(AR)$ (or else, $satr(AR)$ contains an e-literal contrary to l, and l is not satisfied by W).

ii. l is formed by a member of

$$\{truly\_random(a) \mid \exists p, y : \ random(a, p) \in W \text{ and } do(a, y) \in \Pi\}$$

Since l does not belong to TU, $l \in L_{tr}$. However, since $I \cap L_{tr} = \emptyset$ we cannot have $l \in I$.

Therefore, (A.385) holds. Since f' is a consequence function of Π', we have:

$$W' \text{ satisfies } f'(I \setminus (TU \cup satr(AR)) \tag{A.386}$$

By Lemma 41, we have:

$$W \text{ satisfies } TU \tag{A.387}$$

We next show

$$W \text{ satisfies } f'(I \setminus (TU \cup satr(AR)) \setminus L_{tr} \setminus AR_{NOT} \tag{A.388}$$

By construction, W satisfies every e-literal l of $f'(I \setminus (TU \cup satr(AR)) \setminus L_{tr} \setminus AR_{NOT}$ which is not formed by $do(a)$ such that $do(a) \in \Pi$, $obs(a)$ such that $obs(a) \in \Pi$ or one of the attribute terms from

$$\{truly\_random(a) \mid \exists p, y : \ random(a, p) \in W \text{ and } do(a, y) \in \Pi\}.$$

We consider the remaining forms of l in i-ii below.

i. for $f \in \{obs, do\}$, l is formed by $f(l)$, and $f(l) \in \Pi$. From (A.388) we have W' satisfies l. Since W' is an answer set of Π' that doesn't contain actions or observations, l must have default negation. Since $l \notin AR_{NOT}$, we only have two possibilities:

A. l is of the form $not\ f(l) = false$. Since W contains $f(l)$, it does not satisfy $f(l) = false$, and satisfies l.

B. l is of the form $not\ f(l) \neq true$. Since W contains $f(l)$, it does not satisfy $f(l) \neq true$, and satisfies l.

ii. l is formed by a member of

$$\{truly\_random(a) \mid \exists p, y:\ random(a,p) \in W \text{ and } do(a,y) \in \Pi\}.$$

Since $l \notin L_{tr}$, we must have $l \in TU$. By (A.387), we have W satisfies l.

Therefore, (A.388) holds.

Since W is a possible world of Π containing facts AR we have:

$$W \text{ satisfies } AR \tag{A.389}$$

From (A.389) , (A.388) and (A.387) we have:

$$(W \text{ satisfies } f'(I \setminus (TU \cup satr(AR)) \setminus L_{tr} \setminus AR_{NOT}) \cup AR \cup TU \tag{A.390}$$

From (A.390) by Lemma 36 we have that W is compatible with $f(I)$.

Finally, we show that no attribute term in $f(I) \setminus I$ is formed by a random attribute term of Π. Clearly, since AR, TU, L_{tr}, AR_{NOT} do not contain e-literals formed by random attribute terms, it is sufficient to show that no e-literal from $f'(I \setminus (TU \cup satr(AR)) \setminus I$ is formed by a random attribute term.

Since f' is a consequence function, no literal from $f'(I \setminus (TU \cup satr(AR)) \setminus (I \setminus (TU \cup satr(AR)))$ is formed by a random attribute term. Therefore, since AR, TU do not contain e-literals formed by random attribute terms, we have that no e-literal from $f'(I \setminus (TU \cup satr(AR)) \setminus I$ is formed by a random attribute term.

□

A.3.5 Proof of Proposition 7

We start from introducing some notation and definitions. Let Π be a program from $\mathcal{B}$ with signature Σ, f be a consequence function of Π, $T_\Pi\langle f\rangle$ be an AI-tree of Π, and I be an i-node of $T_\Pi\langle f\rangle$. By $Ratoms_\Pi(I)$ we will denote the set of atoms of Σ:

$$\{a = y \mid a = y \in I \text{ and } a \text{ is a random attribute term of } \Pi\}$$

Definition 75 (Reachable sequence)**.**
Let $I = I_0, \ldots, I_n$ be a non-empty sequence of e-interpretations of Σ. We will say that $I_0, \ldots, I_n$ is *reachable* w.r.t. Π iff:

1. I_0 is a consequence of $\{\}$ w.r.t Π.
2. For every $i \in \{1..n\}$, there exists a unique random attribute term a ready in I such that I_i is a consequence of $satr(I_{i-1} \cup a = y)$ for some possible value y of a in I_{i-1}. We will refer to a as $rdiff_I(I_i)$
3. For every $i \in \{1..n\}$, every e-literal in $I_i \setminus I_{i-1}$ formed by a random attribute term of Π is formed by $rdiff_I(I_i)$.

We will say that I is reachable *via a consequence function* f of Π if for every $i \in \{1..n\}$, $I_i = f(I_i \cup a = y)$ for a and y satisfying conditions from 2.

□

Definition 76 (Reachable e-interpretation)**.**
Let Π be a program with signature Σ. Let I be an e-interpretation of Σ. We will say that I is *reachable* w.r.t program Π if there exists a reachable sequence $I_1, \ldots, I_n$ such that $I_n = I$. We will refer to $I_0, \ldots, I_n$ as a *corresponding sequence* for I. For

a consequence function f of Π, we will say that I is *reachable via* f iff $I_0, \dots, I_n$ is reachable via f.

$\square$

We will often omit the program from the consideration if it is clear from the context. For a reachable sequence $I = I_0, I_1, \dots, I_n$ by $ats(I_0, I_1, \dots, I_n)$ we will denote the sequence $a_1 = y_1, \dots, a_n = y_n$ such that:

$$\text{for every } i \in \{1..n\}, a_i = diff_I(I_i) \text{ and } a_i = y_i \subseteq I_i$$

Lemma 44. Let A be a set of attribute terms of Σ, L_A be the set of all e-literals formed by A, and Y is a subset of L_A. Let X be a set of e-literals of Σ. Let I be an interpretation of Σ. If I satisfies X, then I satisfies $satr(X \cup Y) \setminus L_A$. $\square$

Proof. We first show

$$satr(X \cup Y) \setminus L_A \subseteq satr((X \cup Y) \setminus L_A) \tag{A.391}$$

For a set of e-literals L of Σ, and an attribute term a, by L^a we will denote the subset of L consisting of e-literals formed by a. Suppose (A.391) does not hold. In this case there exists an attribute term b such that

$$(satr(X \cup Y) \setminus L_A)^b \neq (satr((X \cup Y) \setminus L_A))^b$$

Clearly, $satr(X \cup Y) \setminus L_A$ is saturated. Therefore, by Lemma 38

$$L^b = (satr(X \cup Y) \setminus L_A)^b \cap (satr((X \cup Y) \setminus L_A))^b$$

is saturated. But then $satr((X \cup Y) \setminus L_A) \cap L^b$ is a proper subset of $satr((X \cup Y) \setminus L_A)$ containing $(X \cup Y) \setminus L_A$ which is saturated, which is a contradiction.

Therefore, (A.391) holds. By Lemma 39, since $(X \cup Y) \setminus L_A \subseteq X$ we have:

$$satr((X \cup Y) \setminus L_A) \subseteq satr(X) \tag{A.392}$$

From (A.391) and (A.392) we have:

$$satr(X \cup Y) \setminus L_A \subseteq satr(X) \tag{A.393}$$

Therefore, since I satisfies X, by Lemma 36 and the definition of satisfiability we have I satisfies $satr(X \cup Y) \setminus L_A$

□

Lemma 45. Let Π be a program from $\mathcal{B}$ with signature Σ. Let Π' be obtained from Π by removing all activity records. Let f be an admissible consequence function of Π. Let I an e-interpretation reachable w.r.t Π such that $J_0, J_1, \ldots, J_n$ is a corresponding sequence for I reachable via f. Let $a_1 = y_1, \ldots, a_n = y_n$ be $ats(J_0, \ldots, J_n)$. Let W' be a possible world of Π' such that $ats(J_0 \ldots, J_n) \cap Ratoms_{\Pi}(I) \subseteq W'$. Let I' be an interpretation obtained from I by removing:

1. e-literals formed by $truly\_random(a)$ for every a such that $do(a = y) \in \Pi$,
2. e-literals formed by $do(a)$ such that $do(a) \in \Pi$,
3. e-literals formed by $obs(l)$ such that $obs(l) \in \Pi$

W' is compatible with I'.

□

Proof. Let R be the set of attribute terms:

$$R = \{truly\_random(a) \mid do(a = y) \in \Pi \text{ for some } y\}$$

Let A_{tr} be the set of attribute terms:

$$A_{tr} = R \cup \{do(a) \mid do(a) \in \Pi\}$$
$$\cup \{obs(l) \mid obs(l) \in \Pi\}$$

and E_{tr} be the set of e-literals of Σ formed by attribute terms in A_{tr}, L_{tr}, TU, and AR_{NOT} be the sets of e-literals defined as in definition 50. Let $J'_0, \ldots, J'_n$ be a sequence of e-interpretations such that for every $i \in \{1..n\}$ $J'_i = J_i \setminus E_{tr}$. We will prove by induction on i that for every $i \in \{0..n\}$, W' is compatible with J'_i. The correctness of the lemma then follows immediately.

Let f' be the consequence function of Π' such that f is induced by f'.

Base Case We have $J_0 = satr((f'(\{\}) \setminus L_{tr} \setminus AR_{NOT}) \cup TU \cup AR)$

Since f' is a consequence of Π',

$$W' \text{ satisfies } f'(\{\}) \tag{A.394}$$

Therefore,

$$W' \text{ satisfies } f'(\{\}) \setminus L_{tr} \setminus AR_{NOT} \tag{A.395}$$

Since $TU \cup AR \subseteq E_{tr}$, by Lemma (44) we have that

$$W' \text{ satisfies } satr(f'(\{\}) \setminus L_{tr} \setminus AR_{NOT} \cup TU \cup AR) \setminus E_{tr} \tag{A.396}$$

That is, W' satisfies J'_0.

Ind. Hyp. Suppose

$$W' \text{ is compatible with } J'_k \tag{A.397}$$

for some $k < n$.

Ind. Step We will show that

$$W' \text{ is compatible with } J'_{k+1} \tag{A.398}$$

By inductive hypothesis, W' is compatible with J'_k, therefore, we have that

$$W' \text{ satisfies every e-literal from } J'_k \tag{A.399}$$

By definition of a reachable sequence, we have:

$$J_{k+1} = f(satr(J_k \cup \{a_{k+1} = y\})) \tag{A.400}$$

where

$$a_{k+1} \text{ is ready in } J_k \tag{A.401}$$

and

$$y \text{ is a possible value of } a_{k+1} \text{ in } J_k$$

We show

$$W' \text{ satisfies } a_{k+1} = y \tag{A.402}$$

We consider two cases: $a_{k+1} = u$ and $a_{k+1} \neq u$.

1. $a_{k+1} = u$.

   Since a_{k+1} is ready in J_k, and u is a possible value of a_{k+1} in J_k, we have that a_{k+1} is disabled in J_k. Since the bodies of random selection rules of Π do not contain literals formed by *truly_random*, *obs* and *do*, a_{k+1} is disabled in J'_k. Therefore, by Lemma 33, W' does not assign a value to a_{k+1}. and we have (A.402).

2. $a_{k+1} = y$ for $y \neq u$. Since $ats(J_0, \ldots, J_n) \cap Ratoms_{\Pi}(I) \subseteq W'$, and $a_{k+1} = y \subseteq J_{k+1} \subseteq J_n$, we have (A.402).

Then we have:

$$\begin{aligned} J'_{k+1} &= f(satr(J_k \cup \{a_{k+1} = y\})) \setminus E_{tr} \\ &= satr(f'(satr(J_k \cup \{a_{k+1} = y\} \setminus (TU \cup satr(AR)))) \setminus L_{tr} \setminus AR_{NOT}) \\ &\quad \cup TU \cup AR) \setminus E_{tr} \end{aligned} \tag{A.403}$$

Therefore, since $TU \cup AR \subseteq E_{tr}$, and f' is a consequence function of Π', it is sufficient to show

$$W' \text{ satisfies } \; satr(J_k \cup \{a_{k+1} = y\}) \setminus (TU \cup satr(AR)) \tag{A.404}$$

Since f is defined on $satr(J_k \cup \{a_{k+1} = y\})$, we have:

$$satr(J_k \cup \{a_{k+1} = y\}) \cap L_{tr} = \emptyset \tag{A.405}$$

and

$$satr(J_k \cup \{a_{k+1} = y\}) \cap AR_{NOT} = \emptyset \tag{A.406}$$

Therefore, by minimality of saturation $satr(J_k \cup \{a_{k+1} = y\}) \setminus (TU \cup satr(AR))$ does not contain literals formed by *do* and *obs* not preceded by default negation. W' satisfies all such literals. Also, from (A.405), $satr(J_k \cup \{a_{k+1} = y\}) \setminus (TU \cup satr(AR))$ does not contain literals formed by attribute terms from R. Therefore,

$$W' \text{ satisfies } satr(J_k \cup \{a_{k+1} = y\}) \setminus (TU \cup satr(AR)) \cap E_{tr} \tag{A.407}$$

We have: Let $Z = TU \cup satr(AR)$

$$\begin{aligned} satr(J_k \cup \{a_{k+1} = y\}) \setminus TU \setminus satr(AR) &= satr(J'_k \cup Y \cup \{a_{k+1} = y\}) \setminus Z \\ &= satr(J'_k \cup \{a_{k+1} = y\}) \cup satr(Y) \setminus Z \end{aligned}$$

for some Y consisting of e-literals from E_{tr}.

From (A.397) and (A.402) by Lemma 36 we have

$$W' \text{ satisfies } satr(J'_k \cup \{a_{k+1} = y\}) \tag{A.408}$$

From (A.408) and (A.407) we have (A.404).

Therefore, (A.398) holds.

□

Lemma 46. Let Π be a program from $\mathcal{B}$ with signature Σ. Let Π' be the program obtained from Π by removing activity records. Let $a_1, \ldots, a_m$ be a probabilistic leveling of Π' satisfying condition from Definition 20. Let $T_1, \ldots, T_m$ be the sequence of trees described in the proof of lemma 8 such that $T_m = T$ is a tableau which represents Π'. Let f be an admissible consequence function of Π, and I be an e-interpretation of Σ reachable w.r.t f. Let A be the set of atoms of I formed by random attribute terms of Π. There exists a leaf node n of T such that $A \subseteq p_T(n)$.

□

Proof. For a subset A' of A, let $h(A')$ denote the statement:

$$\text{there exists a leaf node } n \text{ of } T \text{ such that } A' \subseteq p_T(n)$$

Clearly, $h(\{\})$ holds.(by construction, T is non-empty and $\{\}$ is a subset of any leaf node of T).

We will prove the lemma by defining an order $\prec$ of subsets of A such that:

$$\text{for any proper subset } A' \text{ of } A,\ A' \prec A \tag{A.409}$$

$$\text{for any subset } A' \text{ of } A,\ \neg A' \prec A' \tag{A.410}$$

$$\text{for any three subsets } A_1, A_2, A_3 \text{ of } A,\ A_1 \prec A_2 \wedge A_2 \prec A_3 \text{ implies } A_1 \prec A_3 \tag{A.411}$$

and showing that if for a proper subset A' of A, there exists a node n of T such that $A' \subseteq p_T(n)$, then there exists another subset A'' of A such that:

(a) $A' \prec A''$

(b) there exists a node n' of T such that $A'' \subseteq p_T(n')$

The existence of A'' for every subset A' of A satisfying (a)-(b) and the properties (A.409) - (A.411) of $\prec$ imply that there exists a node n_f of T such that $A \subseteq p_T(n_f)$. Therefore, for any leaf node n_l of T which is descendant of n_f, $A \subseteq p_T(n_l)$. Since for any node of a tree there exists at least one leaf descendant, the lemma holds.

1. We define $\prec$ satisfying (A.409) - (A.411). Let $A' = \{b_1, \ldots, b_j\}$ be a subset of A. By for a random attribute term a of Π, by $it(a)$ we denote the index of a in $a_1, \ldots, a_m$. (that is, $it(a) = i$ iff $a = a_i$). We will first define a function $val : 2^A \rightarrow \{0..(2^{|A|} - 1)\}$:

$$val(A') = \sum_{i=1}^{j} 2^{m - it(b_j)} \tag{A.412}$$

The relation $\prec$ is defined as follows:

$$A_1 \prec A_2 \text{ iff } val(A_1) < val(A_2) \tag{A.413}$$

It is easy to check that the subsets of A are in one-to-one correspondence with numbers in $\{0..(2^{|A|} - 1)\}$ obtained by applying the function val to them.

Therefore,

$$\text{for any two subsets } A_1 \neq A_2 \text{ of } A,\ val(A_1) \neq val(A_2) \tag{A.414}$$

and (A.410) holds. Also, (A.409) holds because $val(A) > val(A')$ for any subset A' of A.

Therefore, since $<$ is transitive, (A.411) holds.

2. Let $A' = \{b_1, \ldots, b_k\}$ be a proper subset of A. Let d a the node of T such that $A' \subseteq p_T(d)$. We show that there exists another subset A'' of A such that:

   (a) $A' \prec A''$, and

   (b) there exists a node d' of T such that $A'' \subseteq p_T(d')$.

   Let $CI = I_0, I_1, \ldots, I_n$ be the corresponding sequence for I, and $ats(J_0, \ldots, J_n)$ be $a_{s_1} = y_1, \ldots, a_{s_n} = y_n$.

   Let j be the smallest integer such that

$$y_j \neq u \tag{A.415}$$

   and

$$a_{s_j} = y_j \notin A' \tag{A.416}$$

   Note that, since A' is a proper subset of A, j exists and $j \leq n$.

   By definition of corresponding sequence,

$$I_j \text{ is a consequence of } satr(I_{j-1} \cup \{a_{s_j} = y_j\}) \tag{A.417}$$

   where

$$a_{s_j} \text{ is ready in } I_{j-1} \tag{A.418}$$

and

$$y_j \text{ is a possible value of } a_{s_j} \text{ in } I_{j-1} \tag{A.419}$$

From (A.415) and (A.418) we have that a_{s_j} is active in I_{j-1}. Therefore, there exists a unique rule r of the form

$$random(a_{s_j} : \{X : p(X)\}) \leftarrow B \tag{A.420}$$

such that

$$B \text{ is satisfied by } I_{j-1} \tag{A.421}$$

$$\text{every atom of the form } p(y_1)\text{, where } y_1 \in range(a_{s_j})\text{, is decided in } I_{j-1} \tag{A.422}$$

$$p(y_j) \text{ is satisfied by } I_{j-1} \tag{A.423}$$

$$\text{for every } y \in PO(I_{j-1}, r, a_{s_j}),\ I_{j-1} \cup \{a_{s_j} = y\} \text{ is consistent} \tag{A.424}$$

and

$$\text{the body of every pr-atom for } a_{s_j} \text{ is either satisfied or falsified by } I_{j-1} \tag{A.425}$$

Let d_l be a leaf of T with ancestor d. By Lemma 18, there exists a unique possible world W_k of Π' such that

$$W_k \text{ satisfies } p_T(d_l) \tag{A.426}$$

Let I'_{j-1} be obtained from I_{j-1} by removing:

(a) e-literals formed by $truly\_random(a)$ for every a such that $do(a = y) \in \Pi$,

(b) e-literals formed by $do(a)$ such that $do(a) \in \Pi$,

(c) e-literals formed by $obs(l)$ such that $obs(l) \in \Pi$

Since j is the smallest index such that $y_j \neq u$ and $a_{s_j} = y_j \notin A'$, every atom from $a_{s_1} = y_1, \dots, a_{s_{j-1}} = y_{j-1}$ belongs to A'. Since $A' \subseteq p_T(d) \subseteq p_T(d_l)$, by lemma (45) we have:

$$W_k \text{ is compatible with } I'_{j-1} \tag{A.427}$$

Since B does not contain literals formed by $truly\_random$, do and obs, from (A.427) and (A.421) by lemma 32 we have:

$$W_k \text{ satisfies } B \tag{A.428}$$

Let $n_{it(a_{s_j})-1}$ be the ancestor of d_l such that

$$n_{it(a_{s_j})-1} \text{ is a leaf node in } T_{it(a_{s_j})-1} \tag{A.429}$$

We will prove

$$n_{it(a_{s_j})-1} \text{ is ready to branch on } a_{s_j} \text{ relative to } \Pi' \tag{A.430}$$

We will prove (A.431) by contradiction. Suppose

$$n_{it(a_{s_j})-1} \text{ is not ready to branch on } a_{s_j} \text{ relative to } \Pi' \tag{A.431}$$

In this case, by construction of T,

$$p_T(d_l) \text{ contains no atoms formed by } a_{s_j} \tag{A.432}$$

Since W_k is the only possible world satisfying $p_T(d_l)$, from (A.428) we have:

$$p_T(d_l)\ \Pi'\text{-guarantees } B \tag{A.433}$$

From (A.423) and (A.427) we have:

$$W_k \text{ satisfies } p(y_j) \tag{A.434}$$

Therefore, since W_k is the only possible world of Π' containing $p_T(d_l)$:

$$p_T(d_l)\ \Pi'\text{-guarantees } p(y_j) \tag{A.435}$$

From (A.435), (A.433) (A.432) and the fact that W_k is the only possible world of Π' satisfying $p_T(d_l)$ we have

$$d_l \text{ is ready to branch on } a_{s_j} \text{ relative to } \Pi' \tag{A.436}$$

Therefore, T is a not a tableau of Π' (it is not maximal with respect to subtree relation), which is a contradiction. Therefore, (A.431) does not hold, and (A.430) holds.

We next show:

$$n_{it(a_{s_j})-1}\ \Pi' \text{ -guarantees } p(y_{s_j}) \tag{A.437}$$

From (A.436) we have that $p_T(d_l)$ has an atom formed by a_{s_j}. Since W_k satisfies

$p_T(d_l)$ $n_{it(a_{s_j})-1}$ is an ancestor of d_l, we have that W_k satisfies $p_T(n_{it(a_{s_j})-1})$. Therefore, from (A.430) and (A.434) we have (A.437).

Therefore, by construction, T contains a child n_c of $n_{it(a_{s_{j+1}})-1}$ such that

$$p_T(n_c) = p_T(n_{it(a_{s_{k+1}})-1}) \cup \{a_{s_j} = y_j\}.$$

Now consider the set of atoms $A^n = A \cap p_T(n_c)$. Clearly there exists a node of T which contains A^n (it's n_c). Therefore, we only need to show

$$A' \prec A^n \tag{A.438}$$

By construction, the nodes d, $n_{it(a_{s_j})-1}$ and d_l belong to the same path from the root of T to its leaf n_f. Let us denote this path by P. Consider the sets $A_{up} = A' \cap p_T(n_{it(a_{s_j})-1})$ and $A_{down} = A' \setminus p_T(n_{it(a_{s_j})-1})$. Since n_c is a child of $n_{it(a_{s_j})-1}$, it is sufficient to show that:

$$\text{for every } a = y \in A_{down}, it(a) > it(a_{s_j}) \tag{A.439}$$

If $n_{it(a_{s_j})-1}$ has no children in P, then, since d belongs to P, $A' \subseteq p_T(d)$, A_{down} is empty, and (A.439) is vacuously true.

Let n'_c be the child of $n_{it(a_{s_j})-1}$ in P. Since $n_{it(a_{s_j})-1}$ is ready to branch on a_{s_j} and belongs to the tree $T_{it(a_{s_j})-1}$, $p_T(n'_c) = p_T(n_{it(a_{s_j})-1}) \cup \{a_{s_j} = y'\}$ for some y'. Therefore, since every node A_{down} does not belong to $p_T(n_{it(a_{s_j})-1})$, and no node of A_{down} is equal to $a_{s_j} = y'$ (since no atom in A' is formed by a_{s_j}), every atom of A_{down} labels an edge below n'_c. Therefore, by construction of the tree, we have (A.439).

□

Proposition 7 Let f be an admissible consequence function of Π. Let I be a definite i-node of $T_\Pi\langle f\rangle$. I is incompatible iff there exists an axiom in $\mathcal{X}(\Pi)$ whose body is satisfied by I.

□

Proof. $\Rightarrow$ We will prove that if I is incompatible, then there exists an axiom in $\mathcal{X}(\Pi)$ whose body is satisfied by I. We prove the contrapositive: if the body of every axiom in $\mathcal{X}(\Pi)$ is not satisfied by I, then I is compatible. Suppose the body of every axiom in $\mathcal{X}(\Pi)$ is not satisfied by I. Since I is a definite node, we have:

$$I \text{ falsifies the body of every axiom in } \mathcal{X}(\Pi) \tag{A.440}$$

Let U be the set of activity records of Π. Let Π' be the program $\Pi \setminus U$. By construction, I is reachable w.r.t f. Let A be the set of atoms in I formed by random attribute terms of Π. Let $T_1, \ldots, T_m$ be the sequence of trees described in the proof of lemma 8 such that $T_m = T$ is a tableau which represents Π'. By lemma (46), there exists a leaf node n of T such that $A \subseteq p_T(n)$. Let W' be the possible world of Π' represented by n. Let I' be an interpretation obtained from I by removing all e-literals formed by $truly\_random(a)$ for every a such that $do(a = y) \in \Pi$. By Lemma 45,

$$W' \text{ is compatible with } I' \tag{A.441}$$

We next show:

$$\text{for every action } do(a = y) \text{ of } \Pi,\ W' \text{ satisfies } a = y \tag{A.442}$$

The axiom

$$\leftarrow do(a, y), not\ a = y$$

belongs to $\mathcal{X}(\Pi)$.

By (A.440), the fact that I is in the image of an admissible consequence function that includes all activity records of Π, we have that I satisfies $a = y$. By construction if I', I' satisfies $a = y$. Therefore, by (A.441) we have (A.442). Similarly, using the other axioms:

$$\leftarrow obs(l), not\ l$$

and

$$\leftarrow do(a, y), not\ random(a, p_1), \ldots, random(a, p_n)$$

we can show:

$$\text{for every observation } obs(l) \text{ of } \Pi,\ W' \text{ satisfies } l \tag{A.443}$$

and

$$\text{for every action } do(a = y) \text{ of } \Pi,\ W' \text{ satisfies } random(a, p) \text{ for some } p \tag{A.444}$$

From (A.442) - (A.444) by lemma (31) we have that the set of atoms:

$$W = W' \setminus \{truly\_random(a) \mid do(a, y) \in \Pi\} \cup U \tag{A.445}$$

is a possible world of Π.

It is sufficient show that:

$$W \text{ is compatible with } I \tag{A.446}$$

Let A_1 be the set of attribute terms from activity records of Π, and A_2 be the set of attribute terms

$$\{truly\_random(a) \mid do(a = y) \in \Pi\}$$

From (A.441) by construction of I' and W, W satisfies all e-literals from I that are not formed by attribute terms from $A_1 \cup A_2$. In 1 and 2 we show that W satisfies all e-literals from I formed by attribute terms from A_1 and A_2 respectively.

1. We prove that

$$W \text{ satisfies all e-literals from } I \text{ formed by attribute terms from } A_1 \quad \text{(A.447)}$$

   Since I is reachable via f, we have $I = f(I')$ for some interpretation I', where

$$I = f(I') = satr((f'(I' \setminus (TU \cup satr(AR)) \setminus L_{tr} \setminus AR_{NOT}) \cup TU \cup AR) \quad \text{(A.448)}$$

   By lemma (39), we have:

$$satr(AR) \subseteq I \quad \text{(A.449)}$$

   Since W is a possible world of Π, W satisfies AR. Therefore, by lemma 36,

$$W \text{ satisfies } satr(AR) \quad \text{(A.450)}$$

   Since every e-literal formed by A_1 not belonging to $satr(AR)$ is contrary to an e-literal from $satr(AR)$, and I is a consistent set of e-literals, from (A.450) we have (A.447).

2. We prove that

$$W \text{ satisfies all e-literals from } I \text{ formed by attribute terms from } A_2 \quad \text{(A.451)}$$

   Similarly to 1, from (A.448) we can see that I can only contain e-literals formed by A_2 belonging to TU. Therefore, by Lemma 41, W satisfies all such e-literals.

$\Leftarrow$ Suppose there exists an axiom r in $\mathcal{X}(\Pi)$ whose body is satisfied by I. Suppose there exists a possible world compatible with I. By Proposition 4, the possible world does not satisfy r, which contradicts Proposition 1.

□

A.3.6 Proof of Proposition 8

Lemma 47. Let Π be a program with signature Σ, I be an e-interpretation of Σ such that every random attribute term of Π is decided in I. There exists at most one possible world of Π compatible with I.

□

Proof. Let A be the set of atoms in I formed by random attribute terms of Π. For the sake of contradiction, suppose there exists two different possible worlds, W_1 and W_2 of Π, compatible with I. We have:

$$A \text{ is the set of atoms in } W_1 \text{ formed by random attribute terms of } \Pi \tag{A.452}$$

$$A \text{ is the set of atoms in } W_2 \text{ formed by random attribute terms of } \Pi \tag{A.453}$$

Let Π' be a program obtained from Π by removing activity records. By lemma 37, there exists two possible worlds, W_1' and W_2' of Π', which coincide with W_1 and W_2 respectively on the atoms formed by random attribute terms. Therefore, from (A.452) and (A.453) we have a contradiction to Lemma 18.

□

Let Π be a program from $\mathcal{B}$ with signature Σ.

Lemma 48. Let I be a reachable interpretation of Σ and $I_0, \ldots, I_n$ be a corresponding sequence for I. For every random attribute term a of Π we have:

1. for every e-literal l in I, if a is formed by random attribute term, then a is decided in I,
2. $a = y \in I$, then $a = y$ is an element of $ats(I_0, \ldots, I_n)$
3. if $a = u \subseteq I$, then $a = u$ is an element of $ats(I_0, \ldots, I_n)$.

□

Proof. This follows immediately from the definition of a reachable sequence (Definition 75). □

If $I_0, \ldots, I_n$ is a reachable sequence with $ats(I_0, \ldots, I_n) = a_1 = y_1, \ldots, a_n = y_n$, and $a = y$ is an atom in I_n , then we will refer to i such that $a_i = a$ and $y_i = y$ as the *spot* of $a = y$ in $I_0, \ldots, I_n$.

Lemma 49. Let $I_0, \ldots, I_n$ be a reachable sequence of program Π. We have:

$$\text{for every } i, j \in \{0..n\} \text{ such that } i \leq j,\ I_i \subseteq I_j$$

□

Proof. It is sufficient to show that for every $i \in \{0..n\}$, $I_i \subseteq I_{i+1}$. The lemma then follows immediately from the reflexivity and transitivity of $\subseteq$. By definition of a reachable sequence, we have I_{i+1} is a consequence of $satr(I_i \cup \{a = y\})$ for some atom $a = y$. By Lemma (39) we have:

$$I_i \subseteq satr(I_i \cup \{a = y\}) \tag{A.454}$$

By clause 1 of Definition 36 we have:

$$satr(I_i \cup \{a = y\}) \subseteq I_{i+1} \tag{A.455}$$

From (A.454) and (A.455) we have:

$$I_i \subseteq I_{i+1} \tag{A.456}$$

□

Lemma 50. Let I be a reachable interpretation of Σ s.t. $\hat{\mu}^*(I)$ is defined. Let $a = y$ be an atom of Σ such that a is a random attribute term of Σ. We have $P(I, a = y)$ defined iff $truly\_random(a) \in I$. □

Proof. If $truly\_random(a) \notin I$, then, clearly, $P(I, a = y)$ is not defined. Suppose now $P(I, a = y)$ is not defined. Let $I_0, \ldots, I_n$ be a corresponding sequence for I. Let i be the spot of $a = y$ in $I_0, \ldots, I_n$. We have that a is active in I_{i-1}, which means:

$$B \text{ is satisfied by } I_{i-1} \tag{A.457}$$

$$\text{for every } y \in range(a),\ p(y) \text{ is decided in } I_{i-1} \tag{A.458}$$

$$\text{for every pr-atom } pr(a = y_1 \mid B_1) = v \text{ of } \Pi,\ B_1 \subseteq I_{i-1} \text{ or } B_1 \text{ is falsified by } I_{i-1} \tag{A.459}$$

By Lemma 49:

$$I_{i-1} \subseteq I \tag{A.460}$$

From (A.460) and (A.457) - (A.459):

$$B \text{ is satisfied by } I \tag{A.461}$$

$$\text{for every } y \in range(a),\ p(y) \text{ is decided in } I \tag{A.462}$$

$$\text{for every pr-atom } pr(a = y_1 \mid B_1) = v \text{ of } \Pi,\ B_1 \subseteq I_i \text{ or } B_1 \text{ is falsified by } I \tag{A.463}$$

Since W is compatible with I, from A.460 we have:

$$W \text{ is compatible with } I_{i-1} \tag{A.464}$$

Since $a = y \in I$, and W is compatible with I, by Proposition 4 we have $a = y \in W$. From (A.458), (A.464) and lemma 33 and the fact that a is active in I_i we have:

$$p(y) \in I_{i-1} \tag{A.465}$$

From (A.465) and (A.460) we have:

$$p(y) \in I \tag{A.466}$$

Since $P(I, a = y)$ is not defined, one of the conditions from (5.6) - (5.9) has to be violated. Therefore, from (A.465), (A.461) - (A.463) we have that

$$truly\_random(a) \notin I \tag{A.467}$$

□

Lemma 51. Let Π be a program from $\mathcal{B}$. Let I be a reachable e-interpretation of Π such that:

1. $\hat{\mu}^*(I)$ is defined, and
2. for every random attribute term a decided in I, $truly\_random(a)$ is decided in I.

Let W be a possible world compatible with I. For every random attribute term decided in a we have:

1. if $P(I, a = y)$ is defined, then $P(W, a = y)$ defined and $P(W, a = y) = P(I, a = y)$

2. if $P(I, a = y)$ is undefined, then $P(W, a = y)$ is undefined

□

Proof. 1. Suppose $P(I, a = y)$ is defined. By condition (5.6) - (5.7) we have that there exists a rule r of Π such that:

$$y \in PO(I, r, a) \text{ for some random selection rule } random(a, p) \leftarrow B of \Pi \text{ such that } B \subseteq I \tag{A.468}$$

$$truly\_random(a) \in I, \tag{A.469}$$

for every pr-atom $pr(a = y_1 \mid B) = v$ of Π, either $B \subseteq I$, or B is falsified by I (A.470)

$$\text{for every } y \in range(a),\ p(y) \text{ is decided in } I \tag{A.471}$$

From (A.468) we have $B \subseteq I$, then by Proposition 4 we have:

$$W \text{ satisfies } B \tag{A.472}$$

From (A.468) we have $p(y) \subseteq I$, then by Proposition 4 we have:

$$W \text{ satisfies } p(y) \tag{A.473}$$

From (A.469) by Proposition 4 we have:

$$W \text{ satisfies } truly\_random(a) \tag{A.474}$$

Since $P(I, a = y)$ is defined, we have that

$$a = y \in I \tag{A.475}$$

From (A.475) by Proposition 4 we have:

$$a = y \in W \tag{A.476}$$

From (A.476), (A.472) - (A.474) we have that $P(W, a = y)$ is defined.

From (A.470) by Proposition 4 we have:

$$\text{for every pr-atom } pr(a = y_1 \mid B_1) = v \text{ of } \Pi,\ B_1 \subseteq I \text{ iff } W \models B_1 \tag{A.477}$$

From (A.471)) by Proposition 4 we have:

$$\text{for every } y \in range(a),\ p(y) \in I \text{ iff } p(y) \in W \tag{A.478}$$

From (A.477) and (A.478) we have $P(I, a = y) = P(W, a = y)$.

2. Suppose $P(I, a = y)$ is undefined. By lemma 50, $truly\_random(a) \notin I$. Since $truly\_random(a)$ is decided in I, we must have $truly\_random(a) = u \in I$ or $truly\_random(a) = falseinI$. In both cases, by Proposition 4, W does not satisfy $truly\_random(a)$. Therefore, $P(W, a = y)$ is undefined.

□

Lemma 52. Let Π be a program with signature Σ. Let I be a compatible reachable e-interpretation of Σ. We have that $\hat{\mu}^*(I)$ is defined.

Proof. Let W be a possible world compatible with I. For the sake of contradiction, suppose $\hat{\mu}^*(I)$ is undefined. In this case, one of the conditions 4 -6 has to be violated for I. We consider each condition separately:

1. Suppose Condition 4 is violated for I. In this case there are two rules

$$random(a, p_1) \leftarrow B_1$$

and

$$random(a, p_2) \leftarrow B_2$$

such that $B_1, B_2 \subseteq I$. Therefore, by Proposition 4, we have $W \models B_1$ and $W \models B_2$. In this case, Condition 1 is violated for Π. Contradiction.

2. Suppose Condition 5 is violated for I. In this case there is a random selection rule $random(a, p_1) \leftarrow B$ and two pr-atoms $pr(a(\bar{t}) \mid B_1) = v_1$ and $pr(a(\bar{t}) \mid B_2) = v_2$ such that $B_1, B_2, B \subseteq I$. Therefore, from (A.480) we have $B_1, B_2, B \subseteq I$. In this case, Condition 2 is violated for Π. Contradiction.

3. Suppose Condition 6 is violated for I. In this case Π contains a random selection rule $r_1 : random(a, p) \leftarrow B_1$, a probability atom $pr(a = y| \ B_2) = v$. such that $B_1, B_2 \subseteq I$ and $p(y) \notin I$. Let $I_0, \ldots, I_n$ be the corresponding sequence for I. Let i be the spot of $a = y$ in $I_0, \ldots, I_n$. a is active in I_{i-1}. Therefore, there is a rule $r_2 : random(a, p_2) \leftarrow B_3$ s.t. I_{i-1} satisfies B_3 and $p_2(y)$ is decided in I_{i-1} for every $y \in range(a)$. Since W is compatible with both I_{i-1} and I, and Condition 3 is satisfied by W, we have $r_1 = r_2$. Therefore, $p = p_2$ and $p(y)$ is decided in I_{i-1} for every $y \in range(a)$. By Lemma 54, $p(y) \notin I_{i-1}$. Therefore, since $p(y)$ is decided in I_{i-1}, we have $p(y) = false$ or $p(y) = u$ in I_{i-1}. Since W is compatible with I_{i-1}, $p(y) \notin W$. Therefore, Condition 3 is violated for Π. Contradiction.

□

Lemma 53. Let I be a reachable interpretation of Σ s.t. $\hat{\mu}^*(I)$ is defined. Let $I_0, \ldots, I_n$ be a corresponding sequence of I. Let $a = y$ be an atom of Σ such that $P(I, a = y)$ is defined and i be the spot of $a = y$ in $I_0, \ldots, I_n$. $P(satr(I_{i-1} \cup a = y \cup truly\_random(a)), a = y)$ is defined. [3]

□

[3]Note that, since $P(I, a = y)$ is defined, we have $a = y \in I$ and $truly\_random(a) \in I$. We also have $I_i \subseteq I$ by Lemma 49. Therefore, by Lemma 39, $satr(I_i \cup a = y \cup truly\_random(a))$ is a subset of I, and, is therefore consistent.

Proof. We first prove that

$$\hat{\mu}^*(satr(I_{i-1} \cup a = y \cup truly\_random(a))) \text{ is defined} \tag{A.479}$$

We know:

$$satr(I_{i-1} \cup a = y \cup truly\_random(a)) \subseteq I \tag{A.480}$$

and that

$$u^*(I) \text{ is defined} \tag{A.481}$$

For the sake of contradiction suppose (A.479) doesn't hold. In this case, one of the Conditions 4 - 6 has to be violated. We consider each condition in 1-3 below.

1. Suppose Condition 4 is violated for $satr(I_i \cup a = y \cup truly\_random(a))$. In this case there are two rules $random(a, p_1) \leftarrow B_1$ and $random(a, p_2) \leftarrow B_2$ such that $B_1 \subseteq satr(I_i \cup a = y \cup truly\_random(a))$ and $B_2 \subseteq satr(I_i \cup a = y \cup truly\_random(a))$. Therefore, from (A.480) we have $B_1 \subseteq I$ and $B_2 \subseteq I$. In this case, Condition 4 is violated for interpretation I, and $\hat{\mu}^*(I)$ is undefined. Contradiction.

2. Suppose Condition 5 is violated for $satr(I_i \cup a = y \cup truly\_random(a))$. In this case there is a random selection rule $random(a, p_1) \leftarrow B$ and two pr-atoms $pr(a(\bar{t}) \mid B_1) = v_1$ and $pr(a(\bar{t}) \mid B_2) = v_2$ such that $B_1, B_2, B \subseteq satr(I_i \cup a = y \cup truly\_random(a))$. Therefore, from (A.480) we have $B_1, B_2, B \subseteq I$. In this case, Condition 5 is violated for interpretation I, and $\hat{\mu}^*(I)$ is undefined. Contradiction.

3. Suppose Condition 6 is violated for $satr(I_i \cup a = y \cup truly\_random(a))$. In this case Π contains a random selection rule $r_1 : random(a, p) \leftarrow B_1$, a probability atom $pr(a = y | B_2) = v$. such that $B_1, B_2 \subseteq satr(I_i \cup a = y \cup truly\_random(a))$ and $p(y) \notin satr(I_{i-1} \cup a = y \cup truly\_random(a))$. Since i is the spot of $a = y$ in $I_0, \ldots, I_{i-1}$, a is active in I_{i-1}. Therefore, there is a rule $r_2 : random(a, p_2) \leftarrow B_3$

s.t. I_{i-1} satisfied B_3 and $p_2(y)$ is decided in I_{i-1} for every $y \in range(a)$. Since $\hat{\mu}^*(I)$ is defined, and $I_{i-1} \subseteq satr(I_{i-1} \cup a = y \cup truly\_random(a)) \subseteq I$, by condition 4 we have: $r_1 = r_2$. Therefore, $p = p_2$ and $p(y)$ is decided in I_{i-1} for every $y \in range(a)$. By Lemma 54, $p(y) \in satr(I_i \cup a = y \cup truly\_random(a))$ iff $p(y) \in I$. Therefore, $p(y) \in I$. Since $satr(I_i \cup a = y \cup truly\_random(a)) \subseteq I$, $B_1, B_2 \subseteq I$. Therefore, Condition 4 is violated for I, and $\hat{\mu}^*(I)$ is undefined. Contradiction.

Therefore, (A.479) holds. We show $P(satr(I_{i-1} \cup a = y \cup truly\_random(a)), a = y)$ is defined. Since a is active in I_{i-1}, by Definition 39, there is a rule $r : random(a, p) \leftarrow B$ such that:

$$I_{i-1} \text{ satisfies } B \tag{A.482}$$

$$\text{every attribute term } p(x) \text{, where } x \in range(a) \text{, is decided in } I_{i-1} \tag{A.483}$$

$$\text{for every pr-atom } pr(a = y' \mid B_2) = v, B_2 \subseteq I_{i-1} \text{ or } B \text{ is falsified by } I_{i-1} \tag{A.484}$$

Since i is the spot of $a = y$, By clause 2 of Definition 75 we have:

$$y \in PO(I_{i-1}, r, a) \tag{A.485}$$

We prove Conditions (5.6) - (5.9) in 1-4 respectively.

1. From (A.480), (A.482) and (A.485) we have $satr(I_{i-1} \cup a = y \cup truly\_random(a))$ satisfies B, and $y \in PO(satr(I_{i-1} \cup a = y \cup truly\_random(a)))$. Therefore, (5.6) is satisfied.

2. Clearly, $truly\_random(a) \in satr(I_{i-1} \cup a = y \cup truly\_random(a))$. Therefore,

(5.7) is satisfied.

3. From (A.480), (A.484) and (A.482) we have that for every pr-atom $pr(a = y' \mid B_2) = v$, $B_2 \subseteq I$ or B is falsified by I. Therefore, (5.8) is satisfied.

4. By condition 1, r is the only random selection rule whose body is satisfied by $satr(I_{i-1} \cup a = y \cup truly\_random(a))$. By (A.480), (A.483) every attribute term $p(y)$ s.t. $y \in range(a)$ is decided in $satr(I_{i-1} \cup a = y \cup truly\_random(a))$. Therefore, condition (5.9) is satisfied.

□

Before moving to the next lemma, we introduce a definition.

Definition 77 (Future probability)**.**
Let I be a reachable interpretation of Σ s.t. $\hat{\mu}^*(I)$ is defined. Let $I_0, \ldots, I_n$ be a corresponding sequence of I. Let $a = y$ be an atom of Σ such that $P(I, a = y)$ is defined and i be the spot of $a = y$ in $I_0, \ldots, I_n$. For each possible value y of a in I_{i-1}, The *future probability* of $a = y$ in I_{i-1}, $P^*(I_{i-1}, a = y)$, is equal to $P(satr(I_{i-1} \cup a = y \cup truly\_random(a)), a = y)$.

□

Lemma 54. Let I, J be two e-interpretations of Σ such that $I \subseteq J$, and a be an attribute term decided in I. An atom $a = y$ belongs to I iff $a = y$ belongs to J. □

Proof. Suppose $a = y \in I$. Since $I \subseteq J$, $a = y \in J$.

Suppose $a = y \in J$. Since a is decided in I, we have that

$$\exists y' \in range(a) : a = y' \in I, \text{ or } a = u \in I \tag{A.486}$$

If $a = u \in I$, then $a = u \in J$, which is impossible because $a = y \in J$, and *not* $a = y$ and $a = y$ are contrary. Therefore,

$$\exists y' \in range(a) : a = y' \in I \tag{A.487}$$

It is impossible to have $y' \neq y$, because in that case from $I \subseteq J$ we will have $a = y \in J$ and $a = y' \in J$, two contrary literals in J. Therefore, $a = y \in J$.

□

Lemma 55. Let I be a reachable interpretation of Σ s.t. $\hat{\mu}^*(I)$ is defined. Let $I_0, \ldots, I_n$ be a corresponding sequence of I. Let $a = y$ be an atom of Σ such that $P(I, a = y)$ is defined and i be the spot of $a = y$ in $I_0, \ldots, I_n$. We have that $P^*(I_{i-1}, a = y)$ is defined and $P(I, a = y) = P^*(I_{i-1}, a = y)$

□

Proof. By lemma 53, $P^*(I_{i-1}, a = y)$ is defined. Since $\hat{\mu}^*(I)$ and $P(I, a = y)$ are defined, by Condition 4 and (5.6), there is a unique rule of the form:

$$random(a, p) \leftarrow B$$

s.t.

$$I \text{ satisfies } B \quad \text{(A.488)}$$

Since i is the spot of $a = y$ in $I_0, \ldots, I_n$, a is active in I_{i-1}. Therefore, by Definition 39,

$$a_{i-1} \text{ is not decided in } I_{i-1} \quad \text{(A.489)}$$

and there exists a rule $r' : random(a, p') \leftarrow B'$ in Π such that:

$$I_{i-1} \text{ satisfies the body of } r' \quad \text{(A.490)}$$

$$\text{every attribute term } p'(x), \text{ where } x \in range(a), \text{ is decided in } I_{i-1} \quad \text{(A.491)}$$

$$\text{for every pr-atom } pr(a = y \mid K) \text{ of } \Pi,\ K \subseteq I_{i-1} \text{ or } K \text{ is falsified by } I_{i-1} \quad \text{(A.492)}$$

By Lemma 49, we have:

$$I_{i-1} \subseteq I \tag{A.493}$$

From (A.493) and (A.490) we have:

$$I \text{ satisfies the body of } r' \tag{A.494}$$

Since $\hat{\mu}^*(I)$ is defined, by Condition 4 from (A.494) and (A.488) we have:

$$r = r' \tag{A.495}$$

From (A.491) and (A.493) by Lemma 54 we have:

$$\text{for every atom } p(x)\text{, where } x \in range(a)\text{, } p(x) \in I \text{ iff } p(x) \in I_{i-1} \tag{A.496}$$

From (A.492) and (A.493) by Lemma 54 we have:

$$\text{for every pr-atom } pr(a = y \mid K) \text{ of } \Pi\text{, } K \subseteq I_{i-1} \text{ iff } K \subseteq I \tag{A.497}$$

By definition of future probability:

$$P^*(I_{i-1}, a = y) = P(satr(I_{i-1} \cup a = y \cup truly\_random(a)), a = y) \tag{A.498}$$

By Lemma 39 we have:

$$I_{i-1} \subseteq satr(I_{i-1} \cup a = y \cup truly\_random(a)) \tag{A.499}$$

From (A.491) and (A.499) we have by Lemma 54:

$$\text{for every atom } p(x)\text{, where } x \in range(a)\text{, } p(x) \in I_{i-1} \text{ iff } \\ p(x) \in satr(I_{i-1} \cup a = y \cup truly\_random(a)) \tag{A.500}$$

From (A.500) and (A.496) we have:

for every atom $p(x)$, where $x \in range(a)$, $p(x) \in I$ iff

$$p(x) \in satr(I_{i-1} \cup a = y \cup truly\_random(a)) \tag{A.501}$$

From (A.499), (A.495) and (A.490) we have:

$$satr(I_{i-1} \cup a = y \cup truly\_random(a)) \text{ satisfies the body of } r \tag{A.502}$$

From (A.492) and (A.499) we have by Lemma 54:

for every pr-atom $pr(a = y \mid K)$ of Π, $K \subseteq I_{i-1}$ iff

$$K \subseteq satr(I_{i-1} \cup a = y \cup truly\_random(a)) \tag{A.503}$$

From (A.503) and (A.497) we have:

for every pr-atom $pr(a = y \mid K)$ of Π, $K \subseteq I$ iff

$$K \subseteq satr(I_{i-1} \cup a = y \cup truly\_random(a)) \tag{A.504}$$

From (A.504), (A.501), (A.502) and (A.488) we have:

$$P(I, a = y) = P(satr(I_{i-1} \cup a = y \cup truly\_random(a)), a = y) \tag{A.505}$$

Therefore, from (A.498):

$$P(I, a = y) = P^*(I_{i-1}, a = y) \tag{A.506}$$

□

Lemma 56. Let Π be a program from $\mathcal{B}$, $I_0, \dots, I_n$ and $J_0, \dots, J_n$ be two reachable sequences of Π such that:

1. $ats(I_0, \dots, I_n) = ats(J_0, \dots, J_n)$
2. for every $i < n$, $I_i = J_i$
3. $I_n \subseteq J_n$
4. for every random attribute term a decided in I_n, $truly\_random(a)$ is decided in I_n
5. I_n is compatible

We have:

1. $\Omega^{I_n} = \Omega^{J_n}$
2. if I_n is compatible and $\hat{\mu}^*(I_n)$ is defined, then $\hat{\mu}^*(J_n)$ is defined, and $\hat{\mu}^*(J_n) = \hat{\mu}^*(I_n)$

Proof. 1. We prove 1. Consider two cases:

(a) $n = 0$. In this case I_n and J_n are both consequence of $\{\}$. Therefore, every possible of Π is compatible with both I_n and J_n, and we have $\Omega^{I_n} = \Omega^{J_n}$.

(b) $n > 0$. In this case

$$I_n \text{ is a consequence of } I_{n-1} \cup a_1 = y_1 \tag{A.507}$$

$$J_n \text{ is a consequence } J_{n-1} \cup a_2 = y_2 \tag{A.508}$$

Since for every $i < n$, $I_i = J_i$, from (A.508) we have:

$$J_n \text{ is a consequence } I_{n-1} \cup a_2 = y_2 \tag{A.509}$$

Since $ats(I_0, \ldots, I_n) = ats(J_0, \ldots, J_n)$, from (A.509) we have:

$$J_n \text{ is a consequence } I_{n-1} \cup a_1 = y_1 \tag{A.510}$$

From (A.507), (A.508), (A.510) we have:

$$\Omega^{I_{n-1} \cup a_1 = y_1} = \Omega^{I_n} = \Omega^{J_n} \tag{A.511}$$

2. Suppose I_n is compatible, $\hat{\mu}^*(I_n)$ is defined. We will prove that $\hat{\mu}^*(J_n)$ is defined, and $\hat{\mu}^*(J_n) = \hat{\mu}^*(I_n)$ From (A.511) we have that J_n is compatible. Therefore, by Lemma 52, $\hat{\mu}^*(J_n)$ is defined. If $n = 0$, both I_n and J_n have no atoms formed by random attribute terms, so $\hat{\mu}^*(J_n) = \hat{\mu}^*(I_n) = 1$. Suppose now $n > 0$. Let A_I (A_J) be the set of atoms s.t. for every $a \in A$, $P(I_n, a)$ ($P(J_n, a)$). By Lemma 48, since $ats(I_0, \ldots, I_n) = ats(J_0, \ldots, J_n)$, we have that the collections of atoms formed by random attribute terms coincide in both I_n and J_n. By Lemma 54, $truly\_random(a) \in I_n$ iff $truly\_random(a) \in J_n$. Therefore,

$$A_I = A_J \tag{A.512}$$

Now we will prove:

$$\text{for every } a \in A_I,\ P(I_n, a) = P(J_n, a) \tag{A.513}$$

Let a be an atom in A_I, and Let i be the spot of a in $I_0, \ldots, I_n$. By Lemma 55,

$$P(I_n, a) = P^*(I_{i-1}, a) \tag{A.514}$$

Since $ats(I_0, \ldots, I_n) = ats(I_0, \ldots, I_n)$, i is the spot of a in $J_0, \ldots, J_n$, and, by Lemma 55,

$$P(J_n, a) = P^*(J_{i-1}, a) \tag{A.515}$$

Since $j \leq n$, we have $I_{i-1} = J_{i-1}$. Therefore, from (A.514) and (A.515), we have $P(I_n, a) = P(J_n, a)$. Therefore, (A.513) holds. From (A.513) and (A.512) we have $\hat{\mu}^*(I_n) = \hat{\mu}^*(J_n)$.

□

Lemma 57. Let Σ' be a signature consisting of a collection A of attribute terms from Σ. Let W be an interpretation of Σ'. Let I be an e-interpretation of Σ such that:

1. I contains all atoms of W_i that are not formed by *do*, *obs*, and *truly_random*

2. for every attribute term a such that:

   (a) W has no atoms formed by a,

   (b) a is from Σ'

   (c) a is not formed by *do*, *obs* or *truly_random*

   we have $a = u \subseteq I$

Let l be an e-literal from Σ' not formed by *do*, *obs* and *truly_random*. We have:

- if W satisfies l, then I satisfies l,

- if W does not satisfy l, then I falsifies l

□

Proof.

- We first prove the case W satisfies l. We consider all possible forms of l:

  1. l is $a = y$. In this case $a = y \in W$. By clause 1, $a = y \in I$.

  2. l is $a \neq y$. In this case $a = y_1 \in W$ for $y_1 \neq y$. By condition 1, $a = y_1 \in I$. Since I is saturated, $a \neq y \in I$.

3. l is *not* $a = y$. In this case we have only two possibilities:

   (a) W has no atoms formed by a. In this case by clause 2 *not* $a = y \in a = u \subseteq I$.

   (b) W has an atom $a = y_1$, for $y_1 \neq y$. In this case by clause 1 $a = y_1 \in I$. By clause 1 of Definition 29, $a \neq y \in I$. By clause 3 of Definition 29, $l \in I$.

4. l is *not* $a \neq y$. We have two cases:

   (a) W has no atoms formed by a. In this case by clause 2 $a = u \subseteq I$. By clause 5 of Definition 29, $l \in I$.

   (b) W has an atom $a = y_1$, for $y_1 \neq y$. In this case by clause 1 $a = y_1 \in I$. By clause 2 of Definition 29, *nota* $\neq y \in I$. Then by clause 4 of Definition 29 we have $l \in I$

- We next prove the case W does not satisfy l. We consider all possible forms of l:

1. l is $a = y$. We consider two cases:

   (a) W has no atoms formed by a. In this case $a = u \subseteq I$. Therefore, since $a = y$ is contrary to *not* $a = y$, I falsifies l.

   (b) W has an atom $a = y_1$ where $y_1 \neq y$. In this case $a = y_1 \in I$. Since $a = y_1$ and $a = y$ are contrary, I falsifies l.

2. l is $a \neq y$. We consider two cases:

   (a) W has no atoms formed by a. In this case $a = u \subseteq I$. Therefore, since $a \neq y$ is contrary to *not* $a \neq y$, I falsifies l.

   (b) W has an atom $a = y$. In this case $a = y \in I$. Since $a \neq y$ and $a = y$ are contrary, I falsifies l.

3. l is *not* $a = y$. In this $a = y \in W$, $a = y \in I$. Since $a = y$ and *not* $a = y$ are contrary, I falsifies l

4. l is *not* $a \neq y$. In this case W has an atom $a = y_1$ where $y_1 \neq y$. Therefore, $a = y_1 \in I$. By clause 1 of Definition 29, $a \neq y \in I$. Since $a \neq y$ and *not* $a \neq y$ are contrary, I falsifies l.

□

Lemma 58. Let Π be a program from $\mathcal{B}$ with signature Σ Let I be a compatible e-interpretation of Σ. Let Π' be obtained from Π by removing activity records. Let $a_1, \ldots, a_n$ is a probabilistic leveling of Π such that Π is dynamically causally ordered via $a_1, \ldots, a_n$. Let $|\ |$ be the total leveling of Π determined by $a_1, \ldots, a_n$. Let $\Pi_0, \ldots, \Pi_n$ be a dynamic structure of Π' induced by $a_1, \ldots, a_n$. Let W_i be a possible world of Π_i for some $i \in \{0..n-1\}$. If I satisfies the following conditions:

1. I contains no e-literals formed by a_{i+1}
2. I contains all atoms of W_i not formed by *random*, *do*, *obs*, and *truly_random*
3. for every attribute term a such that:
    - W_i has no atoms formed by a,
    - $|a| \leq i$, and
    - a is not formed by *do*, *obs* or *truly_random*, *random*

   we have $a = u \subseteq I$

then a_{i+1} is ready in I.

□

Proof. By clause 1 definition 20, there are only two possibilities:

- W_i falsifies the body B of every random selection rule $random(a_{i+1}, p) \leftarrow B$. We prove that, in this case, a_{i+1} is disabled in I. Condition 1 of Definition 40 is satisfied, because I contains no e-literals formed by a_{i+1}. Condition 2 follows from Lemma 57.

- for some random selection rule r: $random(a_{i+1}, p) \leftarrow B$ of Π we have:

$$W_i \text{ satisfies } B \tag{A.516}$$

By condition 1 from Definition 20 we have

$$r \text{ is active in } W_{i-1} \tag{A.517}$$

Therefore, by condition 2 of Definition 19

$$\text{for every } y \in range(a),\ |p(y)| \leq i \tag{A.518}$$

and, by condition 3 of the same definition:

$$\text{there exists } y \in range(a),\ p(y) \in W. \tag{A.519}$$

From condition 1 of Definition 20 we have:

$$\text{for every pr-atom } pr(a_{i+1} \mid B) = v,\ B \text{ is either falsified or satisfied by } W_i \tag{A.520}$$

In this case we will prove that a_{i+1} is active in I. We will prove all the conditions from Definition 39:

1. a_{i+1} is not decided by I – this is true by condition 1 from the lemma.
2. r satisfies the following conditions:
    (a) the head of r is of the form $random(a_{i+1}, p)$;
    (b) I satisfies B – this follows from (A.516) and Lemma 57;
    (c) every attribute term $p(x)$, where $x \in range(a)$, is decided in I – this follows from (A.518) and Lemma 57;

(d) $PO(I, r, a) \neq \emptyset$ – this follows from (A.519) and Lemma 57;

(e) for every $y \in PO(I, r, a)$, $satr(I \cup a = y)$ is consistent – this follows from the fact that I contains no e-literals formed by a.

3. For every probability atom

$$pr(a = y \mid B_2) = v$$

B_2 is either falsified or satisfied by I – this follows from (A.520) and Lemma 57.

□

Lemma 59. Let I be a compatible interpretation of Σ and attribute term a be active in I via rule r. All the possible worlds compatible with I belong to a unique scenario s for r. □

We will refer to the scenario from lemma 59 as the scenario determined by I.

Proof. Let W_1 be a possible world compatible with I, and s be a scenario of W for r. Let W_2 be another possible world compatible with I. We need to show

$$W_2 \in s \tag{A.521}$$

Let r be of the form

$$random(a, p) \leftarrow B$$

Since I is active in a, by condition 2 (c) of definition 39 we have:

$$\text{for every } x \in range(a),\ p(x) \text{ is decided in } I \tag{A.522}$$

and, by condition 3:

$$\text{for every pr-atom } pr(a = y \mid K) = v \in \Pi),\ K \text{ is decided by } I \tag{A.523}$$

If W_1 satisfies $p(y)$ for some $y \in range(a)$, then, by (A.522), $p(y) \in I$, and, since W_2 is compatible with I, $p(y) \in W_2$. If W_2 does not satisfy $p(y)$ for some $y \in range(a)$, then, by (A.522), $p(y) = u \in I$ or $p(y) = false \in I$ and, since W_2 is compatible with I, $p(y) \notin W_2$.

Similar arguments using (A.523) show that for each pr-atom $pr(a = y \mid K) = v \in \Pi)$, W_1 satisfies K iff W_2 satisfies K.

Therefore, $W_2 \in s$.

□

Lemma 60. Let Π be a program from $\mathcal{B}$ with signature Σ, f be an admissible consequence function of Π, I be an e-interpretation of Σ reachable via f. If

1. I is compatible and definite, and
2. for every random attribute term a decided in I, $truly\_random(a)$ is decided in I

then I is informative (see definition 53).

□

Proof. Let I be a reachable interpretation of Σ satisfying conditions 1-2 from the lemma. We will show that I is informative. Let $W_1, \ldots, W_k$ be the possible worlds of Π compatible with I.

For an interpretation J, Let $und(J)$ be the number of random attribute terms undecided in J. We will prove by induction on n the following claim (which implies the lemma immediately): *For any admissible consequence function f of Π, and any interpretation I reachable via f such that $und(I) = n$: if I satisfies conditions 1-2 from the lemma, then I is informative.*

Base Case $n = 0$. In this case I decides all random attribute terms of Π. Since I is compatible, there exists a possible world W of Π compatible with I. By Lemma 47,

$$W \text{ is the only possible world compatible with } I \tag{A.524}$$

By Lemma 52,

$\hat{\mu}^*(I)$ is defined.

Therefore, by Definition 52 we have:

$$\hat{\mu}^*_\Pi(I) = \prod_{a=y \in I} P(I, a = y) \tag{A.525}$$

where the product is taken over atoms for which $P(I, a = y)$ is defined.

Also, by Definition 3 we have:

$$\hat{\mu}_\Pi(W) = \prod_{a=y \in W} P(I, a = y) \tag{A.526}$$

where the product is taken over atoms for which $P(W, a = y)$ is defined.

Since every random attribute is decided in I, and W is compatible with I, by Lemma 51 from (A.525) and (A.526) we have:

$$\hat{\mu}(W) = \hat{\mu}^*(I) \tag{A.527}$$

Therefore, we have:

$$\begin{aligned} \hat{\mu}(I) &= \hat{\mu}(W) && \text{(by Def. 37 and (A.524))} \\ &= \hat{\mu}^*(I) && \text{(by } (A.527)) \end{aligned}$$

Ind. Hyp. Suppose that for any admissible consequence function f of Π, and any interpretation I reachable via f such that exactly k random attribute terms of Π are undecided in Π: if I satisfies conditions 1-2 from the lemma, then I is informative.

Ind. Step Let f be an admissible consequence function and I be an interpretation of Σ reachable via f such that:

$$und(I) = k + 1 \tag{A.528}$$

$$I \text{ is definite and compatible} \tag{A.529}$$

and

$$\begin{gathered}\text{for every random attribute term } a \text{ decided in } I,\\ truly\_random(a) \text{ is decided in } I\end{gathered} \tag{A.530}$$

We will prove:

$$I \text{ is informative} \tag{A.531}$$

Let $I_0, \ldots, I_h$ be a corresponding sequence for I having f as a consequence function, and $a_1 = v_1, \ldots, a_h = v_h$ be $ats(I_0, \ldots, I_h)$.

The rest of the proof will be organized as follows. In 1 we construct an admissible consequence function f_n of Π and an interpretation I^* is an e-interpretation reachable via f_n. In 2 we will prove

$$I \subseteq I^* \tag{A.532}$$

In 3 we will prove, using Lemma 56:

$$\hat{\mu}^*(I) = \hat{\mu}^*(I^*) \tag{A.533}$$

and

$$\hat{\mu}(I) = \hat{\mu}(I^*) \tag{A.534}$$

In 4 we will show, using Lemma 58, that a_u is ready in I^*. In 5 will will show that, if Y is the set of possible values of a_u in I^*, then each interpretation in the set $\{I^* \cup \{a_u = y\} \mid y \in Y\}$ is compatible. In 6 we will describe a new admissible function of Π, f_{nn}, construct a family $I^*_Y = \{f_{nn}(I^* \cup \{a_u = y\}) \mid y \in Y\}$ of e-interpretations reachable via f_{nn}, indexed by the set Y of the possible values of a_u in I^*, and will prove, using the result from 5, that:

- for each $y \in Y$, $und(I^*_Y) = k$
- each e-interpretation is I^*_Y is compatible, definite and, for every random attribute term decided in I^*_Y, $truly\_random(a)$ is decided. That is, by inductive hypothesis:

$$\text{the members of } I^*_Y \text{ are informative} \tag{A.535}$$

In 7 we will show that the measure of I^* is equal to the sum of measures of e-interpretations in I^*_Y:

$$\hat{\mu}(I^*) = \sum_{y \in Y} \hat{\mu}(I^*_y) \tag{A.536}$$

In 8 we will show that the candidate measure of I^* is equal to the sum of candidate measures of e-interpretations in I^*_Y:

$$\hat{\mu}^*(I^*) = \sum_{y \in Y} \hat{\mu}^*(I^*_y) \tag{A.537}$$

From (A.533) - (A.537) we get $\hat{\mu}(I) = \hat{\mu}^*(I)$. That is, (A.531) holds.

1. We first construct an admissible consequence function f_n of Π. Since f is an admissible consequence function of Π, there exists a consequence function f' of $\Pi \setminus AR$ such that f is induced by f'. Let $f'_* : int(\Sigma) \rightsquigarrow int(\Sigma)$ be a consequence function of Π' which computes all possible consequences w.r.t Π' of every compatible interpretation E as follows:

$$f'_*(E) = \{l \mid \forall W \in \Omega_{\Pi \setminus AR} : W \models E \Rightarrow W \models l$$
$$\text{and if } atf(l) \text{ is random, then } atf(l) \in atf(E)\}$$

In (a) and (b) we argue that $f'_*(E)$ is well defined for every compatible e-interpretation E, that is $f'_*(E)$ is an e-interpretation for each compatible E. In (a) we show that $f'_*(E)$ is consistent and in (b) that $f'_*(E)$ is saturated. Then we show that f'_* is a consequence function. In (c) we show $f'_*(\{\})$ is defined, in (d) that for every E, if $f'_*(E)$ is defined, than it is a consequence of E. And in (e) that $f'_*(E) \setminus E$ has no literals formed by random attribute terms

(a) since $W_E \models f'_*(E)$, and no interpretation satisfies two contrary literals, $f'_*(E)$ is consistent

(b) for the sake of contradiction, suppose for some E, $f'_*(E)$ is not saturated. In this case there exists an e-literal l in $f'_*(E)$ such that $satr(\{l\}) \not\subseteq f'_*(E)$. Let l' be an e-literal s.t.

$$l' \in satr(\{l\}) \tag{A.538}$$

and

$$l' \notin f'_*(E) \tag{A.539}$$

By construction of f'_*:

$$\forall W \in \Omega_{\Pi \setminus AR} : W \models E \Rightarrow W \models l \tag{A.540}$$

By Lemma 36:

$$\forall W \in \Omega_{\Pi \setminus AR} : W \models l \Rightarrow W \models satr(l) \tag{A.541}$$

From (A.541) and (A.538):

$$\forall W \in \Omega_{\Pi \setminus AR} : W \models l \Rightarrow W \models l' \tag{A.542}$$

From (A.540) and (A.542):

$$\forall W \in \Omega_{\Pi \setminus AR} : W \models E \Rightarrow W \models l' \tag{A.543}$$

Therefore, l' must be in $f'_*(E)$, which contradicts (A.539).

(c) From Lemma 18 it follows $\Pi \setminus R$ has a possible world. Therefore, $\{\}$ is compatible and, by construction, $f'_*(\{\})$ is defined.

(d) Clearly, $E \subseteq f'_*(E)$ because $\forall l \in E : \forall W \in \Omega\Pi \setminus AR : W \models E \Rightarrow W \models l$ and $\forall l \in E : atf(l) \in atf(E)$ by definition of atf. Moreover, $\forall l \in f'_*(E) : \forall W \in \Omega\Pi \setminus AR : W \models E \Rightarrow W \models l$. Therefore, $f'_*(E)$ is a consequence of E for every compatible interpretation E.

(e) By construction, if some random attribute term formed a literal l in $f_*(E)$, then $atf(l) \in atf(E)$. Therefore, $atf(f'_*(E)) \setminus atf(E)$ has no random attribute terms.

Using the set TU and AR as in Definition 50, we will define a function $f'_n : int(\Sigma) \rightsquigarrow int(\Sigma)$ as follows:

$$f'_n(E) = \begin{cases} f'(E), \text{ if } f'(E) \text{ is defined}, h > 0, \\ \qquad \text{and } E \neq \{I_{h-1} \cup b_h = v_h\} \setminus satr(AR) \setminus TU \\ f'_*(E), \text{ otherwise} \end{cases}$$

We have that

$$f'_n \text{ is a consequence function of } \Pi \setminus AR \tag{A.544}$$

because f' and f'_* are consequence functions of $\Pi \setminus AR$.

Let f_n be the admissible consequence function of Π induced by f'_n.

We will show that

$$\text{if } h > 0, f_n \text{ is defined on } \{I_{h-1} \cup b_h = v_h\} \tag{A.545}$$

By Lemma 39,

$(I_{h-1} \cup b_h = v_h) \setminus satr(AR) \setminus TU \subseteq I_{h-1} \cup b_h = v_h \subseteq I$. Therefore, from (A.559), $(I_{h-1} \cup b_h = v_h) \setminus satr(AR) \setminus TU$ is compatible (w.r.t $\Pi \setminus R$).

Therefore,

$$f'_n(\{I_{h-1} \cup b_h = v_h\} \setminus satr(AR) \setminus TU) \tag{A.546}$$

is defined.

Since f is defined on $\{I_{h-1} \cup b_h = v_h\}$, we have that conditions 1-2 from Definition 50 are satisfied for $\{I_{h-1} \cup b_h = v_h\}$. Therefore, from (A.546), (A.545) holds.

We next show:

$$\text{if } h = 0, f_n \text{ is defined on } \{\} \tag{A.547}$$

Any consequence function is defined on $\{\}$, so this is trivially holds. We next define I^* as follows:

$$I^* = \begin{cases} f_n(\{I_{h-1} \cup b_h = v_h\}), \text{ if } h > 0 \\ f_n(\{\}), \text{ otherwise} \end{cases}$$

Note that I^* is well defined by (A.545) - (A.547). Also, since, by construction f_n is an admissible consequence function of Π, we have:

$$I^* \text{ is an e-interpretation reachable via } f_n \tag{A.548}$$

Moreover, since, if $h > 0$ then $f_n(\{\}) = f(\{\})$ and for any $j \in \{0..h-2\}$ we have $f^*(I_j \cup \{b_j = v_j\})) = f(I_j \cup \{b_j = v_j\}))$ by construction of f, we have:

$$I_0, \ldots, I_{h-1}, I^* \text{ is a sequence reachable via } f_n \tag{A.549}$$

such that, since $b_h = v_h \in I^*$:

$$ats(I_0, \ldots, I_{h-1}, I^*) = ats(I_0, \ldots, I_{h-1}, I_h) \tag{A.550}$$

follows immediately from the facts that f' and f'_* are consequence functions of $\Pi \setminus AR$.

2. We will prove (A.532).

   Let J denote $\{\}$ if $h = 0$ and $I_{h-1} \cup \{b_h = v_h\}$ otherwise. We have:

$$\begin{aligned} I^* &= f_n(\{J\}) && \text{(A.551)} \\ &= satr((f'_n(J) \setminus L_{tr} \setminus AR_{NOT}) \cup TU \cup AR) \\ &= satr((f'_*(J) \setminus L_{tr} \setminus AR_{NOT}) \cup TU \cup AR) && \text{(A.552)} \end{aligned}$$

   and

$$\begin{aligned} I &= f(\{J\}) && \text{(A.553)} \\ &= satr((f'(J) \setminus L_{tr} \setminus AR_{NOT}) \cup TU \cup AR) \end{aligned}$$

   Suppose $l \in f'(J)$. Then, since f' is a consequence function of $\Pi \setminus R$

we have: $\forall W \in \Omega_{\Pi \setminus AR} : W \models J \Rightarrow W \models l$. Therefore, $l \in f'_*(J)$ by construction

$$f'(J) \subseteq f'_*(J) \tag{A.554}$$

From (A.551), (A.553) and (A.554) by Lemma 39 we have (A.532).

3. Recall that by Lemma 52, we have:

$$\hat{\mu}^*(I) \text{ is defined} \tag{A.555}$$

From (A.532), (A.550), (A.549), the fact that $I_0, \ldots, I_h$ is a corresponding sequence of I and condition 2 of this lemma from Lemma 56 we have:

$$\Omega^I = \Omega^{I^*} \tag{A.556}$$

and, in addition, from the fact that I is compatible and (A.555) we have (A.533). (A.534) follows immediately from (A.556) and Definition 37.

4. We will show that

$$a_u \text{ is ready in } I^* \tag{A.557}$$

We will use the result from Lemma 58.

We start from constructing W_{u-1} and then will argue that W_{u-1} and I^* satisfy the conditions from the lemma.

Let AR be the set of activity records of Π. Let $a_1, \ldots, a_m$ be the probabilistic leveling satisfying conditions 1-3 from Definition 21 for program $\Pi \setminus AR$. Let a_u be the first attribute in the sequence $a_1, \ldots, a_m$ such that:

$$a_u \text{ is not decided in } I \tag{A.558}$$

Let $a_1 = y_1, \ldots, a_{u-1} = y_{u-1}$ be the assignments to $a_1, \ldots, a_{u-1}$ in I. Let $T_0, \ldots, T_m$ be the sequence of trees for $\Pi \setminus AR$ described in the proof of Lemma 8. Let

$$S = \{a_i = y_i \mid 1 \leq i \leq u-1 \text{ and } y_i \neq u\}$$

Let W be a possible world s.t:

$$W \text{ is compatible with } I \tag{A.559}$$

Clearly,

$$S \subseteq W \tag{A.560}$$

Let W' be the set of atoms:

$$W' = W \setminus AR \cup \{truly\_random(a) \mid \exists p, y : \ random(a, p) \in W, do(a, y) \in \Pi\} \tag{A.561}$$

By Lemma 37, W' is a possible world of $\Pi \setminus AR$. By Lemma 19, T_m has a unique leaf node n s.t. W' satisfies $p_T(n)$. Let n_{u-1} be the ancestor of n, which is also a leaf of the tree T_{u-1}. Let $\Pi_0, \ldots, \Pi_m$ be the dynamic structure of $\Pi \setminus AR$ induced by $a_1, \ldots, a_m$. By Lemma 18 , there exists a unique possible world W_{u-1} of Π_{u-1} such that:

$$W_{u-1} \text{ satisfies } p_T(n_{u-1}) \tag{A.562}$$

We next show that W_{u-1} and I^* satisfies all the premises from Lemma 58. In (a) we show that I^* is compatible. In (b) we show I^* contains no e-literals formed by a_u. In (c) we prove that

$$\begin{aligned}&\text{each atom from } W_{u-1} \text{ not formed by } do, obs \\ &\text{and } truly\_random \text{ belongs to } I^*\end{aligned} \tag{A.563}$$

For each attribute term such that:

$$W_{u-1} \text{ has no atoms formed by } a, \tag{A.564}$$

$$a\text{'s level at most } u-1 \text{ in the total leveling induced by } a_1, \dots, a_m, \tag{A.565}$$

$$a \text{ is not formed by } do, obs \text{ and } truly\_random, \tag{A.566}$$

we have

$$a = u \subseteq I^*. \tag{A.567}$$

(a) In 2 we have shown $\Omega^I = \Omega^{I^*}$. Therefore, since W is compatible with I, we have:

$$I^* \text{ is compatible} \tag{A.568}$$

(b) From (A.549), (A.550) and (A.558) by Lemma 48 we have that I^* contains no e-literals formed by a_u.

(c) Let $J = \{\}$ if $h = 0$ and $I_{h-1} \cup b_h = v_h$ otherwise. Given that TU, AR, AR_{NOT} and L_{tr} are defined as in Def. 50 we have:

$$\begin{aligned}I^* &= f_n(J) \\ &= (f'_*((I_{h-1} \cup b_h = v_h) \setminus TU \setminus satr(AR)) \setminus L_{tr} \setminus AR_{NOT}) \\ &\quad \cup satr(AR) \cup TU\end{aligned} \tag{A.569}$$

Let V be a possible world of $\Pi \setminus AR$ such that:

$$\text{V is compatible with } \{I_{h-1} \cup b_h = v_h\} \setminus (satr(AR) \cup TU) \quad \text{(A.570)}$$

By Lemma 13, there exists a possible world W'_{u-1} of Π_{u-1} such that:

$$W'_{u-1} \subseteq V \quad \text{(A.571)}$$

and

$$(V \setminus W'_{u-1}) \cap L_{u-1} = \emptyset \quad \text{(A.572)}$$

We next show

$$W'_{u-1} \text{ satisfies } p_{T_{u-1}}(n_{u-1}) \quad \text{(A.573)}$$

Let now $c = x$ be an atom in $p_{T_{u-1}}(n_{u-1})$, different from $true$. From (A.562) we have:

$$c = x \in W_{u-1} \quad \text{(A.574)}$$

Therefore, from (A.571) we have:

$$c = x \in W' \quad \text{(A.575)}$$

Therefore, from (A.561):

$$c = x \in W \quad \text{(A.576)}$$

From (A.574), since W_{u-1} is a possible world of Π_{u-1}, and $W_{u-1} \in L_{u-1}$, and

$$c = x \in L_{u-1} \quad \text{(A.577)}$$

$$c \in \{a_1, \ldots, a_{u-1}\} \tag{A.578}$$

Therefore, since I decides all random attribute terms from $a_1, \ldots, a_{u-1}$ and W is compatible with I we have:

$$c = x \in I \tag{A.579}$$

Since $c = x$ is an atom in $p_{T_{u-1}}(n_{u-1})$, c is a random attribute term. Therefore, by Lemma 48 we have $c = x$ is an element of $ats(I)$. Therefore, since $I_{h-1} \cup b_h = v_h$ contains all elements of $ats(I)$, $c = x$ belongs to $\{I_{h-1} \cup b_h = v_h\} \setminus (satr(AR) \cup TU)$, and since V is compatible with $\{I_{h-1} \cup b_h = v_h\} \setminus (satr(AR) \cup TU)$, we have:

$$c = x \in V \tag{A.580}$$

From (A.571), (A.572), (A.580) and (A.577) we have:

$$c = x \in W'_{u-1} \tag{A.581}$$

Therefore, (A.573) holds. From (A.573) and (A.562) by Lemma 18 we have:

$$W'_{u-1} = W_{u-1} \tag{A.582}$$

From (A.582) and (A.571)

$$V \text{ satisfies } W_{u-1} \tag{A.583}$$

Therefore, since V was chosen arbitrarily from the possible worlds compatible with $\{I_{h-1} \cup b_h = v_h\} \setminus (satr(AR) \cup TU)$, by definition of

f'_*:

$$W_{u-1} \subseteq f'_*(\{I_{h-1} \cup b_h = v_h\} \setminus (satr(AR) \cup TU)) \tag{A.584}$$

Therefore, from (A.569) we have (A.563).

From (A.582) and (A.572) we have:

$$(V \setminus W_{u-1}) \cap L_{u-1} = \emptyset \tag{A.585}$$

Let now a be an attribute term satisfying (A.564) -(A.566). From (A.565) we have that every atom $a = y$ belongs to L_{u-1}. Since W_{u-1} has no atoms formed by a, from (A.585) we have that V has no atoms formed by a. Therefore, V satisfies *not* $a = y$. Hence, since V was chosen arbitrarily from the possible worlds compatible with $\{I_{h-1} \cup b_h = v_h\} \setminus (satr(AR) \cup TU)$, we must have *not* $a = y \in I^*$ by (A.569).

Therefore, by Lemma 58, a_u is ready in I^*.

5. Let Y be the set of possible values of a_u in I'_*. We will show that

$$\text{every interpretation in } \{I^* \cup \{a_u = y\} \mid y \in Y\} \text{ is compatible} \tag{A.586}$$

We will use the result from Proposition 7. We will first construct an auxiliary admissible function f_{aux} of Π and show that

$$\text{each element of } \{I^* \cup \{a_u = y\} \mid y \in Y\} \text{ is reachable via } f_{aux} \tag{A.587}$$

Since I is definite and compatible and reachable via an admissible consequence function f, by Proposition 7 we have:

$$I \text{ falsifies the body of every axiom in } \mathcal{X} \tag{A.588}$$

From 2 and Lemma 39 we have:

$$I \subseteq I^* \subseteq I^* \cup \{a_u = y\} \tag{A.589}$$

From (A.588) and (A.589) we have:

$$I^* \cup \{a_u = y\} \text{ falsifies the body of every axiom in } \mathcal{X} \tag{A.590}$$

From (A.590) and (A.587) by Proposition 7 we will conclude that (A.586). We will construct f_{aux} in (a) and show that (A.587) holds in (b).

(a) Let AU denote $satr(AR) \cup TU$. Let $f_{aux} : int(\Sigma) \rightsquigarrow int(\Sigma)$ be defined as follows:

$$f'_{aux}(E) = \begin{cases} f'_n(E), \text{ if } f'_n(E) \text{ is defined,} \\ \qquad\quad \text{and } E \notin \{(I^* \cup a_u = y) \setminus AU \mid y \in Y\} \\ E, \text{ if } E \in \{(I^* \cup a_u = y) \setminus AU \mid y \in Y\} \end{cases}$$

The fact that f'_{aux} is a consequence function of $\Pi \setminus AR$ follows immediately from (A.544). Let f_{aux} be the admissible consequence function determined by f'_{aux}.

(b) We prove (A.587).

We first that

$$\text{for any } y \in Y,\ f_{aux} \text{ is defined on } I^* \cup a_u = y \tag{A.591}$$

In 2, we have shown that $\Omega^I = \Omega^{I^*}$. Since I is compatible,

$$I^* \text{ is compatible} \tag{A.592}$$

We prove conditions 1-3 from Definition 50 in i-iii respectively. We will use L_{tr} and AR_{NOT} as defined there.

i. from (A.592) and Lemma 41, I^* cannot contain e-literals contrary to TU. That is, $I^* \cap L_{tr} = \emptyset$. Therefore, by minimality of saturation,

$$\{I^* \cup a_u = y\} \cap L_{tr} = \emptyset. \tag{A.593}$$

ii. Since I^* is in the image of an admissible consequence function f_n, $satr(AR) \subseteq I^* \subseteq \{I^* \cup a_u = y\}$. Therefore, since $I^* \cup a_u = y$ is an e-interpretation, it is consistent and

$$(I^* \cup a_u = y) \cap AR_{NOT} = \emptyset \tag{A.594}$$

iii. f'_{aux} is defined on $(I^* \cup a_u = y) \setminus TU \setminus satr(AR)$ by construction

Then, using the sets of e-literals AR_{NOT}, L_{tr} and TU from Definition 50 and denoting $AR_{NOT} \cup L_{TR}$ by AL and $TU \cup satr(AR)$ by TA, for every $y \in Y$ we have:

$$\begin{aligned} f_{aux}(I^* \cup a_u = y) &= (f'_{aux}((I^* \cup a_u = y) \setminus TA) \setminus AL) \cup TA \\ &= (I^* \cup a_u = y) \setminus TA \setminus AL \cup TA \end{aligned} \tag{A.595}$$

From (A.595) and (A.594) and (A.593) we have:

$$f_{aux}(I^* \cup a_u = y) = (f'_{aux}((I^* \cup a_u = y) \setminus TA) \cup TA$$
$$= (I^* \cup a_u = y) \tag{A.596}$$

Since f_{aux} agrees with f_n on any interpretation not deciding a_u, by Lemma 48 and the definition of a reachable sequence from (A.549) we have:

$$I_1, \ldots, I'_* \text{ is reachable via } f_{aux} \tag{A.597}$$

Since, by A.557, a_u is ready in I^*, by definition of reachable sequence from (A.596) we have that for every $y \in Y$:

$$I_1, \ldots, I^*, I^* \cup a_u = y \text{ is reachable via } f_{aux} \tag{A.598}$$

Therefore, (A.587) holds.

6. We will start by constructing a new consequence function f_{nn} of Π. As before, we start from describing a consequence function for $\Pi \setminus AR$, and then define f_{nn} in terms of it. Let Y be the set of possible values of a_u in I^*. Let $f'_{nn} : int(\Sigma) \rightsquigarrow int(\Sigma)$ be a partial function defined on every compatible interpretation as follows (we use the set TU as defined in Definition 50, and AU will denote $TU \cup satr(AR)$, and the consequence function f'_* described in 1):

$$f'_{nn}(E) = \begin{cases} f'_n(E), \text{ if } f'_n(E) \text{ is defined,} \\ \qquad\qquad \text{and } E \notin \{(I^* \cup a_u = y) \setminus AU \mid y \in Y\} \\ f'_*(E), \text{ otherwise} \end{cases}$$

The fact that f'_{nn} is a consequence function of $\Pi \setminus R$ follows immediately from the facts that both f'_n and $f'_*(E)$ are consequence functions of $\Pi \setminus R$. Let f_{nn} be the admissible consequence function of Π induced by f'_{nn}. For each $y \in Y$, we define:

$$I^*_y = f_{nn}(I_* \cup a_u = y) \tag{A.599}$$

We will show that

$$\text{for every } y \in Y \; I^*_y \text{ is informative} \tag{A.600}$$

Using the inductive hypothesis, it is sufficient to show: (a) I^*_y is reachable via f_{nn}, (b) $ind(I^*_y) = k$. (c) I^*_y is definite (d) I^*_y is compatible (e) for every random attribute term a decided in I^*_y $truly\_random(a)$ is decided. We will prove the claims in (a) - (e) below.

(a) We will prove:

$$I^*_y \text{ is reachable via } f_{nn} \tag{A.601}$$

Since f_{nn} agrees with f_n on any interpretation not deciding a_u, by Lemma 48 and the definition of a reachable sequence from (A.549) we have:

$$I_1, \ldots, I^* \text{ is reachable via } f_{nn} \tag{A.602}$$

$$\text{for any } y \in Y,\ f_{nn} \text{ is defined on } I^* \cup a_u = y \tag{A.603}$$

In 5 (b) we have already checked that conditions 1-2 of Definition 50 are satisfied for $I^* \cup a_u = y$. Therefore, it is sufficient to prove:

$$f'_{nn}((I^* \cup a_u = y) \setminus AU) \text{ is defined} \tag{A.604}$$

By 5, there exists a possible world U of Π such that

$$U \text{ is compatible with } I^* \cup \{a_u = y\} \tag{A.605}$$

By lemma 37, the set

$$U' = U \cup \{truly\_random(a) | \exists p, y : random(a,p) \in W, do(a,y) \in \Pi\} \\ \setminus AR \tag{A.606}$$

is a possible world of $\Pi \setminus AR$.

Since $I^* \cup a_u = y$ satisfies conditions 1-2 from Definition 50, the set $(I^* \cup a_u = y) \setminus AU$ contains no e-literals formed by *do*, *obs* and *truly_random*. Therefore, from (A.606) and (A.605) we have:

$$U' \text{ satisfies } (I^* \cup a_u = y) \setminus AU \tag{A.607}$$

Therefore, by construction of f'_{nn}, we have (A.604). Hence, (A.603) holds as well.

Since, by (A.557), a_u is ready in I^*, by definition of reachable sequence from (A.602) we have that for every $y \in Y$:

$$I_1, \ldots, I^*, I^*_y \text{ is reachable via } f_{nn} \tag{A.608}$$

Therefore, (A.601) holds.

(b) Let By Lemma 48,

$$\begin{aligned} ind(I_y^*) &= m - (h+1) \\ &= m - h - 1 \\ &= ind(I_h) - 1 \\ &= k + 1 - 1 \\ &= k \end{aligned}$$

(c) We have:

From (A.608) by Lemma 49 we have:

$$I_* \subseteq I_y^* \tag{A.609}$$

From (A.532) and (A.609) we have:

$$I \subseteq I_y^* \tag{A.610}$$

It is easy to see that for any e-interpretation M, if M satisfies (falsifies) an e-literal, then any superset of M satisfies (falsifies) the e-literal. Therefore, since I is definite, from (A.610) we have that I_y^* is definite.

(d) Similarly to (c), from $I \subseteq I_y^*$ and the fact that I falsifies the bodies of all axioms from $\mathcal{X}(\Pi)$, we have that I_y^* falsifies them too. Therefore, from (A.601), the fact that f_{nn} is admissible, and Proposition 7, I_y^* is compatible.

(e) By Lemma 48, $b_1, \ldots, b_h, a_u$ are the only random attributes decided in I_y^*. Since, by condition 2 of the lemma, for every $b \in \{b_1, \ldots, b_h\}$, I decides $truly\_random(b)$, and $I \subseteq I_y^*$ (see (c)), we have that I_y^* decides $truly\_random(b)$ for every $b \in \{b_1, \ldots, b_h\}$ as well. We now consider a_u. If $do(a_u, y) \in \Pi$, then, by definition of an admissible consequence function, I^*_y must contain $truly\_random(a_u) = u$. We now consider the case when Π does not contain actions for a_u. By construction of f_{nn}, it is sufficient show that $truly\_random(a_u)$ is decided in $f'_*((I^* \cup a_u = y) \setminus AU)$. There are two possibilities:

i. a_u is active in I^*. In this case $y \neq u$. Let V be a possible world of $\Pi \setminus AR$ s.t

$$\text{V is compatible with } (I^* \cup a_u = y) \setminus AU \tag{A.611}$$

We will show:

V is compatible with $satr(((I^* \cup a_u = y) \setminus AU) \cup truly\_random(a))$

Since a_u is active in I^*, there exists a random selection rule

$$random(a_u, p) \leftarrow B$$

such that I^* satisfies B. Since B does not contain e-literals formed by $do, obs, truly\_random$ we have that $(I^* \cup a_u = y) \setminus AU$ satisfies B. Therefore, by Proposition 4 from (A.611) we have:

$$\text{V satisfies B} \tag{A.612}$$

Therefore, by Proposition 1, V contains $random(a_u, p)$. Since $\Pi \setminus AR$ does not rules with heads formed by do, V does not contain atoms formed by do. Therefore, V satisfies the body of the axiom

$$\begin{aligned} truly\_random(a_u) \;\; \leftarrow \;\; & random(a_u, p), \\ & not\ do(a_u, y_1), \ldots, not\ do(a_u, y_k) \end{aligned} \tag{A.613}$$

Therefore, by Proposition 1, $truly\_random(a_u) \in V$. Therefore, since V was chosen arbitrarily from the possible worlds compatible with $(I^* \cup a_u = y) \setminus AU$, by definition of f'_*:

$$truly\_random(a_u) \in f'_*((I^* \cup a_u = y) \setminus AU) \quad \text{(A.614)}$$

ii. a_u is disabled in I^*. In this case $y = u$. Let V be a possible world of $\Pi \setminus AR$ s.t

$$\text{V is compatible with } (I^* \cup a_u = u) \setminus satr(AR) \setminus TU \quad \text{(A.615)}$$

We will show:

$$\begin{array}{l} V \text{ is compatible with} \\ satr(((I^* \cup a_u = y) \setminus AU) \cup truly\_random(a) = u) \end{array} \quad \text{(A.616)}$$

Since a_u is disabled in I^*, for every random selection rule of the form $random(a_u, p) \leftarrow B$, I^* falsifies B. Since the bodies of random selection rules do not contain e-literals formed by *do*, *obs*, *truly_random*, we have that $(I^* \cup a_u = y) \setminus satr(AR) \setminus TU$ does not satisfy any of them. Therefore, by Proposition 4 from (A.615) we have:

$$\text{V does not satisfy the body of every random selection rule for } a_u \quad \text{(A.617)}$$

Therefore, by minimality of possible worlds, for every p,

$$random(a_u, p) \notin V.$$

Therefore, V does not satisfy the body of the axiom

$$\begin{aligned} truly\_random(a_u) \ \leftarrow \ & random(a_u, p), \\ & not\ do(a_u, y_1), \ldots, not\ do(a_u, y_k). \end{aligned}$$

Therefore, by minimality of possible worlds,

$$truly\_random(a_u) \notin V.$$

Therefore, since V was chosen arbitrarily from the possible worlds compatible with $(I^* \cup a_u = y) \setminus AU$, by definition of f'_*:

$$truly\_random(a_u) = u \subseteq f'_*((I^* \cup a_u = y) \setminus AU) \qquad \text{(A.618)}$$

Therefore, in both cases, $truly\_random(a_u)$ is decided in I^*_y.

7. We prove (A.536). There are only two cases:

   (a) a_u is disabled in I^*. In this case $Y = \{u\}$. In (A.610) we have shown $I^* \subseteq I^*_u$. Therefore,

   $$\Omega^{I^*_u} \subseteq \Omega^{I^*} \qquad \text{(A.619)}$$

   We also have:

   $$\begin{aligned} \Omega^{I^*} & \subseteq \Omega^{I^* \cup a_u = u} && \text{(by Lemmas 36 and 33 (clause 2))} \\ & \subseteq \Omega^{I^*_u} && \text{(since } f_{nn} \text{ is a consequence function of } \Pi) \qquad \text{(A.620)} \end{aligned}$$

   From (A.619) and (A.620) we have:

   $$\Omega^{I^*_u} = \Omega^{I^*} \qquad \text{(A.621)}$$

Therefore, by Definition 37:

$$\hat{\mu}(I^*) = \hat{\mu}(I_u^*) \tag{A.622}$$

From (A.622) we have (A.536).

(b) a_u is active in I^*. In (A.610) we have shown that for every $y \in Y$, $I^* \subseteq I_y^*$. Therefore,

$$\bigcup_{y \in Y} \Omega^{I_y^*} \subseteq \Omega^{I^*} \tag{A.623}$$

On the other hand:

$$\begin{aligned} \Omega^{I^*} &\subseteq \bigcup_{y \in Y} \Omega^{I^* \cup \{a_u = y\}} && \text{(by Lemmas 36 and 33 (clause 1))} \\ &\subseteq \bigcup_{y \in Y} \Omega^{I_y^*} && \text{(since } f_{nn} \text{ is a consequence function of } \Pi) \end{aligned} \tag{A.624}$$

From (A.623) and (A.624) we have:

$$\bigcup_{y \in Y} \Omega^{I_y^*} = \Omega^{I^*} \tag{A.625}$$

Since no possible world can assign two different values to a_u, we have that for every $y_1, y_2 \in Y$:

$$\Omega^{I_{y_1}^*} \cap \Omega^{I_{y_2}^*} = \emptyset \tag{A.626}$$

We have:

$$\begin{aligned}
\sum_{y \in Y} \hat{\mu}(I_y^*) &= \sum_{y \in Y} \Bigg(\sum_{W \in \Omega^{I_u^y}} \hat{\mu}(W) \Bigg) && \text{(by Def. 37)} \\
&= \sum_{W \in \bigcup_{y \in Y} \Omega^{I_u^y}} \hat{\mu}(W) && \text{(by (A.626))} \\
&= \sum_{W \in \Omega^{I^*}} \hat{\mu}(W) && \text{(by (A.623))} \\
&= \hat{\mu}(I^*) && \text{(by Def. 37)}
\end{aligned}$$

Therefore, (A.536) holds.

8. We prove (A.537). By Lemma 48, every atom $a = y$ from I_y^* formed by a random attribute term belongs to $\{b_1 = v_1, \ldots, b_h = v_h, a_u = y\}$. Let y be a member of Y. Let A^* be the set of atoms in I^* formed by an attribute term from $\{b_1, \ldots, b_h\}$ s.t. for every $b = v \in A^*$, $P(I^*, b = v)$ is defined. Let A_y^* be the set of atoms in I_y^* formed by an attribute term from $\{b_1, \ldots, b_h\}$ s.t. for every $b = v \in A_y^*$, $P(I_y^*, b_1 = v_1)$ is defined. By Lemma 49, we have:

$$I^* \subseteq I_y^* \tag{A.627}$$

From A.532 and the fact that for each $b \in \{b_1, \ldots, b_h\}$ $truly\_random(b)$ is decided in I, we have:

$$\text{for each } b \in \{b_1, \ldots, b_h\} \; truly\_random(b) \text{ is decided in } I^* \tag{A.628}$$

From (A.628) and (A.627) by Lemma 54 we have:

$$\text{for each } b \in \{b_1, \ldots, b_h\} \; truly\_random(b) \in I^* \text{ iff } truly\_random(b) \in I_y^* \tag{A.629}$$

Therefore, by Lemma 50:

$$A^* = A^*_y \tag{A.630}$$

We now prove

$$\text{for every } b = v \text{ in } A^*, P(I^*, b = v) = P(I^*_y, b = v) \tag{A.631}$$

Let $b = v$ be an atom in A^*, and let i be the spot of a in $I_0, \ldots, I_{h-1}, I^*, I^*_y$. Since $b \neq a_u$, $i \leq h$. By Lemma 55, we have:

$$P(I^*, a) = P^*(I_{i-1}, a) \tag{A.632}$$

and

$$P(I^*_y, a) = P^*(I_{i-1}, a) \tag{A.633}$$

From the last two equations we have $P(I^*_y, a) = P(I^*, a)$. Therefore, (A.631) holds.

From (A.631) and (A.630) we have:

$$\hat{\mu}^*(I^*_y) = \begin{cases} \hat{\mu}^*(I^*) * P(I^*_y, a_u = y), \text{ if } P(I^*_y, a_u = y) \text{ is defined} \\ \hat{\mu}^*(I^*), \text{ otherwise} \end{cases} \tag{A.634}$$

Now we consider two cases:

(a) a_u is disabled in I^*. In this case, $Y = \{u\}$, no atoms formed by a_u belong to I^*_u, therefore, by (A.634) we have:

$$\hat{\mu}^*(I^*) = \hat{\mu}^*(I^*_u) \tag{A.635}$$

Therefore, (A.537) holds

(b) a_u is active in I^*. In this case we prove what

$$\text{for every } y \in Y,\ P(I^*_y, a_u = y) \text{ is defined} \tag{A.636}$$

Let y be a member of Y. Since a_u is active in I^*, there exists a random selection rule $random(a_u, p) \leftarrow B$ such that I^* satisfies $B \cup p(y)$ By Lemma 49, $I^* \subseteq I^*_y$, therefore,

$$I^*_y \text{ satisfies } B \cup p(y) \tag{A.637}$$

Now we show that

$$truly\_random(a_u) \in I^*_y \tag{A.638}$$

Let y' be an arbitrary value from $range(a_u)$. Consider an axiom ax_1 :

$$\leftarrow do(a_u = y'), not\ a_u = y'$$

Since a_u is active in I^*,

$$I^* \text{ does not contain } a_u = y' \tag{A.639}$$

Since I is a definite node which falsifies the bodies of all axioms in $\mathcal{X}(\Pi)$, by A.532 we have that

$$I^* \text{ also falsifies the bodies of all axioms in } \mathcal{X}(\Pi) \tag{A.640}$$

From (A.639) and (A.640) and the fact that $ax_1 \in \mathcal{X}(\Pi)$, we have:

$$I^* \text{ falsifies } do(a_u = y') \tag{A.641}$$

Since y' was chosen arbitrarily from the $range(a_u)$, we have:

$$\text{for every } y' \in range(a_u), I^* \text{ falsifies } do(a_u = y') \tag{A.642}$$

Let W be a possible world of Π compatible with I. In 2 we have shown $\Omega^I = \Omega^{I^*}$. Therefore, W is compatible with I^*. From (A.642) we have that for every $y' \in range(a_u)$, $do(a_u = y')$ does not belong to W. Therefore,

$$\Pi \text{ does not contain actions for } a_u \tag{A.643}$$

In 6 (e) i (see equation (A.614)) we have proved that, if a_u is active in I^*, then:

$$truly\_random(a_u) \in f'_*((I^* \cup a_u = y) \setminus AU) \tag{A.644}$$

Therefore, by construction of f_{nn} from (A.644) and (A.643) we have (A.638).

Since a_u is active in I^* via $random(a_u, p) \leftarrow B$, by conditions 2(c) and we have:

$$\begin{gathered}\text{for every pr-atom } pr(a_u = y' \mid B') = v \in \Pi, \\ B' \subseteq I^*, \text{ or } B' \text{ is falsified by } I^*\end{gathered} \tag{A.645}$$

$$\text{for every } y' \in range(a_u), \ p(y') \text{ is decided in } I^* \tag{A.646}$$

Therefore, by (A.532) we have:

$$\begin{gathered}\text{for every pr-atom } pr(a_u = y' \mid B') = v \in \Pi, \\ B' \subseteq I^*_y, \text{ or } B' \text{ is falsified by } I^*_y\end{gathered} \tag{A.647}$$

$$\text{for every } y' \in range(a_u), \ p(y') \text{ is decided in } I^*_y \tag{A.648}$$

From (A.637) , (A.638), (A.647), (A.648) we have that conditions (5.6)

- (5.9) are satisfies for I^*_y and $a_u = y$. Therefore, (A.636) holds. From (A.634) we have:

$$\hat{\mu}^*(I^*_y) = \hat{\mu}^*(I^*) * P(I^*_y, a_u = y) \tag{A.649}$$

In (6) we have shown that for each $y \in Y$, I^*_y is compatible. Therefore, for each $y \in Y$, there exists a possible world W_y compatible with I^*_y. By Lemma 59, all possible worlds compatible with I^* belong to a unique scenario s^* for r. Therefore,

$$\text{for every } y' \in Y, W_y \in s^* \tag{A.650}$$

Therefore, we have:

$$\begin{aligned}
\sum_{y \in Y} \hat{\mu}(I^*_y) &= \sum_{y \in Y} (\hat{\mu}(I^*) \cdot P(I^*_y, a_u = y)) && \text{(by (A.649))} \\
&= \hat{\mu}(I^*) \cdot \sum_{y \in Y} (P(I^*_y, a_u = y) \\
&= \hat{\mu}(I^*) \cdot \sum_{y \in Y} (P(W_y, a_u = y) && \text{(by Lemma 51)} \\
&= \hat{\mu}(I^*) \cdot \sum_{y \in Y} (P(W_{y'}, a_u = y) && \text{for some } y' \in Y \text{(by (A.650))} \\
&= \hat{\mu}(I^*) && \text{(since } \Pi \text{ is unitary)}
\end{aligned}$$

Therefore, (A.537) holds.

□

Proposition 8. Let $T_\Pi\langle f\rangle$ be an AI-tree of program Π from $\mathcal{B}$. Let I be an i-node of $T_\Pi\langle f\rangle$. If

1. I is compatible and definite, and

2. for every random attribute term a decided in I, $truly\_random(a)$ is decided in I

then I is informative (see definition 53).

□

□

Proof. Since every i-node of $T_\Pi\langle f\rangle$ is a reachable interpretation of Σ, the proposition follows immediately from Lemma 60. □

A.3.7 Proof of Proposition 9

Proposition 9 For every e-interpretation I of Σ, there exists a fixed point X of H such that

1. $I \subseteq X$,

2. no fixed point of H is a subset of X, and

3. no other fixed point of H satisfies conditions (a), (b).

We will refer to X satisfying conditions (a) - (c) as *the least fixed point of H relevant to I*.

Proof. We first prove that H is monotonic. We have: $H(L) = satr(L \cup \{head(r) \mid r \in nr(\Pi), body(r) \subseteq L\} \cup N(L))$ By definition of $satr$, $L \subseteq satr(L)$. By Lemma 39, $satr(L) \subseteq satr(L \cup \{head(r) \mid r \in nr(\Pi), body(r) \subseteq L\} \cup N(L))$. Therefore, $L \subseteq satr(L \cup \{head(r) \mid r \in nr(\Pi), body(r) \subseteq L\} \cup N(L) = H(L)$. Therefore, H is monotonic.

Consider now an operator H' which is defined on all sets of e-literals containing L, and $H'(I) = H(I)$ for every such set. Clearly, all interpretations containing L form a complete lattice with supremum L and infimum $elit(\Sigma)$. By Knaster-Tarski Theorem (see, for instance, Theorem A.2.1. from [Baral, 2003]), H' has the least fixed point F, which is a also the fixed point of H satisfying the conditions from the proposition.

□

A.3.8 Proof of Proposition 10

Proposition 10. For every e-interpretation I of Σ, there exists a fixed point X of G_I such that:

1. $I \subseteq X$,

2. no fixed point of G_I is smaller than X, and

3. no other fixed point of G_I satisfies conditions (a), (b).

We will refer to X satisfying conditions (a) - (c) as *the least fixed point of* G_I *relevant to* I.

□

Proof. We will first prove that G_I is monotonic. We have:

$$G_I(J) = satr(J \cup \{head(r) \mid r \in \Pi_{cons}(I), body^+(r) \subseteq J \text{ and } body^-(r) \text{ is not falsified by } I\})$$

By definition of $satr$, $J \subseteq satr(J)$. By Lemma 39,

$$satr(J) \subseteq satr(J \cup \{head(r) \mid r \in \Pi_{cons}(I), body^+(r) \subseteq J \text{ and } body^-(r) \text{ is not falsified by } I\}).$$

Therefore,

$$J \subseteq satr(J) \subseteq satr(J \cup \{head(r) \mid r \in \Pi_{cons}(I), body^{+}(r) \subseteq J$$
$$\text{and } body^{-}(r) \text{ is not falsified by } I\}) = G_I(J),$$

and G_I is monotonic.

Consider now an operator G'_I which is defined on all sets of e-literals containing J, and $G'_I(X) = G_I(X)$ for every such set X. Clearly, all interpretations containing J form a complete lattice with supremum J and infimum $elit(\Sigma)$. By Knaster-Tarski Theorem (see, for instance, Theorem A.2.1. from [Baral, 2003]), G'_I has the least fixed point F, which is a also the fixed point of H satisfying the conditions from the proposition.

□

A.3.9 Proof of Proposition 11 (Proof for f_1, f_2 and Sketch for f_3)

Lemma 61. Let Π be a P-log program not necessarily containing all the general axioms, and I be an e-interpretation of its signature.

$$\Omega^I_\Pi \subseteq \Omega_{\Pi \cup ENC(I)} \tag{A.651}$$

□

Proof. Let W be a member of Ω^I_Π. Since W is a possible world of Π, it satisfies the rules of Π. Since W is compatible with I, it satisfies the rules of $ENC(I)$. Therefore, W satisfies the rules of the reduct $(\Pi \cup ENC(I))^W$. For the sake of contradiction suppose there exists a proper subset W' of W which satisfies the rules in $(\Pi \cup ENC(I))^W$. Then W' satisfies the rules in Π^W, which is a contradiction to the fact that W is a possible world of Π.

□

We next introduce some notation. We first introduce some notation.

For a program Π (not necessarily containing all the general axioms) with signature Σ and e-interpretation I of Σ, by $drop(\Pi, I)$ we denote a program obtained from Π as follows:

1. removing every rule of the form

$$a = y \leftarrow B$$

   where $a = y \in I$ and B is non-empty

2. replacing every rule of the form

$$a = y \leftarrow B$$

   where $a = y_1 \subseteq I$ for some $y_1 \neq y$ with the constraint

$$\leftarrow B$$

Lemma 62. Let Π be a program (not necessarily containing all general axioms) containing a fact $a = y$. Let Π_2 be a program obtained from Π by removing a rule r of the form $a = y \leftarrow B$ with a non-empty body. We have

$$\Omega_{\Pi} = \Omega_{\Pi_2} \tag{A.652}$$

□

Proof. Let W be a possible world of Π. $\Pi_2^W = (\Pi \setminus r)^W$. So, clearly, W satisfies the rules of Π_2^W. For the sake of contradiction, suppose there is $W' \subsetneq W$ such that W' satisfies the rules of Π_2^W. Since Π_2^W contains a fact $a = y$, $a = y \in W'$. Therefore, W' also satisfies the rules of Π_2^W. Contradiction.

Suppose now W is a possible world of Π_2. $\Pi^W = (\Pi_2 \cup \{r\})^W$. Since Π_2 contains a fact $a = y$, $a = y \in W$. Therefore, W satisfies $\Pi^W = \Pi_2^W \cup \{r\}^W$. For the sake of contradiction, suppose there exists $W' \subsetneq W$ such that W' satisfies the rules of Π^W. Then, clearly, W' satisfies the rules of $\Pi_2^W \subseteq \Pi^W$, which is a contradiction to the fact that W is a possible world of Π_2^W.

□

Lemma 63. Let Π be a program (not necessarily containing all general axioms) containing a fact $a = y$. Let Π_2 be a program obtained from Π by replacing a rule r of the form $a = y_1 \leftarrow B$, where $y_1 \neq y_2$, which a constraint $\leftarrow B$.

$$\Omega_\Pi = \Omega_{\Pi_2} \tag{A.653}$$

□

Proof. We start from proving $\Omega_\Pi \subseteq \Omega_{\Pi_2}$. Let W be a possible world of Π. W contains $a = y$, therefore,

$$W \text{ does not satisfy } B \tag{A.654}$$

(or else it would not satisfy the rule $a = y_1 \leftarrow B$). Next, we have:

$$\Pi_2^W = (\Pi \setminus \{a = y_1 \leftarrow B\})^W \cup \{\leftarrow B\}^W \tag{A.655}$$

We prove:

$$W \text{ satisfies } \{\leftarrow B\}^W \tag{A.656}$$

If W does not satisfy one of the e-literals with default negation, (A.656) clearly holds. Otherwise, there rule $a = y_1 \leftarrow B'$, where B' is obtained from B by removing all e-literals with default negation, belongs to Π^W. Since W is a possible world of Π, it satisfies $a = y_1 \leftarrow B'$. Since $a = y \in W$, we have that W does not satisfy B'. Since $B' \subseteq B$, W does not satisfy B, and, therefore, (A.656) holds. Therefore, since W is a possible world of Π, from (A.656) and (A.655) we have:

$$W \text{ satisfies the rules of } \Pi_2^W \tag{A.657}$$

We now prove W is minimal such set. For the sake of contradiction, suppose there is $W' \subseteq W$ such that W' satisfies the rules of Π_2^W. We have

$$\Pi^W \subseteq \Pi_2^W \cup \{a = y_1 \leftarrow B\}^W \tag{A.658}$$

Since W' satisfies the rules of Π_2^W, it satisfies $\{\leftarrow B\}^W$. Therefore, W' satisfies $\{a = y_1 \leftarrow B\}^W$, and, by (A.658), Π^W. So, we have a contradiction to the fact that W is a possible world of Π.

We now prove $\Omega_{\Pi_2} \subseteq \Omega_\Pi$. Suppose now W is a possible world of Π_2. Since W satisfies the rules of Π_2^W, it satisfies $\{\leftarrow B\}^W$. Therefore, W satisfies $\{a = y_1 \leftarrow B\}^W$, and, by (A.658), Π^W, To conclude the proof, we need to show that W is a minimal set satisfying Π^W. Suppose there exists $W' \subsetneq W$ such that W' satisfies Π^W. As in the first part of the proof, we can use the relation

$$\Pi_2^{W'} = (\Pi \setminus \{a = y_1 \leftarrow B\})^{W'} \cup \{\leftarrow B\}^{W'} \tag{A.659}$$

and show that, since W' satisfies $\{a = y_1 \leftarrow B\}^{W'}$, it also satisfies $\{\leftarrow B\}^{W'}$, and, therefore, $\Pi_2^{W'}$, which contradicts the fact that W is a possible world of Π_2.

□

Lemma 64. Let Π be a program (not necessarily containing all general axioms) containing a constraint of the form $\leftarrow a = y_1$ for each $y_1 \in range(a)$. Let Π_2 be a program obtained from Π by replacing a rule r of the form $a = y \leftarrow B$ with a constraint $\leftarrow B$.

$$\Omega_\Pi = \Omega_{\Pi_2} \tag{A.660}$$

Proof. We start from proving $\Omega_\Pi \subseteq \Omega_{\Pi_2}$. Let W be a possible world of Π. W does not contain $a = y$ for any $y \in range(a)$, therefore,

$$W \text{ does not satisfy } B \tag{A.661}$$

(or else it would not satisfy the rule $a = y \leftarrow B$). The remaining reasoning is similar to the one from the proof of Lemma 63, only the explanation of why any of the possible worlds of both Π_2 and Π does not satisfy any head of the form $a = y$ is different. □

Lemma 65. Let Π be a program (not necessarily containing all general axioms) with signature Σ. Let I be an e-interpretation of Σ. We have:

$$\Omega_{drop(\Pi,I)\cup ENC(I)} = \Omega_{\Pi\cup ENC(I)} \tag{A.662}$$

□

Proof. The lemma immediately follows from the definition of *drop* and Lemmas 62 - 64. □

Lemma 66. Let Π be a P-log program (not necessarily containing all general axioms). Let C be a set of constraints in Π and Π_2 be a program obtained from Π by removing C. We have:

$$\Omega_{\Pi} \subseteq \Omega_{\Pi_2} \tag{A.663}$$

□

Proof. It is well known that the effect of adding a constraint to a program is to eliminate some of its stable models [Lifschitz, 2008]. The lemma then follows from Lemma 4 (it is easy to see that $\tau(\Pi)$ and $\tau(\Pi_2)$ only differ in constraints obtained by translating C). □

Lemma 67. Let Π be a P-log program (not necessarily containing all general axioms). Let C be a set of constraints in Π. Let Π_2 be the program obtained from Π by removing C. If every member of Ω_{Π_2} satisfies every constraint in C, then:

$$\Omega_{\Pi} = \Omega_{\Pi_2} \tag{A.664}$$

□

Proof. Suppose every member of Ω_{Π_2} satisfies every constraint in C. We will show $\Omega_\Pi = \Omega_{\Pi_2}$. In Lemma 66 we have shown that $\Omega_\Pi \subseteq \Omega_{\Pi_2}$. Here we will prove $\Omega_{\Pi_2} \subseteq \Omega_\Pi$. Let W be a possible world of Π_2. Since W satisfies C, it is easy to see that W satisfies $(\Pi_2 \cup C)^W$. Suppose now there exists $W' \subsetneq W$ satisfying $(\Pi_2 \cup C)^W$. Then W' would satisfy Π_2^W, which is a contradiction to the fact that W is a possible world of Π_2. □

Proposition 11. Let Π be a program from $\mathcal{B}$ with signature Σ. f_1, f_2 and f_3 are admissible consequence functions of Π. □

Proof. Let AR be the set of activity records in Π. Let $\Pi_0, \ldots, \Pi_n$ be the dynamic structure of $\Pi \setminus AR$ satisfying the condition from Definition 20. We prove the claim for functions $f_1 - f_3$ in 1-3 respectively.

1. We prove that f_1 is an admissible consequence function. By construction, it is sufficient to show that f'_1 is a consequence function of $\Pi \setminus AR$. Recall that for each I such that

$$\Pi_{cons}(I) \cup ENC(I) \text{ has a unique possible world } W \tag{A.665}$$

we have defined $f'_1(I)$ as follows:

$$f'_1(I) = I \cup satr(W \cup \bigcup_{a \in A} a = u)$$

where

$$A = \{a \mid a \in NRT(I) \text{ and } W \text{ does not contain atoms formed by } a\}.$$

In 1.1 we will prove that if $f'_1(I)$ is defined, then it is an e-interpretation of Σ. In 1.2 we will prove that $f'_1(\{\})$ is defined. In 1.3 we will show that for every $I \in int(\Sigma)$, if $f'_1(I)$ is defined, then it is a consequence of I. In 1.4 we will show that for each I s.t. $f(I)$ is defined, $f(I) \setminus I$ has no e-literals formed by random attribute terms.

1.1 Suppose $f'_1(I)$ is defined. In 1.1.1 we will prove that $I \cup satr(W \cup \bigcup_{a \in A} a = u)$ is consistent. In 1.1.2 we will show that $I \cup satr(W \cup \bigcup_{a \in A} a = u)$ is saturated.

1.1.1 By Lemma 43 and the minimality of saturation, $satr(W \cup \bigcup_{a \in A} a = u)$ is consistent. Therefore, it is sufficient to show that I contains no e-literals contrary to a member of $satr(W \cup \bigcup_{a \in A} a = u)$. For the sake of contradiction, suppose some $l \in I$ is contrary to an e-literal in $satr(W \cup \bigcup_{a \in A} a = u)$. We will consider all 4 possible forms of l:

(a) l is $a = y$. In this case $a = y \in ENC(I)$. Therefore, $a = y \in W \subseteq satr(W \cup \bigcup_{a \in A} a = u)$. Since $satr(W \cup \bigcup_{a \in A} a = u)$ is consistent, it cannot contain a literal contrary to l. Contradiction.

(b) l is *not* $a = y$. In this case $\leftarrow a = y \in ENC(I)$. Therefore, W does not satisfy $a = y$. Therefore, neither W nor $\bigcup_{a \in A} a = u$ contain $a = y$, and, by minimality of saturation, the only e-literal contrary to *not* $a = y$ does not belong to $satr(W \cup \bigcup_{a \in A} a = u)$.

(c) l is *not* $a \neq y$. We have $\leftarrow a \neq y \in ENC(I)$. Therefore, W does not satisfy $a \neq y$. Therefore, by Lemma 36,

$$a \neq y \notin satr(W) \tag{A.666}$$

By minimality of saturation:

$$a \neq y \notin satr(\bigcup_{a \in A} a = u) \tag{A.667}$$

By Lemma 42:

$$satr(W \cup \bigcup_{a \in A} a = u) = satr(\bigcup_{a \in A} a = u) \cup satr(W) \qquad \text{(A.668)}$$

From (A.639), (A.640) and (A.668) we have:

$$a \neq y \notin satr(W \cup \bigcup_{a \in A} a = u) \qquad \text{(A.669)}$$

Therefore, the only literal contrary to *not* $a \neq y$ does not belong to $satr(W \cup \bigcup_{a \in A} a = u)$.

(d) l is $a \neq y$. In this case the rules $def(a) \leftarrow a = Y$, $\leftarrow not\ def(a)$, $\leftarrow a = y$. Therefore, W contains an atom $a = y_1$ for $y_1 \neq y$. Therefore, $a \neq y \in satr(W \cup \bigcup_{a \in A} a = u)$, and, since $satr(W \cup \bigcup_{a \in A} a = u)$ is consistent, it doesn't contain e-literals contrary to l.

1.1.2 We will prove that $I \cup satr(W \cup \bigcup_{a \in A} a = u)$ is saturated. For a set of e-literals L and attribute term a, by L_a we will denote the subset of L of e-literals formed by attribute term a. For the sake of contradiction, suppose $I \cup satr(W \cup \bigcup_{a \in A} a = u)$ is not saturated. In this case there must exists a s.t. $(I \cup satr(W \cup \bigcup_{a \in A} a = u))_a$ is not saturated. Clearly, $a \notin A$, because the set $satr(a = u) \subseteq satr(W \cup \bigcup_{a \in A} a = u)$ is saturated, and any literal formed by a is contrary to one in $satr(a = u)$. Similarly, a is not from an atom in W, because $satr(W) \subseteq satr(W \cup \bigcup_{a \in A} a = u)$ is saturated, and any e-literal formed by a not belonging to $satr(W)$ is contrary to a member of $satr(W)$. Therefore, a an attribute term from an e-literal in I not occurring in $W \cup \bigcup_{a \in A} a = u)$. Then we cannot have $(I \cup satr(W \cup \bigcup_{a \in A} a = u))_a$ to be not saturated, because I is saturated, and, by minimality of saturation, $(I \cup satr(W \cup \bigcup_{a \in A} a = u))_a$ are precisely the e-literals in I formed by a.

1.2 We will prove that $f'_1(\{\})$ is defined. $\Pi'_{cons}(\{\}) \cup ENC(I)$ coincides with Π_0, which by Definition 20 has a unique possible world. Therefore, $f'(\{\})$ is defined.

1.3 We prove that that for every $I \in int(\Sigma)$, if $f'_1(I)$ is defined, then it is a consequence of I. By Lemma 39, $I \subseteq f'_1(I)$. Suppose V is a possible world of $\Pi \setminus R$ such that:

$$V \text{ is compatible with } I \tag{A.670}$$

We will prove that

$$V \text{ is compatible with } f'_1(I) \tag{A.671}$$

By Lemma 61, we have:

$$\Omega^I_{red(\Pi)} \subseteq \Omega_{red(\Pi) \cup ENC(I)} \tag{A.672}$$

By Lemma 11, we have:

$$\Omega_{red(\Pi \setminus AR)} = \Omega_{\Pi \setminus AR} \tag{A.673}$$

From (A.672) and (A.673) we have:

$$\Omega^I_{\Pi \setminus AR} \subseteq \Omega_{red(\Pi \setminus AR) \cup ENC(I)} \tag{A.674}$$

By Lemma 65 we have:

$$\Omega_{red(\Pi \setminus AR) \cup ENC(I)} = \Omega_{drop(red(\Pi \setminus AR), I) \cup ENC(I)} \tag{A.675}$$

Let C be the set of constraints in $drop(red(\Pi \setminus AR), I)$. By Lemma 66 we have:

$$\Omega_{red(\Pi \setminus AR) \cup ENC(I)} \subseteq \Omega_{(drop(red(\Pi \setminus AR), I) \setminus C) \cup ENC(I)} \tag{A.676}$$

From (A.674) - (A.676) we have:

$$\Omega^I_{\Pi \setminus AR} \subseteq \Omega_{(drop(red(\Pi \setminus AR), I) \setminus C) \cup ENC(I)} \tag{A.677}$$

Let A_I be the set of attribute terms decided in I, and let $L(I)$ be the set of literals of Σ formed by attribute terms in

$$NRT(I) \cup DRT(I) \cup A_I \setminus \{truly\_random(a) \mid do(a, y) \in \Pi \text{ for some } y\}$$

We will first prove that

$$L(I) \text{ is a splitting set of } drop(red(\Pi \setminus AR), I) \cup ENC(I) \tag{A.678}$$

Indeed, suppose $drop(red(\Pi \setminus AR), I) \cup ENC(I)$ contains a rule r of the form:

$$a = y \leftarrow B \tag{A.679}$$

such that

$$a = y \in L(I) \tag{A.680}$$

and l is a literal such that

$$l \text{ occurs in } B \tag{A.681}$$

To show (A.678), it is sufficient to show that

$$l \in L(I) \tag{A.682}$$

Let a_1 be the attribute term from l. By construction of $drop(red(\Pi \setminus AR), I) \cup ENC(I)$, we have

$$a \text{ depends on } a_1 \text{ in } red(\Pi) \tag{A.683}$$

We next show that

$$a \text{ is a non-random attribute term} \tag{A.684}$$

Indeed, for the sake of contradiction suppose a is a random attribute term. Since $a = y$ belongs to $L(I)$, by construction of $L(I)$, it must be decided in I. But, by construction of $drop(red(\Pi \setminus AR), I) \cup ENC(I)$, the program cannot contain rules with non-empty body with attribute a in the head such that a is interpreted by I. Therefore, (A.679) cannot belong to $drop(red(\Pi \setminus AR), I) \cup ENC(I)$. Contradiction. Therefore, (A.684) holds. By construction of $drop(red(\Pi \setminus AR), I) \cup ENC(I)$, since the body of r is non-empty, we have:

$$a \in NRT(I) \tag{A.685}$$

Then we have two possible remaining cases:

(a) a is a non-random attribute term and a_1 is a random attribute term. From (A.685) and (A.683) we have

$$a_1 \in DRT(I) \tag{A.686}$$

and, therefore, by definition of $L(I)$, (A.682) holds.

(b) Both a and a_1 are non-random attribute terms. For the sake of contradiction, suppose

$$a_1 \notin NRT(I) \tag{A.687}$$

This, by construction of $drop(red(\Pi \setminus AR), I)$, this means one of the two things:

i.

$$a_1 \text{ depends on } a_r \text{ in } red(\Pi \setminus AR) \tag{A.688}$$

for some random attribute a_r term such that

$$a_r \notin DRT(I) \tag{A.689}$$

From (A.683) and (A.688) we have:

$$a \text{ depends on } a_r \text{ in } red(\Pi \setminus AR) \tag{A.690}$$

From (A.690) and (A.689) we have:

$$a \notin NRT(I) \tag{A.691}$$

The last equation contradicts (A.685), therefore, (A.687) does not hold, $a_1 \in NRT(I)$, and, by definition of $L(I)$, (A.682) holds.

ii.

$$a_1 \text{ is of the form } random(a_2, p) \tag{A.692}$$

where

$$random(a_2, p) \notin NRT(I) \tag{A.693}$$

But in this case, since a_1 is non-random, we must have a of the form

$truly\_random(a_2)$, and thus, it by definition of $NRT(I)$, we cannot have $truly\_random(a_2) \in NRT(I)$. Contradiction to A.685.

Therefore, we have (A.678). From (A.678) by the definition of a splitting set we have:

$$L(I) \text{ is a splitting set of } (drop(red(\Pi \setminus AR), I) \setminus C) \cup ENC(I) \quad \text{(A.694)}$$

By Lemma 65 we have:

$$\Omega_{drop(\Pi_{cons}(I),I) \cup ENC(I)} = \Omega_{\Pi_{cons}(I) \cup ENC(I)} \quad \text{(A.695)}$$

We have:

$$drop(\Pi_{cons}(I), I) \cup ENC(I) =$$
$$bot_{L(I)}((drop(red(\Pi \setminus AR), I) \setminus C) \cup ENC(I)) \cup C_2 \quad \text{(A.696)}$$

where C_2 is the collection of constraints from $ENC(I)$ each of which contains an e-literal formed by an attribute term not belonging to $L(I)$.

We first show:

$$\text{all the constraints in } C_2 \text{ are of the form } \leftarrow l, \text{ where } l \text{ is a literal} \quad \text{(A.697)}$$

For the sake of contradiction, suppose there is a constraint in C_2 which is not of the form $\leftarrow l$. This means C_2 contains a constraint

$$\leftarrow not\ a = y_1, \ldots, not\ a = y_k$$

for some $a \notin L(I)$. However, $bot_{L(I)}((drop(red(\Pi \setminus AR), I) \setminus C) \cup ENC(I))$

has no rules with a in the head (because $a \notin L(I)$!), so we must have that

$$bot_{L(I)}((drop(red(\Pi \setminus AR), I) \setminus C) \cup ENC(I)) \text{ is inconsistent} \quad (A.698)$$

On the other hand, from (A.695), the fact that W is a possible world of $\Omega_{\Pi_{cons}(I) \cup ENC(I)}$, and (A.696) we get that

$$W \text{ is a possible world of } bot_{L(I)}((drop(red(\Pi \setminus AR), I) \setminus C) \cup ENC(I)) \quad (A.699)$$

(A.699) and (A.698) contradict each other, so we have (A.697). Suppose now C_2 has a constraint of the form *not* l, where the attribute term in l is not in $L(I)$. In this case, clearly, every possible world in $bot_{L(I)}((drop(red(\Pi \setminus AR), I) \setminus C) \cup ENC(I))$ satisfies every constraint in C_2, and, therefore, by Lemma 67:

$$\Omega_{bot_{L(I)}((drop(red(\Pi \setminus AR),I) \setminus C) \cup ENC(I))} = \Omega_{bot_{L(I)}((drop(red(\Pi \setminus AR),I) \setminus C) \cup ENC(I)) \cup C_2} \quad (A.700)$$

From (A.695), (A.696), (A.700) and (A.665) we have:

$$W \text{ is the unique possible world of } \Omega_{bot_{L(I)}((drop(red(\Pi \setminus AR),I) \setminus C) \cup ENC(I))} \quad (A.701)$$

From (A.670) and (A.677) we have:

$$V \text{ is a possible world of } (drop(red(\Pi \setminus AR), I) \setminus C) \cup ENC(I) \quad (A.702)$$

From (A.701) and (A.702) by splitting set theorem and Lemma 5 we have:

$$V \subseteq W \tag{A.703}$$

and

$$(V \setminus W) \cap L(I) = \emptyset \tag{A.704}$$

Since the set of literals from $NRT(I)$ is a subset of $L(I)$, we have:

$$(V \setminus W) \cap NRT(I) = \emptyset \tag{A.705}$$

Therefore, (A.671) holds.

1.4 We prove that for every $I \in int(\Sigma)$, if $f_1'(I) \setminus I$ does not contain e-literals formed by random attribute terms. We have:

$$f_1'(I) \setminus I \subseteq satr(W \cup \bigcup_{a \in A} a = u)$$

Since Π_{cons} does not have rules whose heads are formed by atoms formed by random attribute terms, and A does not have random attribute terms either, by minimality of saturation none of the e-literals in $satr(W \cup \bigcup_{a \in A} a = u)$ is formed by a random attribute term.

2. We now prove that f_2 is an admissible consequence function of Π. By construction, it is sufficient to show that f_2' is a consequence function of $\Pi \setminus AR$.

We start from proving a claim. Let I be an e-interpretation of Σ such that $least(I)$ is defined. We will prove:

$$\text{every possible world of } \Pi \text{ which satisfies } I \text{ also satisfies } least(I) \tag{A.706}$$

By Knaster-Tarski theorem (see, for instance, Theorem A.2.2 from [Baral, 2003],

$least(I)$ is equal to $\underbrace{H(\ldots H(}_{\text{finite composition}}I)\ldots)$ Therefore, it is sufficient to show:

$$\text{every possible world of } \Pi \text{ which satisfies } I \text{ also satisfies } H(I) \tag{A.707}$$

Let W be a possible world of Π such that:

$$W \text{ satisfies } I \tag{A.708}$$

$$H(I) = satr(I \cup \{head(r) \mid r \in nr(\Pi), body(r) \subseteq I\} \cup N(I)) \tag{A.709}$$

We first prove that W satisfies $\{head(r) \mid r \in nr(\Pi), body(r) \subseteq I\}$. Indeed, by Proposition 4, W satisfies $\{body(r) \mid r \in nr(\Pi), body(r) \subseteq I\}$. Therefore, by Proposition 1,

$$W \text{ satisfies } \{head(r) \mid r \in nr(\Pi), body(r) \subseteq I\} \tag{A.710}$$

We now prove W satisfies $N(I)$. Recall that $N(L)$ is the set of e-literals of the form *not* $a = y$, such that

- a is a non-random attribute term, and
- the body of every rule whose head is $a = y$ contains a literal contrary to some literal from L.

Let *not* $a = y$ be a member of $N(I)$. By Proposition 4, W falsifies the body of every rule with head $a = y$. Therefore,by minimality of possible worlds, W does not contain $a = y$, and satisfies *not* $a = y$. Since *not* $a = y$ was chosen arbitrarily from the $N(I)$, we have

$$W \text{ satisfies } N(I) \tag{A.711}$$

From (A.710) and (A.711) and (A.708) we have:

$$W \text{ satisfies } I \cup \{head(r) \mid r \in nr(\Pi), body(r) \subseteq I\} \cup N(I) \tag{A.712}$$

From (A.712) by Lemma 36 we have that

$$W \text{ satisfies } H(I) \tag{A.713}$$

Therefore, (A.707) and (A.706) hold.

We now prove that f_2' is, indeed, a consequence function of $\Pi \setminus R$. In 2.1 we will show that $f_2'(\{\})$ is defined. In 2.2 we will show that if $f_2'(\{I\})$ is defined, then it is a consequence of I w.r.t Π. In 2.3 we will show that if $f_2'(\{I\})$ is defined, then $f_2'(\{I\}) \setminus I$ does not contain random attribute terms.

2.1 We prove that $f_2'(\{\})$ is defined. In 1 we have shown that f_1' is a consequence function of $\Pi \setminus R$. Lemma 18 implies that $\Pi \setminus R$ is consistent. Let W be a possible world of Π. W satisfies $\{\}$. Since f_1' is a consequence function of $\Pi \setminus R$, W satisfies $f'(\{\})$. By (A.706), W satisfies $least(f'(\{\}))$. Therefore, since no interpretation satisfies inconsistent set of e-literals, W, $least(f'(\{\}))$ is consistent, and, by construction, $f_2'(\{\}) = least(f_1'(\{\}))$ is defined.

2.2 We show that if $f_2'(I)$ is defined, then it is a consequence of I w.r.t Π. We have:

$$f_2'(I) = least(f_1'(I))$$

By definition of *least*,

$$f_1'(I) \subseteq least(f_1'(I)) \tag{A.714}$$

Since f_1' is a consequence function,

$$I \subseteq f_1'(I) \tag{A.715}$$

From (A.714) and (A.715) we have:

$$I \subseteq f_2'(I) \tag{A.716}$$

The second condition follows immediately from (A.706).

2.3 We show that if $f_2'(I)$ is defined, then $f_2'(I) \setminus I$ does not contain e-literals formed by random attribute terms. By construction, the attribute terms in $\{head(r) \mid r \in nr(\Pi), body(r) \subseteq L\} \cup N(L)$ do not contain such literals. Therefore, by minimality of saturation, the subset of $f_2'(I)$ formed by random attribute terms coincide with the same subset of I, which implies $f_2'(I) \setminus I$ does not contain e-literals formed by random attribute terms.

3. The main argument for the third function will be based on the claim that *most* is a consequence function for $\Pi \setminus AR$. Function *most* is similar to function $AtMost$ from Stable Models [Simons et al., 2002] The correctness of the claim can be established using the properties of unfounded sets [Sacca & Zaniolo, 1990].

□

A.3.10 Proof of Proposition 12 (Sketch)

Proposition 12 Let Π be a program from $\mathcal{B}$, and T is one of the AI-trees in $\{T_\Pi\langle f_1\rangle, T_\Pi\langle f_2\rangle, T_\Pi\langle f_3\rangle\}$. For every query Q of Π, there exists a cut of T which is an efficient solution of Π w.r.t Q.

Proof. Let $a_1, \ldots, a_n$ be a causal order of Π. For each function $f \in \{f_1, f_2, f_3\}$, we will define a sequence of cuts of $T_\Pi\langle f\rangle$ as follows:

1. T_0 is a tree consisting of root $f(\{\})$

2. T_i is obtained from T_{i-1} by:

   - adding a child a_i to each leaf of T_{i-1},

- for each newly added node labeled with a_i, adding children $\{f(parent(a_i) \cup \{a_i = y\} \mid y$ is a possible value of a_i in $I\}$ where $parent(a_i)$ is the parent of a_i in $T_\Pi\langle f\rangle$.

It can be then shown that all the leafs of T_n decide all the attribute terms of Σ, so they are final w.r.t. Q. The fact that f on each of the corresponding inputs can be shown using the results from Lemma 58.

□