Fundamenta Informaticae XX (2015) 1–15
DOI 10.3233/FI-2012-0000
IOS Press

Argumentation semantics and belief change within logic programs

Ján Šefránek,
Comenius University. Bratislava, Slovakia,
sefranek@ii.fmph.uniba.sk

Abstract. Belief change for ordered pairs of extended logic programs is studied in this paper. Several belief change operators are introduced. They map sets of belief sets of given logic programs to resulting sets of belief sets. A belief set is defined as the set of all consequences of a set of default literals. The work is focused on sound (undefeated) sets of default literals, which satisfy conditions of an argumentation semantics. An inertia of a current state and an identification of irrelevant updates are among most interesting properties of the approach.[1]

1. Introduction

This paper is connected to the research tradition of logic program updates, represented, e.g., by [1, 3, 11, 15, 14, 17]. The basic feature of those approaches is a manipulation on program rules, e.g., a rejection of less preferred older rules in the case of conflicts.

We will use a more general notion, belief change, instead of update. At least sometimes, it is not clear, whether adding a new program to an original program satisfies intuitions connected to an update or to a revision, i.e. whether they reflect a new state of the world or a new state of an agent's knowledge.

In order to overcome the purely syntactic nature of the approaches cited above, an approach based on SE-models was developed [6]. However, it was shown in [19] (Theorem 20) that an update semantics based on SE-models does not satisfy quite natural conditions on updates. The proof is based on programs that are indistinguishable from the viewpoint of SE-models. An impact of this result is more general – programs that are indistinguishable from the viewpoint of SE-models may manifest unexpected behavior w.r.t. belief change:

Example 1.1. ([19])
Consider strongly equivalent programs $P_1 = \{p \leftarrow; q \leftarrow\}$, $P_2 = \{q \leftarrow; p \leftarrow q\}$.[2]

[1] This research was supported by grant VEGA 1/1333/12.
[2] Let $I \subseteq J \subseteq \mathcal{A}$. (I, J) is an SE-model of a program P iff $J \models P$ and $I \models P^J$. Two programs are strongly equivalent iff they have identical sets of all SE-models. The set of all SE-models of P_1 and also of P_2 is $\{(\{p, q\}, \{p, q\})\}$.

Let a new program $Q = \{\neg q \leftarrow\}$ be given, which adds new information to P_1 and P_2, respectively. Consider now pairs $\langle P_1, Q\rangle$ and $\langle P_2, Q\rangle$. Because P_1 and P_2 are indistinguishable w.r.t. SE-models, each belief change operator, defined on SE-models should give the same result for both pairs of programs. On the other hand, intuitively, both pairs should lead to different resulting meanings (beliefs).

Thus, we return to belief change characterizations, using rejection of rules. However, a kind of semantic flavour is added. First, our focus is oriented towards a dependency of literals on sets of default negations as in [4, 7, 9]. Second, argumentation-theoretic semantics of logic programs [9] are extended also to belief change within logic programs. Characterization of changed beliefs in terms of different argumentation semantics can be given. We will show on the example of an inertia of current information (Section 7) that an argumentation-semantic point of view enables a more subtle analysis of a problem. However, this paper represents only a first step in developing argumentation-theoretic semantics of belief change within logic programs. A future investigation of properties and applications of different argumentation semantics is planned.

An important feature of our approach is that we are developing a non-prioritized belief change. More precisely, it is a kind of limited prioritized belief change – usually new information is preferred, unless it is based on assumptions falsified, in a sense, by the original information.

Non-prioritized belief change in the frame of monotonic logic is determined by such aspects as epistemic value or credibility [12, 16]. It can be said that it is determined by notions on a level of pragmatics. On the other hand, our approach to non-prioritized belief change is based on a level of syntax. Defeasible / falsifiable assumptions (default negations, in the case of logic programs) are inherent in languages of non-monotonic formalisms. Even if a lower epistemic value or credibility may be ascribed to assumptions, assumptions are in non-monotonic formalisms usually parts of expressions, on which no credibility-based hierarchy is determined.

The rest of the paper is organized as follows Terminology and notations for logic programs are introduced. After that, in Section 3, three basic belief change operators are defined. Important notions of defeated and sound sets of assumptions are introduced in Section 4. A short survey of argumentation semantics of logic programs is given in Section 5. Belief change operators satisfying argumentation-semantic conditions are introduced in Section 6. Finally, inertia of a current state and irrelevant belief change are studied.

2. Logic Programs

We will consider propositional extended logic programs in this paper.

Let $\mathcal{A}$ be a set of atoms. The set of objective literals is $\mathcal{O} = \mathcal{A} \cup \{\neg a \mid a \in \mathcal{A}\}$. A convention as follows is accepted: Let a literal l be $\neg a$, where $a \in \mathcal{A}$. Then $\neg l$ is a.

If $B \subseteq \mathcal{O}$, then $not\ B = \{not\ l \mid l \in B\}$. The set of default literals is defined as $\mathcal{D} = not\ \mathcal{O}$. A literal is an objective literal or a default literal. The set of all literals $\mathcal{O} \cup \mathcal{D}$ is denoted by $\mathcal{L}$. A rule r is an expression of the form

$$l \leftarrow l_1, \ldots, l_k, not\ l_{k+1}, \ldots, not\ l_{k+m};\ \text{where } k \geq 0, m \geq 0, l, l_i \in \mathcal{O}$$

l is called the head of the rule and denoted by $head(r)$. The set of literals $\{l_1, \ldots, l_k, not\ l_{k+1}, \ldots, not\ l_{k+m}\}$ is called the body of r, denoted by $body(r)$.

An extended logic program is a finite set of rules. We will often use only the term program. If $\mathcal{A}$ is the set of all atoms used in a program P, it is said that P is over $\mathcal{A}$.

A default literal is called also an *assumption* and $\Delta \subseteq \mathcal{D}$ are called also *sets of assumptions*.

A *proof* of a literal $l \in \mathcal{L}$ from a set of literals T in a program P is a minimal[3] sequence $\langle l_1, \ldots, l_n \rangle$, $n \geq 1$, s.t. for each j, $1 \leq j \leq n$,

- $l_j \in T$, or
- $l_j = head(r)$, where $r \in P$ and $body(r) \subseteq \{l_1, \ldots, l_{j-1}\}$.[4]

We will denote by $T \vdash_P l$ that there is a proof of l from T in P. The symbol $T^{\leadsto_P}$ is used as a shorthand for $\{l \in \mathcal{L} \mid T \vdash_P l\}$. Proofs from sets of assumptions $\Delta \subseteq \mathcal{D}$ are considered usually in this paper.

3. Basic belief change operators

Let P be a program and Δ be a set of assumptions. Then P is called a *belief base*, $\Delta^{\leadsto_P}$ is called a *belief set* of P, if $\Delta \neq \Delta^{\leadsto_P}$.[5] A belief is a member of a belief set. The set of all belief sets of a program P is denoted as $\mathcal{B}_P$.

Let an ordered pair of logic programs $\langle P_1, P_2 \rangle$ be given. P_1 represents older information and P_2 contains new information.

We will study problems of belief change in $\langle P_1, P_2 \rangle$.

Definition 3.1. (Belief change operator)
Let a sequence of programs $\langle P_1, P_2 \rangle$ be given. A *belief change operator* $\mathcal{C}$ assigns a set of belief sets $\mathcal{B}_{\langle P_1, P_2 \rangle}$, where $\mathcal{B}_{\langle P_1, P_2 \rangle} \subseteq \mathcal{B}_{(P_1 \cup P_2) \setminus R}$ for some $R \subseteq P_1 \cup P_2$, to sets of belief sets $\mathcal{B}_{P_1}$ and $\mathcal{B}_{P_2}$ of all belief sets of programs P_1 and P_2, respectively. Formally, $\mathcal{C}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) = \mathcal{B}_{\langle P_1, P_2 \rangle}$. $\Box$

We are going to introduce some specific belief change operators, which specify different understandings of $\mathcal{B}_{\langle P_1, P_2 \rangle}$. First, an operator $\mathcal{C}^{exp}$.

Definition 3.2. ($\mathcal{C}^{exp}$ operator)
$P_1 \cup P_2$ is called the *expansion* of P_1 by P_2. $\mathcal{C}^{exp}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) = \mathcal{B}_{P_1 \cup P_2}$. $\Box$

Definition 3.3. Let be $\Delta \subseteq D$. It is said that Δ *generates a conflict* C[6] w.r.t. a set of rules R iff for some $l \in \mathcal{O}$

- $C = \{not\ l, l\}$ and $not\ l \in \Delta$, $\Delta \vdash_R l$, or

[3] Redundant steps in proofs are eliminated.
[4] If $j = 1$ then $\{l_1, \ldots, l_{j-1}\}$ is understood as $\emptyset$.
[5] A comment to this design decision: Nothing can be believed, if nothing can be derived from a set of assumptions, except of those assumptions. At least one negative consequence of this design decision is, that $\emptyset$, satisfying conditions of an argumentation semantics, but without consequences w.r.t. a set of rules is not considered.
[6] We are interested only in conflicts generated by sets of assumptions. This is a consequence of our definition of a belief set. Consider, e.g., the ordered pair $P_1 = \{a \leftarrow b\}; P_2 = \{\neg a \leftarrow b\}$. There is a kind of conflict between both programs, however, nothing can be derived from them, nothing can be believed, hence, no change of beliefs can be registered. Of course, other design decisions are possible.

- $C = \{l, \neg l\}$ and $\Delta \vdash_R l$, $\Delta \vdash_R \neg l$.

A *solution* of a conflict C generated by a set of assumptions Δ w.r.t. an expansion $P_1 \cup P_2$ is a minimal set of rules R s.t. Δ does not generate C w.r.t. $(P_1 \cup P_2) \setminus R$.

A solution of *all* conflicts generated by Δ w.r.t. $P_1 \cup P_2$ is a minimal set of rules R s.t. Δ generates no conflict w.r.t. $(P_1 \cup P_2) \setminus R$. $\Box$

Definition 3.4. A set of assumptions $\Delta \subseteq \mathcal{D}$ is *conflict-free* w.r.t. a program P iff Δ does not generate a conflict w.r.t. P. $\Box$

Definition 3.5. ($\mathcal{C}^{cs}$ operator)
$\mathcal{C}^{cs}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) = \{\Delta^{\rightsquigarrow (P_1 \cup P_2) \setminus R} \mid \Delta \subseteq \mathcal{D},\ R \subseteq P_1 \cup P_2,\ \Delta \neq \Delta^{\rightsquigarrow (P_1 \cup P_2) \setminus R}$ and R is a solution of all conflicts generated by Δ w.r.t. $P_1 \cup P_2\}$. $\Box$

The following observation shows, that $\mathcal{C}^{cs}$ has a natural property that is not satisfied in dynamic logic programming.

Observation 1. Let be $\Delta \neq \Delta^{\rightsquigarrow P_1}$ and $\Delta \neq \Delta^{\rightsquigarrow P_2}$.

If Δ is conflict-free in P_1 and also in P_2, then there is a solution of all conflicts generated by Δ in $P_1 \cup P_2$.

Hence, $\mathcal{C}^{cs}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) \neq \emptyset$.

$\mathcal{C}^{cs}$ works in two steps - first a revision of a belief base is done (by rejecting some rules in order to solve conflicts) and after that the corresponding set of belief sets is computed.

Two additional remarks: Notice that a rejection of assumptions in order to solve conflicts is not needed – all sets of assumptions are considered; moreover, sometimes a rejection of assumptions makes no sense, see Example 3.6.

Example 3.6. Let $P_1 = \{a \leftarrow\}$ and $P_2 = \{\neg a \leftarrow not\ b\}$. Relevant sets of assumptions are $\emptyset$ and $\Delta = \{not\ b\}$.

There is a conflict $C = \{a, \neg a\}$ generated by Δ w.r.t. $P_1 \cup P_2$. C can be solved by $(P_1 \cup P_2) \setminus P_2$. The effect is equivalent to removing ("forgetting") the assumption $not\ b$, i.e. to replacing Δ by $\emptyset$.

On the other hand, if we replace P_1 by $\{a \leftarrow not\ b\}$ and P_2 remains unchanged, the rejection of the assumption $\{not\ b\}$ is not a reasonable solution of the underlying conflict. $\Box$

Example 3.7 shows that solving conflicts generated by a set of assumptions via rejection of some rules is reasonable, even if there is another conflict-free set of assumptions.

Example 3.7. Let $P_1 = \{r_1 : a \leftarrow; r_2 : a \leftarrow not\ b\}$ and $P_2 = \{r_3 : \neg a \leftarrow not\ b\}$.

$\emptyset$ is conflict-free w.r.t. $P_1 \cup P_2$, but a solution of the conflict $C = \{a, \neg a\}$ generated by $\Delta = \{not\ b\}$ w.r.t. $P_1 \cup P_2$ is justified, in order to capture a belief change in $\langle P_1, P_2 \rangle$.

We get $\{\emptyset^{\rightsquigarrow P_1 \cup P_2}, \Delta^{\rightsquigarrow (P_1 \cup P_2) \setminus \{r_2\}}, \Delta^{\rightsquigarrow (P_1 \cup P_2) \setminus \{r_3\}} \subseteq \mathcal{C}^{cs}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2})$. $\Box$

$\mathcal{C}^{cs}$ does not work sometimes in accord to our intuitions:

Example 3.8. Let be $P_1 = \{a \leftarrow\}, P_2 = \{\neg a \leftarrow\}$.

The operator $\mathcal{C}^{cs}$ does not distinguish between P_1 and P_2. There are two solutions of the conflict generated by the empty set of assumptions: to reject P_1 or P_2. However, information of P_2 should be more preferred, thus a preferred *solution* of the conflict is to reject P_1. $\Box$

Let us introduce now an operator $\mathcal{C}^{pref}$, which prefers the new information P_2, whenever a conflict in some Δ w.r.t. $P_1 \cup P_2$ occurs.

Definition 3.9. Consider two rules $r_1, r_2 \in P_1 \cup P_2$. We say that r_2 is more preferred than r_1 iff $r_2 \in P_2$ and $r_1 \in P_1$, notation: $r_1 \preceq r_2$.

If $r_1 \preceq r_2$ and $r_2 \not\preceq r_1$, we write $r_1 \prec r_2$. $\Box$

It is not assumed that P_1 and P_2 are disjoint.

A more preferred solution of conflicts is defined below. Note that rejected rules are solutions of conflicts. Hence, it is natural to consider less preferred information as the more preferred *solution* of conflicts.

Definition 3.10. Suppose that $Q_1, Q_2 \subseteq P_1 \cup P_2$. If there are $r_1 \in Q_1 \setminus Q_2$ and $r_2 \in Q_2 \setminus Q_1$ such that $r_2 \prec r_1$, and for all $r_3 \in Q_2 \setminus Q_1$ and for all $r_4 \in Q_1 \setminus Q_2$, $r_4 \not\prec r_3$ then Q_2 is a *more preferred solution of conflicts* than Q_1. $\Box$

Definition 3.11. Let $Q \subseteq P_1 \cup P_2$ be a solution of all conflicts generated by Δ w.r.t. $P_1 \cup P_2$.

Q is called a *preferred solution* if there is no set of rules $R \subseteq P_1 \cup P_2$ s.t. R is a solution of all conflicts generated by Δ w.r.t. $P_1 \cup P_2$ and R is more preferred solution of conflicts than Q. $\Box$

Definition 3.12. ($\mathcal{C}^{pref}$ operator)
$\mathcal{C}^{pref}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) = \{\Delta^{\rightsquigarrow (P_1 \cup P_2) \setminus R} \mid \Delta \subseteq \mathcal{D},\ R \subseteq P_1 \cup P_2,\ \Delta \neq \Delta^{\rightsquigarrow (P_1 \cup P_2) \setminus R}$ and R is a preferred solution of all conflicts generated by Δ w.r.t. $P_1 \cup P_2\}$. $\Box$

Also $\mathcal{C}^{pref}$ works in two steps – first, a belief base is changed, after that the corresponding set of belief sets.

Observation 2. $\mathcal{C}^{cs}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) \not\subseteq \mathcal{C}^{exp}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2})$.

$\mathcal{C}^{pref}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) \not\subseteq \mathcal{C}^{exp}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2})$.

Proof: Consider Example 3.8. $\emptyset^{\rightsquigarrow P_2} \in \mathcal{C}^{cs}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) \setminus \mathcal{C}^{exp}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2})$. The same for $\mathcal{C}^{pref}$. $\Box$

Observation 3. $\mathcal{C}^{pref}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) \subseteq \mathcal{C}^{cs}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2})$.

Proof: $\mathcal{C}^{pref}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) = \mathcal{C}^{cs}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) \setminus \{\Delta^{\rightsquigarrow (P_1 \cup P_2) \setminus R} \in \mathcal{C}^{cs}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) \mid R$ is not a preferred solution of all conflicts in $\Delta^{\rightsquigarrow P_1 \cup P_2}\}$. $\Box$

4. Sound sets of assumptions

The aim of the next example is to motivate a new belief change operator that enables to specify non-prioritized belief change.

Example 4.1. ([11])
Let be $P_1 = \{r_1 : a \leftarrow, r_2 : b \leftarrow a\}$. $P_2 = \{r_3 : \neg a \leftarrow not\ b\}$. $\Delta = \{not\ b\}$ generates conflicts $C_1 = \{b, not\ b\}$ and $C_2 = \{a, \neg a\}$. Both conflicts can be solved by rejecting r_1 or r_3. Rejection of r_1 is preferred.

Consider now $\Delta_1 = \emptyset$. We get $\Delta_1^{\rightsquigarrow P_1 \cup P_2} = \Delta_1^{\rightsquigarrow P_1} = \{a, b\}$. We emphasize two advantages of $\Delta_1 = \emptyset$ over Δ. First, $\emptyset$ falsifies, in a sense, Δ: $b \in \Delta_1^{\rightsquigarrow P_1 \cup P_2}$, while the (defeasible) assumption $not\ b$ is in Δ. Intuitively, the set of assumptions $\Delta = \{not\ b\}$ is defeated by its subset Δ_1; a fact (a) and its consequence (b) are dependent on $\emptyset$.

Second, Δ is a superset of Δ_1 which means that more (defeasible) assumptions are accepted in Δ than it is necessary in order to provide a conflict-free characterization of the corresponding belief change. A kind of Occam's razor (a minimization of defeasible assumptions) is accepted in our definition of the fourth belief change operator $\mathcal{C}^{sound}$. $\Box$

Example 4.1 leads to an idea that it is not reasonable to solve conflicts for (and to base a semantics on) an arbitrary set of assumptions. Two main notions of this section – defeated set of assumptions and sound set of assumptions – are introduced below.

Definition 4.2. Let a sequence $\langle P_1, P_2 \rangle$ be given and $\Delta \subset \Omega \subseteq \mathcal{D}$ be sets of assumptions. Suppose that R is a preferred solution of all conflicts generated by Δ w.r.t. $P_1 \cup P_2$.

It is said that Δ *defeats* Ω, iff there is $not\ a \in \Omega$ s.t.$a \in \Delta^{\rightsquigarrow (P_1 \cup P_2) \setminus R}$. $\Box$

The next example provides a comment to Definition 4.2. It illustrates the role of R, a (non-empty) preferred solution of all conflicts generated by Δ.

Example 4.3. Let be $P_1 = \{r_1 : a \leftarrow not\ c; r_2 : b \leftarrow a; r_3 : d \leftarrow not\ c; r_4 : \neg d \leftarrow not\ c\}, P_2 = \{r_5 : \neg a \leftarrow not\ b, not\ c\}$ and $\Delta = \{not\ c\}$.

There are two preferred solutions (no more preferred solutions exist) of all conflicts generated by Δ w.r.t. $P_1 \cup P_2$: $R = r_3$ or $R = r_4$. Thus, Δ is conflict-free w.r.t. $(P_1 \cup P_2) \setminus R$ in both cases and it defeats $\{not\ b, not\ c\}$. $\Box$

Definition 4.4. Let a sequence $\langle P_1, P_2 \rangle$ and $\Delta \subseteq \mathcal{D}$ be given.

Δ is called a *sound* set of assumptions iff Δ is not defeated by any one of its subsets. $\Box$

Example 4.5. Consider again Example 4.1. $\emptyset \subset \{not\ b\}$, $\emptyset \vdash b$, hence, $\{not\ b\}$ is defeated. The empty set of rules is the preferred solution of all conflicts generated by the empty set of assumptions (there is no more preferred solution) and the set of assumptions $\emptyset$ is not defeated. Hence $\emptyset$ is the sound set of assumptions. $\Box$

Preferred solutions of all conflicts generated by a *defeated* set of assumptions are not good candidates for a specification of belief change w.r.t. $\mathcal{C}^{sound}$.

Definition 4.6. ($\mathcal{C}^{sound}$ operator)
$\mathcal{C}^{sound}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) = \{\Delta^{\rightsquigarrow (P_1 \cup P_2) \setminus R} \mid \Delta \subseteq \mathcal{D}, R \subseteq P_1 \cup P_2,\ R$ is a preferred solution of all conflicts generated by Δ; $\Delta \neq \Delta^{\rightsquigarrow (P_1 \cup P_2) \setminus R},\ \Delta$ is a sound set of assumptions w.r.t. $R\}$. $\Box$

Proposition 4.7. $\mathcal{C}^{sound}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) \subseteq \mathcal{C}^{pref}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2})$

Proof: $\mathcal{C}^{sound}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) = \mathcal{C}^{pref}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) \setminus \{\Delta^{\rightsquigarrow (P_1 \cup P_2) \setminus R} \in \mathcal{C}^{pref}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) \mid \Delta$ is not a sound set of assumptions w.r.t. $R\}$. $\Box$

Defeat was defined for proper subsets/supersets of assumptions. However, sometimes it is useful to consider conflicts between arbitrary sets of assumptions. The following example provides a motivation.

Example 4.8. Let be $P_1 = \{a \leftarrow c\}$, $P_2 = \{c \leftarrow not\ b, c \leftarrow not\ a\}$, $\Delta_1 = \{not\ a\}, \Delta_2 = \{not\ b\}$.

A conflict is generated by Δ_1, from $not\ a$ we obtain a. Also after solving the conflict, some problems with Δ_1 remain: Δ_1 is a sound set of assumptions w.r.t. $(P_1 \cup P_2) \setminus P_1$, $\Delta_1^{\rightsquigarrow P_2} = \{not\ a, c\}$. Compare Δ_1 and Δ_2, another sound set of assumptions. $\Delta_2^{\rightsquigarrow P_1 \cup P_2} = \{not\ b, c, a\}$.

From Δ_2 is derivable a, but b is not derivable from Δ_1, it means, Δ_2 attacks Δ_1 in a sense (which will be defined in the next section), but Δ_1 is not able to counterattack Δ_2. $\Box$

To sum up the message of the Example 4.8, it is useful to compare a kind of strength of sets of assumptions. If a set of assumptions Δ attacks a set of assumptions Ω, but Ω does not counterattack Δ (is not able to defend itself), then Δ can be preferred to Ω as a stronger basis for a specification of an understanding of a belief set $\mathcal{B}_{\langle P_1, P_2 \rangle}$. An application of argumentation semantics enables to distinguish sets of assumptions, defended against attacks of other sets of assumptions. An important remark follows.

Remark 1. We will consider *also* attacks (and defense) between sets of assumptions with different solutions of all conflicts, i.e. between conflict-free Δ w.r.t. $(P_1 \cup P_2) \setminus R$ and conflict-free Ω w.r.t. $(P_1 \cup P_2) \setminus Q$, where $R \neq Q$. Motivations may be found in Examples 4.1, 4.8 and 6.2. However, this design decision should be examined more thoroughly in the future.

Argumentation semantics of extended logic programs are overviewed in the next section.

5. Argumentation Semantics of Logic Programs

An argumentation-theoretic semantics of logic programs has been proposed by Dung in [8, 9]. The semantics has been adapted for extended logic programs in [2]. Similarly, argumentation-theoretic viewpoint on nonmonotonic reasoning, particularly on logic programming is accepted in [4].[7] A dependence on a set of default literals is used also in constructions of [7]. We will adapt some basic definitions from [4] in the following paragraphs.

Definition 5.1. (Semantics)
Let a program P over a set of atoms $\mathcal{A}$ be given. A semantics σ assigns a set of sets of assumptions to P, i.e., $\sigma(P) \subseteq 2^{\mathcal{D}}$. $\Box$

[7] Regarding relations of argumentation semantics and "traditional" logic program semantics, see, e.g. [10, 4, 2, 5, 22].

Intuitively, given a program P, a semantics σ selects a subset of all possible sets of assumptions. As a consequence, a set of assumptions Δ together with the set of objective literals in $\Delta^{\rightsquigarrow_P}$ specifies a view on what is true, if we accept P and σ. In general, a semantics σ selects a set of alternative views.

Suppose that there is $S \in \sigma(P)$, $\Delta^{\rightsquigarrow_P} \in \mathcal{B}_P$ and $\Delta \in S$. Then it is said that the belief set $\Delta^{\rightsquigarrow_P}$ is a σ-belief set of P. The set of all σ-belief sets of a program P is denoted as $\mathcal{B}^{\sigma}_P$.

Conflict-freeness and defense are basic notions used for defining some semantics below. The notion of a logic program is replaced by set of rules in the wording of definitions, in order to capture also the case of conflicts solution.

Definition 5.2. Let R be a set of rules.

A set of assumptions Δ *attacks*

- an assumption $not\ l$ w.r.t. R iff $\Delta \vdash_R l$.
- a set of assumptions Δ' w.r.t. R iff Δ attacks w.r.t. R a $not\ l \in \Delta'$.

$\Box$

Consequence 1. Let $\langle P_1, P_2 \rangle$ be an ordered pair of logic programs, Δ and Ω be sets of assumptions and R be a solution of all conflicts generated by Δ w.r.t. $P_1 \cup P_2$.

Then Δ defeats Ω iff $\Delta \subset \Omega$ and Δ attacks Ω w.r.t. $(P_1 \cup P_2) \setminus R$.

Definition 5.3. Let R be a set of rules. An assumption $not\ l$ is *R-defended* by Δ, if Δ attacks w.r.t. R every Δ_1, which attacks $not\ l$ w.r.t. a set of rules Q.

Let us denote the set of all default literals R-defended by Δ as $Df_R(\Delta)$. $\Box$

We are going to define only "classical" semantics of [4]. However, it is possible to apply an arbitrary argumentation semantics to (sets of assumptions of) logic programs. The symbol σ is an abstraction of this possibility.

Definition 5.4. Let R be a set of rules, Δ be a conflict-free set of default literals w.r.t. R. Then Δ is

- *admissible* w.r.t. R iff $\Delta \subseteq Df_R(\Delta)$,
- *preferred* w.r.t. R iff it is maximal (w.r.t. $\subseteq$) admissible set of assumptions w.r.t. R,
- *complete* w.r.t. R iff $Df_R(\Delta) = \Delta$,
- *stable* w.r.t. R iff Δ attacks w.r.t. R every $not\ l \notin \Delta$,
- *well-founded* [4] w.r.t. R iff it is the intersection of all complete sets of assumptions w.r.t. R.

If we will speak about sets of assumptions, satisfying conditions of an arbitrary (abstract) semantics σ, we will use the expression σ-set of assumptions.

Proposition 5.5. ([4])
If a set of assumptions Δ is preferred, stable, well founded, respectively, then it is complete.

6. Argumentation semantics and belief change

We are going to define a belief change operator $\mathcal{C}^{\sigma}$, which outputs belief sets dependent on σ-sets of assumptions for an arbitrary argumentation semantics σ. It is argued later that sound sets of assumptions satisfying argumentation-semantic characterizations allow a more subtle and more intuitive understanding of belief change in sequences of logic programs. Hence, a combined $\mathcal{C}_{\sigma}^{sound}$ operator is defined.

Definition 6.1. ($\mathcal{C}^{\sigma}$ operator)
$\mathcal{C}^{\sigma}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) = \{\Delta^{\leadsto (P_1 \cup P_2) \setminus R} \mid \Delta \subseteq \mathcal{D}, \Delta \neq \Delta^{\leadsto (P_1 \cup P_2) \setminus R},\ \Delta$ is a σ-set of assumptions w.r.t. $(P_1 \cup P_2) \setminus R,$ where R is a preferred solution of all conflicts generated by Δ w.r.t. $P_1 \cup P_2\}$. $\Box$

Special cases of $\mathcal{C}^{\sigma}$, where σ is adm (for admissible), pref (for preferred), comp (for complete), stab (for stable), wf (for well-founded), respectively, and also for all other argumentation semantics can be defined in a straightforward way.

The following example shows that an argumentation semantics alone applied to an arbitrary set of assumptions does not provide an appropriate characterization of belief change.

Example 6.2. ([3])
Consider the case of tautological updates, observed within some semantics of dynamic logic programming.

Let be $P_1 = \{r_1 : d \leftarrow not\ h; r_2 : h \leftarrow not\ d; r_3 : \neg s \leftarrow; r_4 : s \leftarrow h, not\ c\}$ and $P_2 = \{r_5 : s \leftarrow s\}$.

Let be $\Delta = \{not\ d, not\ c, not\ \neg d, not\ \neg c, not\ \neg s, not\ \neg h\}$.

$\Delta^{\leadsto (P_1 \cup P_2) \setminus \{r_3\}} \in \mathcal{C}^{stab}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2})$: Δ is a stable set of assumptions w.r.t. $(P_1 \cup P_2) \setminus \{r_3\}$ – it is conflict-free and it attacks w.r.t. $(P_1 \cup P_2) \setminus \{r_3\}$ each assumption from $\{not\ s, not\ h\}$.

Hence, a result of $\mathcal{C}^{stab}$, satisfying conditions of stable semantics, enables so called tautological updates. $\Box$

On the other hand, we have observed in Example 4.8, that there is a problem with a sound set of assumptions, which suffers from an absence of an argumentation-theoretic characterization. To sum up, we are aiming to combine two requirements on sets of assumptions: soundness and satisfaction of conditions for some argumentation semantics σ.

Definition 6.3. ($\mathcal{C}_{\sigma}^{sound}$ operator)
$\mathcal{C}_{\sigma}^{sound}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) = \{\Delta^{\leadsto (P_1 \cup P_2) \setminus R} \mid \Delta \subseteq \mathcal{D}, \Delta \neq \Delta^{\leadsto (P_1 \cup P_2) \setminus R},\ \Delta$ is a sound and σ-set of assumptions w.r.t. $(P_1 \cup P_2) \setminus R$, where R is a preferred solution of all conflicts generated by Δ w.r.t. $P_1 \cup P_2\}$. $\Box$

Again, special cases for $\sigma \in \{adm, pref, comp, stab, wf\}$ can be defined in a straightforward way.

Proposition 6.4. $\mathcal{C}_{\sigma}^{sound} \subseteq \mathcal{C}^{sound}$,
$\mathcal{C}_{\sigma}^{sound} \subseteq \mathcal{C}^{\sigma}$. $\Box$

7. Inertia of current state

$\mathcal{C}^{sound}$ and $\mathcal{C}_{\sigma}^{sound}$ enable non-prioritized belief change. New information can be ignored, if it is based on defeated sets of assumptions. We can speak about an inertia of a current state, in a sense. In this section we show how argumentation semantics enable a more subtle analysis of non-prioritized belief change (and of an inertia of a current state).

Definition 7.1. Suppose that Δ is a complete set of assumptions w.r.t. a program P_1. If $\Delta^{\leadsto P_1} \neq \Delta$, it is said that $\Delta^{\leadsto P_1}$ is a *current state* and Δ *generates* the state $\Delta^{\leadsto P_1}$.

If $\Delta^{\leadsto P_1} = \Delta^{\leadsto P_1 \cup P_2}$, it is said that the current state *persists* in the expansion $P_1 \cup P_2$. $\Box$

The condition that Δ is a complete set of assumptions is justified by a requirement that as much information from P_1 as possible should be taken into account in descriptions of a current state, see the next example.

Example 7.2. Consider $P_1 = \{a \leftarrow, c \leftarrow not\ b\}$. Note that $\emptyset$ is an admissible set of assumptions, but not a complete one. It holds that $\emptyset^{\leadsto P_1} = \{a\}$, but it is not appropriate to consider it as a (complete) description of a current state described by P_1.

However, $\{not\ b\}^{\leadsto P_1} = \{not\ b, a, c\}$ can be considered to be the description of the current state corresponding to (specified by) P_1. $\Box$

If we would like to claim that the current state is inertial after adding a new information, some desirable properties should be satisfied. We consider as desirable that Δ is complete also w.r.t. $P_1 \cup P_2$ and there is no sound superset of Δ. The next proposition shows that $\Delta^{\leadsto P_1 \cup P_2}$ may not be a complete description of the state after a belief change, even if Δ is a complete set of assumptions w.r.t. P_1 and $\Delta^{\leadsto P_1}$ persists in expansion $P_1 \cup P_2$.

A condition that P_1 is a program over a set of atoms $\mathcal{A}$ and P_2 is a program over a set of atoms $\mathcal{A}'$, where $\mathcal{A}' \subseteq \mathcal{A}$ is accepted in this section (otherwise it does not make a sense to speak about an inertia of a current state).

Proposition 7.3. There is a sequence $\langle P_1, P_2 \rangle$ and a complete set of assumptions Δ w.r.t. P_1 s.t.

1. P_1 is a program over a set of atoms $\mathcal{A}$ and P_2 is a program over a set of atoms $\mathcal{A}'$, where $\mathcal{A}' \subseteq \mathcal{A}$,
2. Δ *generates* the *description of a current state* $\Delta^{\leadsto P_1}$ and $\Delta^{\leadsto P_1} = \Delta^{\leadsto P_1 \cup P_2}$,
3. there is a set of assumptions Ω, where $\Delta \subset \Omega$, and Ω is a sound set of assumptions,
4. Δ is not a complete set of assumptions w.r.t. $P_1 \cup P_2$

Proof:

Consider: $P_1 = \{r_1 : b \leftarrow not\ a, r_2 : p \leftarrow not\ p; r_3 : c \leftarrow not\ b\}$ and $P_2 = \{r_4 : b \leftarrow not\ p; b \leftarrow not\ c\}$.

The set of assumptions $\{not\ a\}$ is complete w.r.t. P_1 and $\{not\ a\}^{\leadsto P_1} = \{not\ a\}^{\leadsto P_1 \cup P_2}$, but $\{not\ a\}$ does not defeat $\{not\ a, not\ p\}$, which is conflict-free w.r.t. $(P_1 \cup P_2) \setminus \{r_2\}$.

Moreover, $\{not\ a\}$ is not complete w.r.t. $P_1 \cup P_2$ – it defends $not\ c$ w.r.t $P_1 \cup P_2$. $\Box$

Observation 4. Suppose that Δ is a complete set of assumptions w.r.t. P_1, Δ *generates* the *description of a current state* $\Delta^{\leadsto P_1}$ and $\Delta^{\leadsto P_1} = \Delta^{\leadsto P_1 \cup P_2}$.

Then Δ is a sound set of assumptions.

Proof: From $\Delta^{\rightsquigarrow P_1} = \Delta^{\rightsquigarrow P_1 \cup P_2}$ and from completeness w.r.t. P_1 follows that Δ is conflict-free w.r.t. $P_1 \cup P_2$. Hence, no subset of Δ attacks Δ, i.e., Δ is sound. $\Box$

Proposition 7.4. (Inertia of a current state)
Let Δ be a stable set of assumptions w.r.t. P_1. Let P_1 be a program over a set of atoms $\mathcal{A}$ and P_2 a program over a set of atoms $\mathcal{A}'$, where $\mathcal{A}' \subseteq \mathcal{A}$. Suppose that Δ *generates* the *description of a current state* $\Delta^{\rightsquigarrow P_1}$ and $\Delta^{\rightsquigarrow P_1} = \Delta^{\rightsquigarrow P_1 \cup P_2}$. Then

- for no Ω s.t. $\Delta \subset \Omega$ and for no $R \subseteq P_1 \cup P_2$ holds that Ω is a sound set of assumptions w.r.t. R,
- Δ is a stable set of assumptions w.r.t. $P_1 \cup P_2$.

Proof: Suppose that $\Delta \subset \Omega$. From the assumption that Δ is stable follows that Δ attacks every $not\ l \in \Omega \setminus \Delta$. Therefore Δ defeats Ω and Ω cannot be a sound set of assumptions w.r.t. any R.

It is supposed that Δ is a stable set of assumptions w.r.t. P_1 and $\Delta^{\rightsquigarrow P_1} = \Delta^{\rightsquigarrow P_1 \cup P_2}$. Hence Δ attacks also w.r.t. $P_1 \cup P_2$ each $not\ l \notin \Delta$. $\Box$

We speak about inertia of a current state because the description of a current state $\Delta^{\rightsquigarrow P_1} = \Delta^{\rightsquigarrow P_1 \cup P_2}$ cannot be extended (made "more complete") in the sense that there is no sound superset of Δ. Moreover, the stability of Δ is preserved.

It is shown by Propositions 7.3 and 7.4 that argumentation semantics introduces a more subtle insight into understanding of non-prioritized belief change in $\langle P_1, P_2 \rangle$.

Now a look on the fact that there are more alternative current states, but it is possible that only some of them are inertial, see the next example.

Example 7.5. $P_1 = \{d \leftarrow not\ c, a \leftarrow d, c \leftarrow not\ d\}, P_2 = \{d \leftarrow c\}$.

Consider stable sets of assumptions w.r.t. P_1: $\Delta_1 = \{not\ c\}$, it attacks $not\ a$ and $not\ d$, and $\Delta_2 = \{not\ d, not\ a\}$, it attacks $not\ c$. Both generate a description of a current state: $\Delta_i \neq \Delta_i^{\rightsquigarrow P_1}, i = 1, 2$ and the set of atoms used in P_2 is a subset of atoms used in P_1.

$\Delta_1^{\rightsquigarrow P_1} = \Delta_1^{\rightsquigarrow P_1 \cup P_2}$, hence, conditions of Proposition 7.4 are satisfied for Δ_1.

On the other hand, $\Delta_2^{\rightsquigarrow P_1} \neq \Delta_2^{\rightsquigarrow P_1 \cup P_2}$. Notice that a solution of a conflict generated by Δ_2 w.r.t. $P_1 \cup P_2$ is needed.

As a consequence, there are two current states in this example, but only one of them is inertial after the addition of P_2. $\Box$

8. Irrelevant belief change

A comment is needed before the definition of an irrelevant belief change. Intuitively, a change is specified by P_2 and the role of P_2 is irrelevant for a belief change operator, if the operator returns the empty set of belief sets or a set of belief sets of P_1. However, we consider also a third option. Conflict solving is a component of most operators introduced in the paper, but a more complex behaviour is expected from those operators (except of $\mathcal{C}^{cs}$). Therefore, a change, specified by P_2, is considered as irrelevant for an operator also if the operator returns only conflict-free belief sets of P_1 (also for $\mathcal{C}^{cs}$, in order to simplify things).

Definition 8.1. $\overline{\mathcal{B}_{P_1}} = \mathcal{C}^{cs}(\mathcal{B}_{P_1}, \mathcal{B}_{P_1})$. $\Box$

Definition 8.2. Let $\mathcal{C}$ be belief change operator and $\langle P_1, P_2\rangle$ be a sequence of programs.

Then a belief change, specified by P_2 is *irrelevant* from the viewpoint of $\mathcal{C}$, if $\mathcal{C}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) = \mathcal{B}_{P_1}$ or $\mathcal{C}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) \subseteq \overline{\mathcal{B}_{P_1}}$[8] or $\mathcal{C}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) = \emptyset$. □

Observation 5. If $P_2 \subseteq P_1$, then P_2 is irrelevant from the viewpoint of $\mathcal{C}^{exp}$. □

Irrelevant belief changes are identified in Propositions 8.3, 8.4, 8.5, 8.8, 8.9 and in Observation 6

Proposition 8.3. If $P_2 \subseteq P_1$, then $\mathcal{C}^{cs}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) = \overline{\mathcal{B}_{P_1}}$.

Proof: $\mathcal{C}^{cs}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) = \{\Delta^{\rightsquigarrow (P_1 \cup P_2)\setminus R} \mid \Delta \subseteq \mathcal{D},\ R \subseteq P_1 \cup P_2$ and R is a solution of all conflicts generated by Δ w.r.t. $P_1 \cup P_2\} = \{\Delta^{\rightsquigarrow P_1\setminus R} \mid \Delta \subseteq \mathcal{D},\ R \subseteq P_1$ and R is a solution of all conflicts generated by Δ w.r.t. $P_1\} = \overline{\mathcal{B}_{P_1}}$. □

Proposition 8.4. If $P_2 \subseteq P_1$, then $\mathcal{C}^{\sigma}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) \subseteq \overline{\mathcal{B}_{P_1}}$, where $\sigma \in \{adm, pref, stab, wf\}$.

Proposition 8.5. There are $\langle P_1, P_2\rangle$ s.t. $P_2 \subseteq P_1$ and $\mathcal{C}^{pref}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) \neq \overline{\mathcal{B}_{P_1}}$. But the subset condition of irrelevancy is satisfied.

The same holds for $\mathcal{C}^{sound}$ and $\mathcal{C}^{sound}_{\sigma}$, where $\sigma \in \{adm, pref, stab, wf\}$.

Proof: Consider $P_1 = \{a \leftarrow, \neg a \leftarrow\}$ and $P_2 = \{a \leftarrow\}$. □

Definition 8.6. A literal $l \in \mathcal{O}$ *depends immediately* w.r.t. a set of rules R on a literal $not\ k \in \mathcal{D}$ (denoted by $l \lhd_R not\ k$) iff there is a rule $r \in R$ s.t. $l = head(r)$ and $not\ k \in body(r)$. □

Observation 6. Suppose that for all $l \in \mathcal{O}$ occurring in $P_1 \cup P_2$ holds that $l \lhd_{P_1 \cup P_2} not\ l$ and there is no rule r with $head(r) = l$ and $not\ l \notin body(r)$.

Then $\mathcal{C}^{cs}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) = \emptyset = \mathcal{C}^{pref}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) = \mathcal{C}^{sound}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) = \mathcal{C}^{\sigma}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2})$. □

Definition 8.7. A literal $l \in \mathcal{O}$ depends w.r.t. a set of rules R on a literal $k \in \mathcal{L}$ (denoted by $l \ll_R k$) iff there is a rule $r \in R$ s.t. $l \in head(r)$ and

- $k \in body(r)$, or
- there is a literal $m \in \mathcal{O}$ s.t. $m \in body(r)$ and $m \ll_R k$.

□

It is supposed in the next observation that at least for one killer of the form $l \leftarrow not\ l$ in $P_1 \cup P_2$ there is a rule, which defines l without a reference to $not\ l$, i.e., $l = head(r), not\ (l) \notin body(r)$. It is a sufficient condition for existence of an non-empty output of $\mathcal{C}^{cs}$.

Observation 7. (Existence)
Suppose $l \in \mathcal{O}$ occurring in $P_1 \cup P_2$ satisfying the following condition: if $l \lhd_{P_1 \cup P_2} not\ l$ then there is a rule $r \in P_1 \cup P_2$ s.t. $head(r) = l$ and $not\ l \notin body(r)$,

Then $\mathcal{C}^{cs}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) \neq \emptyset$.

[8] σ-sets of assumptions are subsets of conflict-free sets of assumptions.

Proof: If some Δ generates a conflict $C = \{l, \neg l\}$, there is a solution of that conflict.

If $l \ll_{P_1 \cup P_2} not\ l$, $not\ l \in \Delta$, then there is a non-empty sequence of literals between $not\ l$ and l in a proof of l from Δ. Hence, there is a solution of that conflict. $\square$

Consequence 2. If conditions of Observation 7 are satisfied, then $\mathcal{C}^{pref}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) \neq \emptyset$. $\square$

Cyclic (i.e., also tautological) belief change is irrelevant:

Proposition 8.8. Suppose that for all $l \in \mathcal{O}$ occurring in P_2 holds that $l \ll_{P_2} l$ and there is no $not\ k$ s.t. $l \ll_{P_1 \cup P_2} not\ k$, then $\mathcal{C}^x(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) = \mathcal{B}_{P_1}$, where $x \in \{cs, pref, sound, \sigma\}$. $\square$

Proposition 8.9. If for each $\Delta \subseteq \mathcal{D}$ holds $\Delta^{\leadsto P_1} = \Delta^{\leadsto P_2}$, then $\mathcal{C}^{cs}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) = \overline{\mathcal{B}_{P_1}}$.

Proof: If $l \ll_{P_2} k$ and $m \ll_{P_1} l$, then also $m \ll_{P_2} l$. Similarly, if $l \ll_{P_1} k$ and $m \ll_{P_2} l$, then $m \ll_{P_1} l$.

Therefore: $\Delta^{\leadsto P_1} = \Delta^{\leadsto P_1} \cup \Delta^{\leadsto P_2} = \Delta^{\leadsto P_1 \cup P_2}$ $\square$

An initial study of irrelevant updates in [18] has been motivated by the problems expressed now in terms of defeated and sound sets of assumptions. The next proposition characterizes an irrelevant belief change for $\mathcal{C}^{sound}$.

Proposition 8.10. If for every Δ_2 s.t. $\Delta_2 \neq \Delta_2^{\leadsto P_2}$ there is Δ_1, which defeats Δ_2, then $\mathcal{C}^{sound}(\mathcal{B}_{P_1}, \mathcal{B}_{P_2}) = \overline{\mathcal{B}_{P_1}}$.

9. Conclusions

Main contributions. A set of belief change operators was introduced in the paper. An argumentation point of view was emphasized: output of operators $\mathcal{C}^{sound}, \mathcal{C}^{\sigma}, \mathcal{C}^{sound}_{\sigma}$ is specified in terms of undefeated (sound) sets of assumptions or in terms of an arbitrary argumentation semantics. Soundness allows non-prioritized belief change. It was argued in favour of a combined application of an argumentation semantics σ and sound sets of assumptions, which is enabled by $\mathcal{C}^{sound}_{\sigma}$.

Further, the paper provides an analysis of an inertia of a current state and of irrelevant belief change. Both belong to novel features in the investigation of belief change within logic programs. It was shown that different argumentation semantics (complete vs. stable) manifest different behaviour from the inertia point of view.

Regarding features of belief change within logic programs, it is shown that so called cyclic belief change is a special case of irrelevant belief change and all our operators avoid cyclic belief change. Further, the satisfiability principle (if both P_1 and P_2 are satisfiable, then $\langle P_1, P_2 \rangle$ is also satisfiable), which is not satisfied in approaches to logic program updates, based on rejection of rules, is satisfied in our approach, see Observation 1.

A brief comparison to [21] is needed. Belief change operators are a new feature. Focus is moved from updates to belief change. More detailed elaboration of an inertia of a current state and of an irrelevant belief change is given.

Related work We are not aware of another application of argumentation semantics to belief change of logic programs. However, relations between belief change and argumentation are topics of current research. Most of the research is focused on applications of belief change (dynamics) of argumentation frameworks.

Non-prioritized belief change, supported by argumentation is presented in [13], where deductive argumentation is used to assess the value of new information. Only justified new information is accepted.

Regarding approaches to belief change within logic programs based on rejection of rules, e.g. [1, 3, 11, 15, 14], we emphasize here the non-prioritized character of our operators $\mathcal{C}^{sound}, \mathcal{C}^{\sigma}, \mathcal{C}_{\sigma}^{sound}$. Some other differences are mentioned among main contributions of this paper.

Open problems, future work We plan an exhaustive comparison of our approach to other approaches to belief change within logic programs. Similarly, a more detailed comparison to the more general research concerning belief change and argumentation represents a necessary continuation of the presented work.

Investigation of (un)satisfaction of fundamental properties (postulates) belongs among our important research goals. Particularly, a further elaboration of postulates, specific for belief change in non-monotonic formalisms is necessary and represents a challenging research topic. We can follow the results of [20].

Much more work is needed for a deeper understanding of belief change operators, based on argumentation semantics, also on semantics not considered in this paper. We plan to explore main features and differences determined by various argumentation semantics. Also a further description and analysis of irrelevant belief change represents a challenge for future research.

Acknowledgment. We are grateful to anonymous reviewers for many helpful comments.

References

[1] J.J. Alferes, Leite J.A., L.M. Pereira, H. Przymusinska, and T.C. Przymusinski. Dynamic logic programming. In *Procs. of KR'98*, pages 98–109, 1998.

[2] José Júlio Alferes, Phan Minh Dung, and Luís Moniz Pereira. Scenario semantics of extended logic programs. In *LPNMR*, pages 334–348, 1993.

[3] F. Banti, J.J. Alferes, A. Brogi, and P. Hitzler. The well supported semantics for multidimensional dynamic logic programs. In *LPNMR, LNCS 3662, Springer*, pages 356–368, 2005.

[4] Andrei Bondarenko, Phan Minh Dung, Robert A. Kowalski, and Francesca Toni. An abstract, argumentation-theoretic approach to default reasoning. *Artif. Intell.*, 93:63–101, 1997.

[5] Martin Caminada, Samy Sá, João Alcântara, and Wolfgang Dvorák. On the equivalence between logic programming semantics and argumentation semantics. *Int. J. Approx. Reasoning*, 58:87–111, 2015.

[6] J. Delgrande, T. Schaub, H. Tompits, and S. Woltran. Belief revision of logic programs under answer set semantics. In *Proc. of KR 2008*, pages 411–421, 2008.

[7] Yannis Dimopoulos and Alberto Torres. Graph theoretical structures in logic programs and default theories. *Theor. Comput. Sci.*, 170(1–2):209–244, 1996.

[8] Phan Minh Dung. Negations as hypotheses: An abductive foundation for logic programming. In *Logic Programming, Proceedings of the Eigth International Conference, Paris, France, June 24-28, 1991*, pages 3–17, 1991.

[9] Phan Minh Dung. An argumentation-theoretic foundations for logic programming. *J. Log. Program.*, 22(2):151–171, 1995.

[10] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.*, 77(2):321–358, 1995.

[11] T. Eiter, G. Sabbatini, M. Fink, and H. Tompits. On properties of update sequences based on causal rejection. *Theory and Practice of Logic Programming*, pages 711–767, 2002.

[12] Sven Ove Hansson, Eduardo L. Fermé, John Cantwell, and Marcelo A. Falappa. Credibility limited revision. *J. Symb. Log.*, 66(4):1581–1596, 2001.

[13] Patrick Krümpelmann, Matthias Thimm, Marcelo A. Falappa, Alejandro J. García, Gabriele Kern-Isberner, and Guillermo R. Simari. Selective revision by deductive argumentation. In *Proceedings of the First International Conference on Theory and Applications of Formal Argumentation*, TAFA'11, pages 147–162. Springer-Verlag, 2012.

[14] J.A. Leite. *Evolving Knowledge Bases: Specification and Semantics*. IOS Press, 2003.

[15] J.A. Leite, J.J. Alferes, and L.M. Pereira. Multi-dimensional dynamic logic programming. In *Procs. of CLIMA'00*, pages 17–26, 2000.

[16] D. Makinson. Screened revision. *Theoria*, 63(1-2):14–23, 1997.

[17] Mauricio Osorio and Víctor Cuevas. Updates in answer set programming: An approach based on basic structural properties. *TPLP*, 7(4):451–479, 2007.

[18] Ján Sefránek. Irrelevant updates and nonmonotonic assumptions. In *Logics in Artificial Intelligence, 10th European Conference, JELIA 2006, Liverpool, UK, September 13-15, 2006, Proceedings*, pages 426–438, 2006.

[19] Martin Slota and João Leite. On semantic update operators for answer-set programs. In *ECAI 2010 - 19th European Conference on Artificial Intelligence, Lisbon, Portugal, August 16-20, 2010, Proceedings*, pages 957–962, 2010.

[20] J. Šefránek. Static and dynamic semantics. preliminary report. In *MICAI 2011, Special Session, Mexican International Conference on Artificial Intelligence*, 2011.

[21] J. Šefránek. Updates of assumption-based frameworks. In *Procs. of NMR*, 2012.

[22] Yining Wu, Martin Caminada, and Dov M. Gabbay. Complete extensions in argumentation coincide with 3-valued stable models in logic programming. *Studia Logica*, 93(2–3):383–403, 2009.