Epistemic Logic Programs: a Novel Perspective and some Extensions

Stefania Costantini[1] and Andrea Formisano[2]

[1] DISIM - Università dell'Aquila, Italy
[2] DMIF - Università di Udine, Italy

Abstract. Epistemic Logic Programs (ELPs) extend Answer Set Programming (ASP) with epistemic operators. The semantics of such programs is provided in terms of *world views*, which are sets of belief sets. Several semantic approaches have been proposed over time to characterize world views. Recent work has introduced semantic properties that should be met by any semantics for ELPs. We propose a new method, easy but, we believe, effective, to compare the different semantic approaches. We also propose some extensions to the ELP approach.

Keywords: Answer Set Programming, Epistemic Logic Programs, ELP semantics

1 Introduction

Epistemic Logic Programs (ELPs, in the following just 'programs' if not explicitly stated differently), were first introduced in [1,2], and extend Answer Set Programs (ASP programs), defined under the Answer Set Semantics of [3], with *epistemic operators* that are able to introspectively "look inside" a program's own semantics, which is defined in terms of its "answer sets". In fact, $\mathbf{K}A$ means that (ground) atom A is true in every answer set of the very program Π where $\mathbf{K}A$ occurs, whereas $\mathbf{M}A$ means that A is true in some of the answer sets of Π. The *epistemic negation operator* **not** A expresses that A *is not provably true*, meaning that A is false in at least one answer set of Π. It is easy to see that the operators are interchangeable, as $\mathbf{M}A$ can be defined as $not\,\mathbf{K}not\,A$, and **not** A as $not\,\mathbf{K}A$, not being standard ASP default negation.

Semantics of ELPs is provided in terms of *world views*: instead of a unique set of answer sets like in Answer Set Programming, there is now a set of such sets. Each world view consistently satisfies (according to a given semantics) the epistemic expressions that appear in a given program. Many semantic approaches for ELPs have been introduced beyond the seminal one of [1], among which we mention [4,5,6,7,8,9,10].

Recent work summarized in [11] has been aimed at extending to Epistemic Logic Programming some notions which have been previously defined for ASP, where many useful results have stemmed from them. So, according to [11,12,13], analogous properties might prove useful in ELPs as well. In particular, they consider *splitting* (introduced for ASP in [14]), which allows a program to be (iteratively) divided into parts ("top" and "bottom") in a principled way: the answer sets of a given program can be computed incrementally, starting from the answer sets of the bottom, which are used to simplify the top, and then the union of each answer set of the bottom with each answer set of the corresponding simplified top forms an answer set of the overall program. They extend to ELPs the concept of splitting and the method of incremental calculation of the semantics (here, it is the world views that must be calculated). This by defining a notion of *Epistemic Splitting*, where top and bottom are defined w.r.t. the occurrence of epistemic operators. Further, they adapt to ELPs other properties of ASP, which are implied by this property, namely the fact that adding constraints leads to reduce the number of answer sets, for ELPs, according to them, of the world views (*Subjective Constraint Monotonicity*), and *Foundedness*, meaning that atoms composing answer sets cannot have been derived through cyclic positive dependencies (where, for ELPs, they redefine positive dependencies so as to involve epistemic operators). In substance, this approach establishes properties that a semantics should fulfil, and then they compare the existing semantics with respect to these properties.

In this paper, we explore a different stance: in order to establish a term of comparison among the various semantics, we introduce a semantic approach which is very plainly based on the basic understanding of ELP and world views. We then experiment the new approach on many examples taken from the relevant literature, and we "observe" its behaviour, in terms of the correspondence or discrepancy with the results returned by other relevant semantic approaches. Then, we propose an extension to ELPs so as to allow for (positive) subjective literals in the head of rules. This extension gives a greater importance to meta-reasoning, and, we argue, this goes in favour of explainability and trustworthy Artificial Intelligence; technically, the extension rules out some unwanted aspects of many semantics, such as unfounded world views. The paper is organized as follows. In Sections 2 and 3 we recall ASP, and ELPs, for which we summarize the related semantic approaches. In Section 4 we introduce and discuss, via many examples, our proposal. In Section 5 we discuss the proposed extensions. Finally, in Section 6 we conclude.

2 Answer Set Programming and Answer Set Semantics

In ASP, one can see an ASP program as a set of statements that specify a problem, where each answer set represents a solution compatible with this specification. Whenever an ASP program has no answer sets (no solution can be found), it is said to be *inconsistent*, otherwise it is said to be *consistent*. Several well-developed freely available *answer set solvers* exist that compute the answer sets of a given program. Syntactically, an ASP program Π is a collection of *rules* of the form

$$A_1 \vee \ldots \vee A_g \leftarrow \; L_1, \ldots, L_n.$$

where each A_i, $0 \leq i \leq g$, is an atom, $\vee$ indicates disjunction (that can be alternatively indicated as $|$), and the L_is, $0 \leq i \leq n$, are literals (i.e., atoms or negated atoms of the form $not\, A$). The left-hand side and the right-hand side of the rule are called *head* and *body*, resp. A rule with empty body is called a *fact*. Disjunction can occur in rule heads only, so, in facts. A rule with empty head (or, equivalently, with head $\bot$), of the form '$\leftarrow L_1, ..., L_n.$' or '$\bot \leftarrow L_1, ..., L_n.$', is a *constraint*, stating that literals $L_1, \ldots, L_n$ are not allowed to be simultaneously true in any answer set; the impossibility to fulfil such requirement is one of the reasons that make a program inconsistent.

All extensions of ASP not explicitly mentioned above are not considered in this paper. We implicitly refer to the "ground" version of Π, which is obtained by replacing in all possible ways the variables occurring in Π with the constants occurring in Π itself, and is thus composed of ground atoms, i.e., atoms which contain no variables.

The answer set (or "stable model") semantics can be defined in several ways [15,16]. However, answer sets of a program Π, if any exists, are the supported minimal classical models of the program interpreted as a first-order theory in the obvious way. The original definition from [3], introduced for programs where rule heads were limited to be single atoms, was in terms of the 'GL-Operator' Given set of atoms I and program Π, $GL_\Pi(I)$ is defined as the least Herbrand model of the program Π^I, namely, the (so-called) Gelfond-Lifschitz reduct of Π w.r.t. I. Π^I is obtained from Π by: 1. removing all rules which contain a negative literal $\ not\, A\ $ such that $A \in I$; and 2. removing all negative literals from the remaining rules. The fact that Π^I is a positive program ensures that a least Herbrand model exists and can be computed via the standard immediate consequence operator [17]. Then, I is an answer set whenever $GL_\Pi(I) = I$.

3 Epistemic Logic Programs, syntax, semantics, and properties

Epistemic Logic Programs allow one to express within ASP programs so-called *subjective literals* (in addition to *objective literals*, that are those that can occur in plain ASP programs, plus the truth constants $\top$ and $\bot$). Such new literals are constructed via the *epistemic operator* **K** (disregarding without loss of generality the other epistemic operators). The literal $\mathbf{K}L$ means that (ground) the literal L is true in every answer set of given program Π (it is a *cautious consequence* of Π). The syntax of rules is analogous to ASP, save that literals in the body of rules now can be either objective or subjective. Nesting of epistemic operators is not considered here. An ELP program is called *objective* if no subjective literals occur therein, i.e., it is an ASP program. A constraint involving (also) subjective literals is called a *subjective constraint*, where one involving objective literals only is an *objective constraint*. Let At be the set of atoms occurring (within either objective or subjective literals) in a given program Π, and $Atoms(r)$ be the set of atoms occurring in rule r. By some abuse of notation, we denote by $Atoms(X)$ the set of atoms occurring in X, whatever X is (a rule, a program, an expression, etc.). Let $Head(r)$ be the head of rule r and $Body_{obj}(r)$ (resp., $Body_{subj}(r)$) be the (possibly empty) set of objective (resp., subjective) literals occurring in the body of r. For simplicity, we often write $Head(r)$ and $Body_{obj}(r)$ in place of $Atoms(Head(r))$ and $Atoms(Body_{obj}(r))$, respectively, when the intended meaning is clear from the context. We call *subjective rules* those rules whose body is made of subjective literals only.

The semantics of ELPs is based on the notion of *world views*: namely, sets of answer sets. Each world view determines the truth value of all objective literals in a program. For example, the program $\{a \leftarrow not\, b,\ b \leftarrow not\, a,\ e \leftarrow not\, \mathbf{K}f,\ f \leftarrow not\, \mathbf{K}e\}$, under every semantics, has two world views: $[\{a, e\}, \{b, e\}]$, where $\mathbf{K}e$ is true and $\mathbf{K}f$ is false, and $[\{a, f\}, \{b, f\}]$ where $\mathbf{K}f$ is true and $\mathbf{K}e$ is false. Note that, according to a widely-used convention, each world view, which is a set of answer sets, is enclosed in []. The presence of two answer sets in each world view is due to the cycle on objective atoms, whereas the presence of two world views is due to the cycle on subjective atoms (in general, the existence and the number of world views is related to such cycles, cf., [18] for a detailed discussion).

Let a semantics $\mathcal{S}$ be a function mapping each program into sets of 'belief views', i.e., sets of sets of objective literals, where $\mathcal{S}$ has the property that, if Π is an objective program, then the unique member of $\mathcal{S}(\Pi)$ is the set of stable models of Π. Given a program Π, each member of $\mathcal{S}(\Pi)$ is called a $\mathcal{S}$-*world view* of Π (we will often write "world view" in place of "$\mathcal{S}$-world view" whenever mentioning the specific semantics is irrelevant).

As usual, for any world view W and any subjective literal $\mathbf{K}L$, we write $W \models \mathbf{K}L$ iff for all $I \in W$ the literal L is satisfied by I (i.e., if $L \in I$ for L atom, or $A \notin I$ if L is $not\, A$). W satisfies a rule r if each $I \in W$ satisfies r.

The property of *Subjective Constraint Monotonicity* states that, for any ELP program Π and any subjective constraint r, W is a world view of $\Pi \cup \{r\}$ iff both W is a world view of Π and W satisfies r. Thus, if this property is fulfilled by a semantic $\mathcal{S}$, a constraint can rule out world views but cannot rule our some answer set from within a world view.

We report below some of the most relevant semantic definitions for ELPs. We start with the seminal definition of the first ELP semantics, introduced in [2], that we call for short G94. Let Π be an ELP program, and r a rule occurring therein.

Definition 1 (G94-world views). *The G94-reduct of Π with respect to a non-empty set of interpretations W is obtained by: (i) replacing by $\top$ every subjective literal $L \in Body_{subj}(r)$ such that L is of the form $\boldsymbol{K}G$ and $W \models L$, and (ii) replacing all other occurrences of subjective literals of the form $\boldsymbol{K}G$ by $\bot$. A non-empty set of interpretations W is a G94-world view of Π iff W coincides with the set of all stable models of the G94-reduct of Π with respect to W.*

This definition was then extended to a new one [4], that we call for short G11.

Definition 2 (G11-world views). *The G11-reduct of Π with respect to a non-empty set of interpretations W is obtained by: (i) replacing by $\bot$ every subjective literal $L \in Body_{subj}(r)$ such that $W \not\models L$, (ii) removing all other occurrences of subjective literals of the form $not\, \boldsymbol{K}L$. (iii) replacing all other occurrences of subjective literals of the form $\boldsymbol{K}L$ by L. The set W is a G11-world view of Π iff W it coincides with the set of all stable models of the G11-reduct of Π with respect to W.*

In [11], it is noticed that K15 [19], reported below, slightly generalizes the semantics proposed in [4].

Definition 3 (K15-world views). *The K15-reduct of Π with respect to a non-empty set of interpretations W is obtained by: (i) replacing by $\bot$ every subjective literal $L \in Body_{subj}(r)$ such that $W \not\models L$, and (ii) replacing all other occurrences of subjective literals of the form $\boldsymbol{K}L$ by L. The set W is a K15-world view of Π iff W it coincides the set of all stable models of the K15-reduct of Π with respect to W.*

Semantics G11 and K15, that are refinements of the original G94 semantics, have been proposed over time to cope with new examples that were discovered, on which existing semantic approaches produced unwanted or unintuitive world views.

K15 can be seen as a basis for the semantics proposed in [7] (called S16 for short). In particular, S16 treats K15 world views as candidate solutions, to be pruned in a second step, where some world views are removed, by applying the principle of keeping those which maximize what is not known. World views in [7] are obtained in particular as follows, where note however that they consider the operator **not**, that can be rephrased as $not\,\mathbf{K}A$ where not is ASP standard 'default negation' (meaning that A must be false in some answer set of a given world view).

Let $EP(\Pi)$ be the set of literals of the form **not** F occurring in given program Π.

Definition 4 (S16-world views). *Given $\Phi \subseteq EP(\Pi)$, the* Epistemic reduct Π^{Φ} *of Π w.r.t. Φ is obtained by: (i) replacing every **not** $F \in \Phi$ with $\top$, and (ii) replacing every **not** $F \notin \Phi$ with $not\,F$. Then, the set $\mathcal{A}$ of the answer sets of Π^{Φ} is a* candidate world view *if every **not** $F \in \Phi$ is true w.r.t. $\mathcal{A}$ (i.e., F is false in some answer set $J \in \mathcal{A}$) and every **not** $F \notin \Phi$ is false (i.e., F is true in every answer set $J \in \mathcal{A}$).*

We say that $\mathcal{A}$ is obtained from Φ (or it is corresponding to Φ, or that it is a candidate world view w.r.t. Φ), where Φ is called a candidate valid guess*. Then, $\mathcal{A}$ is an S16* world view *if it is maximal, i.e., if there exists no other candidate world view obtained from guess Φ' where $\Phi \subset \Phi'$ (so, Φ is called a* valid guess*).*

All the above semantics, in order to check whether a belief view $\mathcal{A}$ is indeed a world view, adopt some kind of reduct, reminiscent of that related to the stable model semantics, and $\mathcal{A}$ is a world view if it is *stable* w.r.t. this reduct. The F15 semantics [6,20] is based on very different principles, namely, it is based on a combination of Equilibrium Logic [21,22] with the modal logic S5. There, an EHT interpretation associates, via a function h, a belief view $\mathcal{A}$ with another belief view $\mathcal{A}'$ composed, for every set $A \in \mathcal{A}$, of sets $A' \subseteq A$. The purpose is to state that an implication is entailed, in any "belief point", i.e., in any interpretation $A \in \mathcal{A}$, by the couple $\langle \mathcal{A}, \mathcal{A}' \rangle$ if it is entailed either by $\mathcal{A}$ or by $\mathcal{A}'$. An EHT interpretation satisfies a theory in the usual way, and is total on a subset $\mathcal{X}$ of $\mathcal{A}$ if h gives back sets in $\mathcal{X}$ unchanged. A *total EHT model* can be an *equilibrium EHT model*, and is defined to be an **F15 world view**, if it is minimal according to two particular minimality conditions (not reported here).

Differently from F15, FAAEL [13] is based on the modal logic KD45. To define FAAEL, a belief view is transformed from a set of interpretations to a set of HT-interpretations, i.e., interpretations in terms of the logic of Here-and-There (HT) [23] which are couples $\langle H, T \rangle$ of 'plain' interpretations. A belief view is *total* if $H = T$ for all composing interpretations, thus reducing to the previous notion of belief view. A total version of any belief view can be formed, taking all the T's as components. A belief interpretation is now a belief view plus an HT interpretation, say $\hat{H}$, possibly not belonging to the belief view. The peculiarity of the entailment relation (defined in terms of HT logic) is in the implication, that must hold (in the usual way) in the belief interpretation, but also in the total version of the belief view therein. For total belief interpretations, the new relation collapses to the modal logic KD45. An epistemic interpretation is defined to be a belief model if all its composing HT interpretation as well as $\hat{H}$ entail all formulas of given theory. It is an epistemic model, if $\hat{H}$ is among the composing interpretations, and it is an *equilibrium belief model* if it satisfies certain minimality conditions. A belief view is a **FAAEL world view** if it is "extracted" from an equilibrium belief model $\mathcal{E}$ by taking all the T components of each $\langle H, T \rangle$ which is

program	world views
$a \vee b$	$[\{a\}, \{b\}]$
$a \vee b$ $a \leftarrow \mathbf{K}b$	$[\{a\}, \{b\}]$
$a \vee b$ $a \leftarrow not\, \mathbf{K}b$	$[\{a\}]$
$a \vee b$ $c \leftarrow not\, \mathbf{K}b$	$[\{a, c\}, \{b, c\}]$
$a \leftarrow not\, \mathbf{K}b$ $b \leftarrow not\, \mathbf{K}a$	$[\{a\}],\ [\{b\}]$
$a \leftarrow not\, \mathbf{K}not\, a$ $a \leftarrow not\, \mathbf{K}a$	$[\{a\}]$

program	G94/G11/FAEEL	K15/F15/S16
$a \leftarrow not\, \mathbf{K}not\, a$	$[\emptyset],\ [\{a\}]$	$[\{a\}]$
$a \vee b$ $a \leftarrow not\, \mathbf{K}not\, b$	none	$[\{a\}]$
$a \vee b$ $a \leftarrow \mathbf{K}not\, b$	$[\{a\}],\ [\{a\}, \{b\}]$	$[\{a\}, \{b\}]$
$a \leftarrow b$ $b \leftarrow not\, \mathbf{K}not\, a$	$[\emptyset],\ [\{a, b\}]$	$[\{a, b\}]$
$a \leftarrow not\, \mathbf{K}not\, b$ $b \leftarrow not\, \mathbf{K}not\, a$	$[\emptyset],\ [\{a\}, \{b\}]$	$[\{a\}, \{b\}]$

Fig. 1. On the left, examples where G94, G11, K15, F15, S16, and FAEEL agree. On the right, examples where G94/G11/FAEEL differ from K15/F15/S16. (Figure taken from [13].)

<table>
<tr><th>program</th><th>G94</th><th>G11/FAEEL</th><th>K15</th><th>F15/S16</th></tr>
<tr><td>$a \leftarrow not\, \mathbf{K}not\, b \wedge not\, b$
$b \leftarrow not\, \mathbf{K}not\, a \wedge not\, a$</td><td colspan="3">$[\emptyset], [\{a\}, \{b\}]$</td><td>$[\{a\}, \{b\}]$</td></tr>
<tr><td>$a \leftarrow \mathbf{K}a$</td><td>$[\emptyset],\ [\{a\}]$</td><td colspan="3">$[\emptyset]$</td></tr>
<tr><td>$a \leftarrow \mathbf{K}a$
$a \leftarrow not\, \mathbf{K}a$</td><td>$[\{a\}]$</td><td colspan="3">none</td></tr>
</table>

Fig. 2. Examples showing differences among several semantics. (Figure taken from [13].)

found in $\mathcal{E}$. For formal definitions of F15 and FAAEL, that for lack of space we cannot report here, we refer the reader to the aforementioned references.

FAAEL satisfies [12] Epistemic Splitting, Subjective Constraint Monotonicity, and Foundedness. G94 satisfies Epistemic Splitting, Subjective Constraint Monotonicity, but not Foundedness. In [13], it is proved that FAAEL world views coincide with *founded* G94 world views, where (roughly) founded world views are those where in every composing interpretation, objective atom G is never derived, directly or indirectly, from $\mathbf{K}G$.

We apologize with the readers and with the authors, because, for lack of space, we do not consider other recent semantics, such as, e.g., [9,24].

In Figures 1 and 2 a summary is reported, taken from [13], of how the semantics presented above behave on some examples which are considered to be significant of situations that can be found in practical programming.

4 Our Observations and Proposal

We propose here a method devised in order to compare the various semantics. We expose the new method, and we experiment it, taking as a base the examples in Figures 1 and 2, with few others.

Let us notice that, actually, in Gelfond's proposal, $\mathbf{K}G$ is intended to mean that G is true in all the answer set of given program, where the set of these answer sets is now called world view, or that G is true in all the answer sets of a certain world view, if there are many of them. It is not really required for G to be derivable from the program in a 'founded' way as it happens in ASP, or, at least, the concept of founded derivation becomes different. In the G94 computation of a world view, what is assumed to be known or not known comes from the world view, not from the program. What is required by this basic approach is that a world view is consistent w.r.t. the program, in the sense that what is assumed to be known is indeed concluded, and what is assumed to be false is not concluded. However, the point is that subjective atoms appearing in the program (and that are not derived, but elicited from the underlying world view) have a role in drawing conclusions.

We introduce an approach where this seminal intuition is literally applied. We then put the new approach at work on a number of examples, taking the occasion for a comparison with the semantics we have introduces above.

4.1 A new approach

We consider in this context only subjective literals $\mathbf{K}G$ and $\mathbf{K}notG$. We will consider them as new atoms, called *knowledge atoms*. Negation not in front of knowledge atoms is assumed to be the standard default negation. So, instead of ELPs proper, we here consider ASP programs possibly involving knowledge atoms. Let $SM(\Pi)$ be the set of answer sets of such a program Π.

First of all we introduce the concept of internal consistency of a set of atoms including knowledge atoms.

Definition 5. *A set A of atoms, composed of objective atoms and knowledge atoms, is said to be* knowledge consistent *iff:*

(i) it contains G whenever it contains the knowledge atom $\mathbf{K}G$;
(ii) it does not contain G whenever it contains the knowledge atom $\mathbf{K}notG$.

Let Π be a program. A set of sets of atoms $\mathcal{W}$, each such set composed of objective atoms and knowledge atoms (occurring in Π), is called here *epistemic interpretation*. For any atom G we write $\mathcal{W} \models G$ iff for all $X \in \mathcal{W}$ it holds that $G \in X$. Similarly, we write $\mathcal{W} \models not\, G$ iff for all $X \in \mathcal{W}$ it holds that $G \notin X$.

Definition 6. *Given ASP program Π possibly involving knowledge atoms, let $SMC(\Pi)$ be the set of those answer sets of the program which are knowledge-consistent.*

Property 1. $SMC(\Pi)$ correspond to the stable models of the program Π' obtained from Π by adding, for each knowledge atom $\mathbf{K}G$ or $\mathbf{K}notG$ occurring in Π, constraints:

$$\leftarrow \mathbf{K}G, not\, G$$
$$\leftarrow \mathbf{K}notG, G.$$

To establish a uniform comparison among semantic approaches, we propose a basic point of view on ELPs, that for convenience we present as a new semantics.

Definition 7. *[CF22-adaptation] The CF22-adaptation Π-$\mathcal{W}$ of a program Π with respect to an epistemic interpretation $\mathcal{W}$ is obtained by adding to Π:*

(i) new fact $\mathbf{K}G$ whenever $\mathcal{W} \models G$, and
(ii) new fact $\mathbf{K}notG$ whenever $\mathcal{W} \models not\, G$.

Let $F_{\Pi\text{-}\mathcal{W}}$ be the set of those newly added facts of the form $\mathbf{K}G$.

Definition 8 (CF22 world view). *An epistemic interpretation* $\mathcal{W}$ *is called a CF22 world view of a program* Π *if* $\mathcal{W} = SM'(\Pi\text{-}\mathcal{W})$*, where* $SM'(\Pi\text{-}\mathcal{W})$ *is obtained from* $SMC(\Pi\text{-}\mathcal{W})$ *by cancelling knowledge atoms.*

As seen, S16 semantics maximizes what is not known, which is equivalent to minimizing what is known. The proposers of S16 consider each potential world view (that in their approach is associated to a guess about what is not known) as a candidate world view, and discard those for which there exists another one with a larger guess on what is not known (equivalently, a smaller guess on what is known), in terms of set inclusion. Rephrasing their criterion (referred to as S16C) in terms of our approach, we have:

Definition 9 (S16 Criterion - CF22+S16C). *Each world view* $\mathcal{W}$ *as of Def. 8 is considered to be a* candidate world view. *A candidate world view* $\mathcal{W}$ *is indeed a world view under CF22+S16C if no other candidate world view* $\mathcal{W}$*' exists, where* $F_{\Pi\text{-}\mathcal{W}'} \subset F_{\Pi\text{-}\mathcal{W}}$.

4.2 CF22 world views: Examples of application

It can be easily seen that, on the examples on the left-column table of the above picture, on which all the above-presented semantic approaches agree, CF22 agrees as well. Below we present in detail a number of less trivial examples, some taken from the right and bottom tables of the above picture, and some from the relevant literature. The aim is to employ CF22 as a term of comparison among the various semantics.

Example 1 Consider the program Π

$a \vee b.$

$a \leftarrow \mathbf{K}b.$

$b \leftarrow \mathbf{K}a.$

Consider epistemic interpretation $\mathcal{W} = [\emptyset]$. According to Definition 7, the added facts are:

$\mathbf{K}not\, a.$ $\quad$ $\mathbf{K}not\, b.$

We have that $SMC(\Pi\text{-}\mathcal{W}) = \emptyset$, because the two rules cannot be applied, and the disjunction would generate answer sets $\{a\}$ and $\{b\}$ that are not knowledge consistent; thus, $SM'(\Pi\text{-}\mathcal{W}) = \emptyset$, so $\mathcal{W}$ **is not** a CF22 world view.

Consider epistemic interpretation $\mathcal{W} = [\{a\}]$ (the analogous can be done for $[\{b\}]$). According to Definition 7, the added facts are:

$\mathbf{K}a.$ $\quad$ $\mathbf{K}not\, b.$

We have the answer set $\{\mathbf{K}a, \mathbf{K}not\, b, a, b\}$ where a comes from the disjunction, and b is derived from the second rule, where however this answer set is not knowledge consistent; thus, $SMC(\Pi\text{-}\mathcal{W}) = SM'(\Pi\text{-}\mathcal{W}) = \emptyset$, so $\mathcal{W}$ **is not** a CF22 world view.

Consider epistemic interpretation $\mathcal{W} = [\{a, b\}]$. According to Definition 7, the added facts are:

$\mathbf{K}a.$ $\quad$ $\mathbf{K}b.$

We have that $SMC(\Pi\text{-}\mathcal{W}) = [\{\mathbf{K}a, \mathbf{K}b, a, b\}]$, with atoms a and b derived via the rules given the facts; this answer set is knowledge consistent, thus $SM'(\Pi\text{-}\mathcal{W}) = [\{a, b\}]$, so $\mathcal{W}$ **is** a CF22 world view.

Consider, finally, $\mathcal{W} = [\{a\}, \{b\}]$. According to Definition 7, there are no added facts. Then, $SMC(\Pi\text{-}\mathcal{W}) = SM'(\Pi\text{-}\mathcal{W}) = [\{a\}, \{b\}]$, deriving from the disjunction, as the two rules cannot be applied; thus, $\mathcal{W}$ **is** a CF22 world view.

This example shows that CF22, that here agrees with G11, does not satisfy foundedness. However, if one augments it with the S16C criterion (we called the combination CF22+S16C), then the unfounded world view $[\{a, b\}]$ is excluded, as there exists the world view $[\{a\}, \{b\}]$ which is based on fewer added positive knowledge literals (none for the latter and $\mathbf{K}a$ and $\mathbf{K}b$ for the former).

One may notice that, for world view $[\{a, b\}]$, these atoms are not derived from the program via a positive circularity: rather, they are supported, in the program, from what is deemed to be known in the world view itself. So, while this world view can be excluded by applying a minimality criterion, it is however not unreasonable in itself.

Example 2 Consider program Π:

$a \leftarrow not\, \mathbf{K} not\, a.$

Consider epistemic interpretation $\mathcal{W} = [\emptyset]$. According to Definition 7, the added facts are:

$\mathbf{K} not\, a.$

We have that $SMC(\Pi\text{-}\mathcal{W}) = [\{\mathbf{K} not\, a\}]$, thus $SM'(\Pi\text{-}\mathcal{W}) = [\emptyset]$, so $\mathcal{W}$ **is** a CF22 world view.

Consider epistemic interpretation $\mathcal{W} = [\{a\}]$. According to Definition 7, the added facts are:

$\mathbf{K}a.$

We have that $SMC(\Pi\text{-}\mathcal{W}) = [\{\mathbf{K}a, a\}]$ (as fact $\mathbf{K} not\, a$ is not present, its negation is true), thus $SM'(\Pi\text{-}\mathcal{W}) = [\{a\}]$, so $\mathcal{W}$ **is** a CF22 world view.

On this example, CF22 agrees with G94, G11, FAAEL.

Example 3 Let us now consider a more problematic example.

$a \leftarrow \mathbf{K}a.$

$a \leftarrow not\, \mathbf{K}a.$

Consider epistemic interpretation $\mathcal{W} = [\emptyset]$. According to Definition 7, the added facts are:

$\mathbf{K} not\, a.$

We have that $SMC(\Pi\text{-}\mathcal{W}) = \emptyset$ (as fact $\mathbf{K}a$ is not present, its negation is true, thus allowing to derive a, within however a stable model which is not knowledge consistent), thus $SM'(\Pi\text{-}\mathcal{W}) = \emptyset$, so $\mathcal{W}$ **is not** a CF22 world view.

Consider epistemic interpretation $\mathcal{W} = [\{a\}]$. According to Definition 7, the added facts are:

$\mathbf{K}a.$

We have that $SMC(\Pi\text{-}\mathcal{W}) = [\{\mathbf{K}a, a\}]$, thus $SM'(\Pi\text{-}\mathcal{W}) = [\{a\}]$, so $\mathcal{W}$ **is** a CF22 world view.

On this example, CF22 agrees with G94, where however all the other semantics provide no world view.

If the program would simply be $a \leftarrow \mathbf{K}a$. then its CF22 world views, as can be easily seen, would be $[\emptyset]$ and $[\{a\}]$, in agreement with G94, or with G11, K15, F15, S16, FAAEL under CF22+S16C.

Example 4 In previous examples, CF22+S16C tended to agree with S16. This is however not always the case.

$a \leftarrow not\, \mathbf{K} not\, b, not\, b.$

$b \leftarrow not\, \mathbf{K} not\, a, not\, a.$

Consider epistemic interpretation $\mathcal{W} = [\emptyset]$. According to Definition 7, the added facts are:

$\mathbf{K} not\, a. \quad \mathbf{K} not\, b.$

We have that $SMC(\Pi\text{-}\mathcal{W}) = [\mathbf{K} not\, a, \mathbf{K} not\, b]$, thus $SM'(\Pi\text{-}\mathcal{W}) = [\emptyset]$, so $\mathcal{W}$ **is** a CF22 world view.

Consider epistemic interpretation $\mathcal{W} = [\{a\}]$ (one can proceed analogously for $[\{b\}]$). According to Definition 7, the added facts are:

$\mathbf{K}a. \quad \mathbf{K} not\, b.$

We have that $SMC(\Pi\text{-}\mathcal{W}) = \emptyset$ (as one can derive b, obtaining however a stable model which is not knowledge consistent, because of fact $\mathbf{K} not\, b$), thus $SM'(\Pi\text{-}\mathcal{W}) = \emptyset$, so $\mathcal{W}$ **is not** a CF22 world view.

Consider epistemic interpretation $\mathcal{W} = [\{a\}, \{b\}]$, where there are no added facts.

We have that $SMC(\Pi\text{-}\mathcal{W}) = SM'(\Pi\text{-}\mathcal{W}) = [\{a\}, \{b\}]$, so $\mathcal{W}$ **is** a CF22 world view. Epistemic interpretation $[\{a, b\}]$ is easily discarded.

On this example, CF22 agrees with G94, G11, K15, FAAEL. Under CF22+S16C nothing changes, as both CF22 world views do not rely on positive knowledge atoms.

If the program is (seemingly) simpler, i.e.:

$a \leftarrow not\, \mathbf{K} not\, b.$

$b \leftarrow not\, \mathbf{K} not\, a.$

we have that, similarly to before, $\{a\}$ and $\{b\}$ are not CF22 world views. However, $SMC(\Pi\text{-}\mathcal{W}) = \emptyset$ now is a CF22 world view, because from added facts

$\mathbf{K} not\, a. \quad \mathbf{K} not\, b.$

one does not derive anything. Instead, $\mathcal{W} = [\{a\}, \{b\}]$ is not, because with no added facts one can derive both a and b, so $SMC(\Pi\text{-}\mathcal{W}) = SM'(\Pi\text{-}\mathcal{W}) = [\{a, b\}]$.

But, $\mathcal{W} = [\{a, b\}]$ is a CF22, world view, because adding new facts

$\mathbf{K}a. \quad \mathbf{K}b.$

both negations in the bodies of the program's two rules are true, so one derives both a and b obtaining $SMC(\Pi\text{-}\mathcal{W}) = SM'(\Pi\text{-}\mathcal{W}) = [\{a, b\}]$.

On this program, CF22 does not agree with the other semantics: it has it has world view $[\emptyset]$ like G94, G11, and FAEEL, but returns $[\{a, b\}]$, that no other semantics provides, and does not return $[\{a\}, \{b\}]$, that is provided by all the other semantics. The rationale underlying world view $[\{a, b\}]$ is that, again, it is consistent with given program, relatively to the positive knowledge atoms that the world view entails.

Example 5 Consider epistemic logic program:

$a \vee b.$

$a \leftarrow \mathbf{K} not\, b.$

Clearly, because of the disjunction $[\emptyset]$ cannot be a CF22 world view. Consider epistemic interpretation $\mathcal{W} = [\{a\}]$ According to Definition 7, the added facts are:

$\mathbf{K}a.$ $\mathbf{K}not\, b.$

We have that $SMC(\Pi\text{-}\mathcal{W}) = [\{\mathbf{K}a, \mathbf{K}not\, b, a\}]$, thus $SM'(\Pi\text{-}\mathcal{W}) = [\{a\}]$, so $\mathcal{W}$ **is** a CF22 world view.

Consider epistemic interpretation $\mathcal{W} = [\{b\}]$. According to Definition 7, the added facts are:

$\mathbf{K}b.$ $\mathbf{K}not\, a.$

We have that $SMC(\Pi\text{-}\mathcal{W}) = [\{\mathbf{K}b, \mathbf{K}not\, a, b\}]$, thus $SM'(\Pi\text{-}\mathcal{W}) = [\{b\}]$, so $\mathcal{W}$ **is** a CF22 world view.

Consider epistemic interpretation $\mathcal{W} = [\{a\}\{b\}]$. According to Definition 7, there are no added facts.

We have that $SMC(\Pi\text{-}\mathcal{W}) = SM'(\Pi\text{-}\mathcal{W}) = [\{a\}, \{b\}]$, so $\mathcal{W}$ **is** a CF22 world view.

It is easy to verify that instead $[\{a, b\}]$ is not a CF22 world view (because the disjunction cannot generate both a and b).

On this example, CF22 does not agree with existing semantics, because of the world view $[\{b\}]$, that they do not produce. Under CF22+S16C, there is agreement with S16, as in fact world view $[\{a\}, \{b\}]$, based upon an empty set of added knowledge atoms of the form $\mathbf{K}A$, rules out both $[\{a\}]$ and $[\{b\}]$.

Example 6 Consider epistemic logic program:

$a \vee b.$

$\leftarrow not\, \mathbf{K}a.$

Clearly, because of the disjunction $[\emptyset]$ cannot be a CF22 world view. Consider epistemic interpretation $\mathcal{W} = [\{a\}]$ According to Definition 7, the added facts are:

$\mathbf{K}a.$ $\mathbf{K}not\, b.$

We have that $SMC(\Pi\text{-}\mathcal{W}) = [\{\mathbf{K}a, \mathbf{K}not\, b, a\}]$ (the stable model with b is excluded as it is not knowledge consistent), thus $SM'(\Pi\text{-}\mathcal{W}) = [\{a\}]$, so $\mathcal{W}$ **is** a CF22 world view.

Consider epistemic interpretation $\mathcal{W} = [\{b\}]$. According to Definition 7, the added facts are:

$\mathbf{K}b.$ $\mathbf{K}not\, a.$

Here, the constraint is clearly violated, then we have $SMC(\Pi\text{-}\mathcal{W}) = SM'(\Pi\text{-}\mathcal{W}) = \emptyset$, thus $\mathcal{W}$ **is not** a CF22 world view.

Consider epistemic interpretation $\mathcal{W} = [\{a\}\{b\}]$. According to Definition 7, there are no added facts.

Again, the constraint is violated, then we have $SMC(\Pi\text{-}\mathcal{W}) = SM'(\Pi\text{-}\mathcal{W}) = \emptyset$, thus $\mathcal{W}$ **is not** a CF22 world view.

It is easy to verify that also $[\{a, b\}]$ is not a CF22 world view (because the constraint is respected, but the disjunction cannot generate both a and b).

Thus, CF22 on this program agrees with K15 and S16, and, like them, it does not satisfy *Subjective Constraint Monotonicity* as defined in [10] and subsequent papers. This property imposes that a constraint, in the above example

$\leftarrow not\, \mathbf{K}a.$

put at a higher level (in the sense of Lifschitz and Turner splitting notion, extended in the above-mentioned works to ELPs) w.r.t. an "object program" that in the above example is

$a \vee b.$

might have one of the following two effects: (i) the constraint is respected in a world view of the object (or "bottom"), program, thus such world view remains untouched; or, (ii) the constraint is violated in a world view, and in this case the world view is excluded. In particular, according to the FAAEL semantics, that satisfies Subjective Constraint Monotonicity, the above program has no world views, since the unique world view of the bottom part, i.e., $[\{a\}, \{b\}]$, is eliminated by the constraint.

However, it is not easy to understand this property, because in the "analogous" ASP program

$a \vee b.$

$\leftarrow not\, a.$

the constraint is indeed allowed, in ASP, to expunge from the (unique) world view $[\{a\}, \{b\}]$ of the bottom part (the set of its answer sets) the answer set b, thus producing for the program the unique world view $[\{a\}]$. This however, according to Subjective Constraint Monotonicity, should not be allowed for ELPs.

5 Extensions

In this section we introduce the possibility of having positive knowledge atoms as heads of rules in ELPs. This goes toward the wish, underlying part of the current literature, to derive what is known in a founded way from the program, instead of just requiring the program and its world views to be mutually consistent. Actually, the proposed extension allows for a mixture of the two attitudes.

The new syntax for ELPs is, synthetically, the following.

Definition 10. *The syntax of enhanced ELP programs (EELPs) is the same as for ELPs, except that the head of a rule can be a positive knowledge atom of the form* $\boldsymbol{K}G$.

Below is the definition of the enhanced program adaptation CF22M, where M stands for "Meta".

Definition 11. *[CF22M-adaptation] The CF22-adaptation* Π*-*$\mathcal{W}$ *of an EELP program* Π *with respect to an epistemic interpretation* $\mathcal{W}$ *is obtained by adding to* Π*:*

(i) new fact $\boldsymbol{K}G$ *whenever* $\mathcal{W} \models G$ *and* $\boldsymbol{K}G$ *does not occur as the head of a rule in* Π*, or*
(ii) new rule $G \leftarrow \boldsymbol{K}G$ *whenever* $\boldsymbol{K}G$ *occurs as the head of a rule in* Π*, or*
(iii) new fact $\boldsymbol{K} not G$ *whenever* $\mathcal{W} \models not\, G$*.*

Let $F_{\Pi\text{-}\mathcal{W}}$ *be the set of those newly added facts of the form* $\boldsymbol{K}G$*.*

Notice that the rule added in point (ii) corresponds to axiom T in modal logic S5, The definition of world view, now called CF22M world view, remains the same as in Definition 8, and can be extended as before to CF22M+S16C.

To see why the proposed extension is epistemically different from the original ELP approach, consider the following ELP program Π_1 (which refers to the Italian system, where in order to get promoted a positive evaluation of behaviour at school is required, in addition to having achieved good grades):

$promoted \leftarrow \mathbf{K}good\_grades, \mathbf{K}good\_behavior.$

A corresponding EELP program Π_2 is:

$\mathbf{K}promoted \leftarrow \mathbf{K}good\_grades, \mathbf{K}good\_behavior.$

$promoted \leftarrow \mathbf{K}promoted.$

where the latter rule is added by definition of CF22M-adaptation. Consider now to add to both programs the set of facts:

$good\_grades.$

$good\_behavior \vee bad\_behavior.$

Both programs have the same world views (where, on Π_1, all existing semantics, including CF22, clearly coincide), i.e.: $[\{promoted, good\_grades, good\_behavior\}]$ and $[\{good\_grades, bad\_behavior\}]$. So, it would seem that there is no advancement in evolving from CF22 to CF22M. Assume, however, to add a different set of facts, namely the single fact:

$promoted.$

In Π_1, as it is customary in ASP and more generally in logic programming, the fact overrides the rule, so the unique world view of the resulting program would be $[\{promoted\}]$. Considering now Π_2 under CF22M: this answer set is not knowledge consistent because $\mathbf{K}$*promoted* is not derived, so there exists no CF22M world view. This is to say, meta-level rules for an atom G, i.e., rules with head $\mathbf{K}G$, if existing, cannot be overridden by object-level ("bottom") rules. In the above example, it can be said that under CF22M *promoted* cannot be concluded because *there is no explanation/justification* for it, as the meta-level rule is not applicable. Notice that, it is left to the programmer to decide for which rules to introduce the head $\mathbf{K}G$, or instead to leave simply the head G. This accounts to deciding which atoms are more "critical", and so one wants a trustworthy derivation for them.

It is easy to prove the following theorem, that deals with the limit case where all atoms defined by rules are "managed" at the meta-level:

Theorem 1. *If, given ELP program Π, one constructs program Π' by substituting every atom G in the head of some rule with $\mathbf{K}G$, then the CF22M world views of Π' coincide with the founded CF22 world views of Π.*

We can see how this happens by means of an example.

Example 7 Consider program Π below.

$a \leftarrow not\, \mathbf{K}b.$

$b \leftarrow not\, \mathbf{K}a.$

$e \leftarrow \mathbf{K}f.$

$f \leftarrow \mathbf{K}e.$

As it is easy to see, CF22 world views are $[\{a, e, f\}]$ and $[\{b, e, f\}]$, both unfounded. Let us now consider Π':

$\mathbf{K}a \leftarrow not\, \mathbf{K}b.$
$\mathbf{K}b \leftarrow not\, \mathbf{K}a.$
$\mathbf{K}e \leftarrow \mathbf{K}f.$
$\mathbf{K}f \leftarrow \mathbf{K}e.$

Note that the CF22M-adaptation will add rule $A \leftarrow \mathbf{K}A$ for each $A \in \{a, b, e, f\}$. CF22M world views would thus be $[\{a\}]$ and $[\{b\}]$ because the last two rules of Π' form now a positive even cycle from which nothing is derived. We emphasize the difference: in Π', under CF22M, what is known is derived by the program; in Π, under CF22 and most of the other semantics, what is known is dictated by the world view, although it must be consistent with the program.

6 Conclusions

In this paper, we discussed Epistemic Logic Program. We have presented a semantic approach for ELPs, called CF22, which applies in a straightforward way the underlying principles of the seminal ELP approach as presented and discussed by Gelfond in [2]. We devised CF22 not exactly to propose "yet another semantics", but rather in order to establish a principled way of comparing the different semantic approaches. We have augmented CF22 to CF22+S16C by adding a minimality criterion, S16C, "inherited" by the semantics S16 [7], that excludes some world views if there are others that rely on fewer assumptions about what is known.

We have experimented CF22 on several examples taken from the relevant literature, for which the outcome of the other most relevant semantic approaches was well-known. Results are quite surprising, as the new semantics does not agree uniformly with the others, and in some cases it agrees with none of them. More investigation is required to understand the reasons for these discrepancies. Moreover, even when CF22 agrees with S16 (which is often the case), it is not always needed to apply the S16C Criterion in order to get the same world views.

Finally, we have taken CF22 as a basis for an extensions of the ELP paradigm, where ELPs are now allowed to include rules with positive knowledge atoms as the head. We have shown the power of this extension, that prevents conclusions to be drawn that are not epistemically justified. This formulation is able to force a founded derivation of "critical" atoms, dictated by the meta-level. In general terms, which knowledge atoms are to occur in rule heads is left to the knowledge engineer. If the approach is applied extensively, i.e., all rules have knowledge atoms as their head, this rules our unfounded world views, because what is known is in this case dictated by program rules.

References

1. Gelfond, M., Przymusinska, H.: Definitions in epistemic specifications. In Nerode, A., Marek, V.W., Subrahmanian, V.S., eds.: Proc. of the 1st Intl. Workshop on Logic Programming and Non-monotonic Reasoning, The MIT Press (1991) 245–259
2. Gelfond, M.: Logic programming and reasoning with incomplete information. Ann. Math. Artif. Intell. **12**(1-2) (1994) 89–116

3. Gelfond, M., Lifschitz, V.: The stable model semantics for logic programming. In Kowalski, R., Bowen, K., eds.: Proc. of the 5th Intl. Conf. and Symp. on Logic Programming, MIT Press (1988) 1070–1080
4. Gelfond, M.: New semantics for epistemic specifications. In Delgrande, J.P., Faber, W., eds.: Proc. of LPNMR'11. Volume 6645 of LNCS., Springer (2011) 260–265
5. Truszczynski, M.: Revisiting epistemic specifications. In Balduccini, M., Son, T.C., eds.: Logic Programming, Knowledge Representation, and Nonmonotonic Reasoning. Volume 6565 of Lecture Notes in Computer Science., Springer (2011) 315–333
6. Fariñas del Cerro, L., Herzig, A., Su, E.I.: Epistemic equilibrium logic. In Yang, Q., Wooldridge, M.J., eds.: Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, AAAI Press (2015) 2964–2970
7. Shen, Y., Eiter, T.: Evaluating epistemic negation in answer set programming. Artificial Intelligence **237** (2016) 115–135
8. Kahl, P.T., Leclerc, A.P.: Epistemic logic programs with world view constraints. In Palù, A.D., Tarau, P., Saeedloei, N., Fodor, P., eds.: Tech. Comm. of ICLP 2018. Volume 64 of OASIcs., Schloss Dagstuhl (2018) 1:1–1:17
9. Su, E.I.: Epistemic answer set programming. In Calimeri, F., Leone, N., Manna, M., eds.: Proc. of JELIA'19. Volume 11468 of LNCS., Springer (2019) 608–626
10. Cabalar, P., Fandinno, J., Fariñas del Cerro, L.: Splitting epistemic logic programs. In Balduccini, M., Lierler, Y., Woltran, S., eds.: Proc. of LPNMR'19. Volume 11481 of Lecture Notes in Computer Science., Springer (2019) 120–133
11. Cabalar, P., Fandinno, J., Fariñas del Cerro, L.: Splitting epistemic logic programs. Theory Pract. Log. Program. **21**(3) (2021) 296–316
12. Cabalar, P., Fandinno, J., Fariñas del Cerro, L.: On the splitting property for epistemic logic programs (extended abstract). In Bessiere, C., ed.: Proceedings of IJCAI 2020, ijcai.org (2020) 4721–4725
13. Cabalar, P., Fandinno, J., Fariñas del Cerro, L.: Autoepistemic answer set programming. Artif. Intell. **289** (2020) 103382
14. Lifschitz, V., Turner, H.: Splitting a logic program. In: Proc. of ICLP'94, MIT Press (1994) 23–37
15. Lifschitz, V.: Thirteen definitions of a stable model. In Blass, A., Dershowitz, N., Reisig, W., eds.: Fields of Logic and Computation. Volume 6300 of Lecture Notes in Computer Science., Springer (2010) 488–503
16. Costantini, S., Formisano, A.: Negation as a resource: a novel view on answer set semantics. Fundamenta Informaticae **140**(3-4) (2015) 279–305
17. Lloyd, J.W.: Foundations of Logic Programming. Springer-Verlag (1987)
18. Costantini, S.: About epistemic negation and world views in epistemic logic programs. Theory Pract. Log. Program. **19**(5-6) (2019) 790–807
19. Kahl, P., Watson, R., Balai, E., Gelfond, M., Zhang, Y.: The language of epistemic specifications (refined) including a prototype solver. J. Log. Comput. **30**(4) (2015) 953–989
20. Su, E.I., Fariñas del Cerro, L., Herzig, A.: Autoepistemic equilibrium logic and epistemic specifications. Artif. Intell. **282** (2020) 103249
21. Pearce, D.: A new logical characterization of stable models and answer sets. In: Non-Monotonic Extensions of Logic Programming. Number 1216 in LNAI. Springer (1997) 55–70
22. Pearce, D., Valverde, A.: Synonymous theories in answer set programming and equilibrium logic. Proc. of ECAI04, 16th Europ. Conf. on Art. Intell. (2004) 388–390
23. Pearce, D.: Equilibrium logic. Ann. Math. Artif. Intell. **47**(1-2) (2006) 3–41
24. Su, E.I.: Refining the semantics of epistemic specifications. In et al., A.F., ed.: Proceedings 37th International Conference on Logic Programming, ICLP Technical Communications 2021. Volume 345 of EPTCS. (2021) 113–126