

Warranted Derivations of Preferred Answer Sets

Ján Šefránek and Alexander Šimko

Comenius University, Bratislava, Slovakia; [sefranek,simko]@fmph.uniba.sk

Abstract. We are aiming at a semantics of logic programs with preferences defined on rules, which always selects a preferred answer set, if there is a non-empty set of (standard) answer sets of the given program.

It is shown in a seminal paper by Brewka and Eiter that the goal mentioned above is incompatible with their second principle and it is not satisfied in their semantics of prioritized logic programs. Similarly, also according to other established semantics, based on a prescriptive approach, there are programs with standard answer sets, but without preferred answer sets.

According to the standard prescriptive approach no rule can be fired before a more preferred rule, unless the more preferred rule is blocked. This is a rather imperative approach, in its spirit.

In our approach, rules can be blocked by more preferred rules, but the rules which are not blocked are handled in a more declarative style, their execution does not depend on the given preference relation on the rules.

An argumentation framework (different from the Dung’s framework) is proposed in this paper. Argumentation structures are derived from the rules of a given program. An attack relation on argumentation structures is defined, which is derived from attacks of more preferred rules against the less preferred rules. Preferred answer sets correspond to complete argumentation structures, which are not blocked by other complete argumentation structures.

Keywords: extended logic program, answer set, preference, preferred answer set, argumentation structure

1 Introduction

The meaning of a nonmonotonic theory is often characterized by a set of (*alternative*) *belief sets*. It is appropriate to select sometimes some of the belief sets as more *preferred*.

We are focused in this paper on *extended logic programs* with a preference relation on rules, see, e.g., [1, 3, 9, 15]. Such programs are denoted by the term *prioritized* logic programs in this paper.

It is suitable to require that some preferred answer sets can be *selected* from a non-empty set of standard answer sets of a (prioritized) logic program.

Unfortunately, there are prioritized logic programs with standard answer sets, but without preferred answer sets according to the semantics of [1] and also of [3] or [15]. This feature is a consequence of the *prescriptive* approach to preference handling [4]. According to that approach no rule can be fired before a more preferred rule, unless the more preferred rule is blocked. This is a rather imperative approach, in its spirit.

An investigation of basic *principles* which should be satisfied by any system containing a preference relation on defeasible rules is of fundamental importance. This type of research has been initialized in the seminal paper [1]. Two basic principles are accepted in the paper.

The second of the principles is violated, if a function is assumed, which always selects a non-empty subset of preferred answer sets from a non-empty set of all standard answer sets of a prioritized logic program.

We believe that the possibility to select always a preferred answer set from a non-empty set of standard answer sets is of critical importance. This goal requires to accept a *descriptive* approach to preference handling. The approach is characterized in [5, 4] as follows: The order in which rules are applied, reflects their “desirability” – it is desirable to apply the most preferred rules.

In our approach, rules can be *blocked* by more preferred rules, but the rules which are not blocked are handled in a more declarative style. If we express this in terms of desirability, it is desirable to apply all

(applicable) rules, which are not blocked by a more preferred rule. The execution of non-blocked rules does not depend on their order. Dependencies of more preferred rules on less preferred rules do not prevent the execution of non-blocked rules.

Our goal is:

- to modify the Principles proposed by [1] in such a way that they do not contradict a selection of a non-empty set of preferred answer sets from the underlying non-empty set of standard answer sets,
- to introduce a notion of preferred answer sets that satisfies the above mentioned modification.

The proposed method is inspired by [7]. While there defeasible rules are treated as (defeasible) arguments, here (defeasible) assumptions, sets of default negations, are considered as arguments. Reasoning about preferences in a logic program is here understood as a kind of argumentation. Sets of default literals can be viewed as defeasible arguments, which may be contradicted by consequences of some applicable rules. The preference relation on rules is used in order to ignore the attacks of less preferred arguments against more preferred arguments. The core problem is to transfer the preference relation defined on rules to argumentation structures and, consequently, to answer sets.¹

The basic argumentation structures correspond to the rules of a given program. Derivation rules, which enable derivation of argumentation structures from the basic ones are defined. That derivation leads from the basic argumentation structures (corresponding to the rules of a given program) to argumentation structures corresponding to the rules of an negative equivalent of the given program introduced in [6].

Attacks of more preferred rules against the less preferred rules are transferred via another set of derivation rules to the attacks between argumentation structures. Preferred answer sets are defined over that background. They correspond to complete and non-blocked (warranted) argumentation structures.

The contributions of this paper are summarized as follows. A modified set of principles for preferred answer set specification is proposed. A new type of argumentation framework is constructed, which enables a selection of preferred answer sets. There are basic arguments (argumentation structures) and attacks in the framework and also derived arguments and attacks. Rules for derivation of argumentation structures and rules for derivation of attacks of some argumentation structures against other argumentation structures are defined. Preferred answer sets are defined in terms of complete and non-blocked (warranted) argumentation structures. Finally, we emphasize that each program with a non-empty set of answer sets has a preferred answer set.

A preliminary version of the presented research has been published in [11]. The main differences between the preliminary and the current version are summarized in the Conclusions.² An extended version of this paper with proofs is accessible as [12].

2 Preliminaries

The language of extended logic programs is used in this paper.

Let At be a set of atoms. The set of *objective literals* is defined as $Obj = At \cup \{\neg A : A \in At\}$. If L is an objective literal then the expression of the form *not* L is called *default* literal. Notation: $Def = \{\text{not } L \mid L \in Obj\}$. The set of literals Lit is defined as $Obj \cup Def$.

A convention: $\neg\neg A = A$, where $A \in At$. If X is a set of objective literals, then $\text{not } X = \{\text{not } L \mid L \in X\}$.

A *rule* is each expression of the form $L \leftarrow L_1, \dots, L_k$, where $k \geq 0$, $L \in Obj$ and $L_i \in Lit$. If r is a rule of the form as above, then L is denoted by $head(r)$ and $\{L_1, \dots, L_k\}$ by $body(r)$. If R is a set of rules, then $head(R) = \{head(r) \mid r \in R\}$ and $body(R) = \{body(r) \mid r \in R\}$. A finite set of rules is called *extended logic program* (program hereafter).

¹ Our intuitions connected to the notion of argumentation structure and also the used constructions are different from Dung's arguments or from arguments of [7, 2]. On the other hand, we plan an elaboration of presented constructions aiming at a theory, which generalizes Dung's abstract argumentation framework, TMS, constructions given, e.g., by [7] or [2]. Anyway, this paper does not present a contribution to argumentation theory.

² They are described in technical terms, assuming a familiarity with this paper. Most importantly, the notion of preferred answer set is changed in this paper.

The set of *conflicting literals* is defined as $CON = \{(L_1, L_2) \mid L_1 = \text{not } L_2 \vee L_1 = \neg L_2\}$. A set of literals S is *consistent* if $(S \times S) \cap CON = \emptyset$. An *interpretation* is a consistent set of literals. A *total interpretation* is an interpretation I such that for each objective literal L either $L \in I$ or $\text{not } L \in I$. A literal L is *satisfied* in an interpretation I iff $L \in I$ (notation: $I \models L$). A set of literals S is satisfied in I iff $S \subseteq I$ (notation: $I \models S$). A rule r is satisfied in I iff $I \models \text{head}(r)$ whenever $I \models \text{body}(r)$.

If S is a set of literals, then we denote $S \cap \text{Obj}$ by S^+ and $S \cap \text{Def}$ by S^- . Symbols $(\text{body}(r))^-$ and $(\text{body}(r))^+$ are used here in that sense (notice that the usual meaning of $\text{body}^-(r)$ is different). If $X \subseteq \text{Def}$ then $\text{pos}(X) = \{L \in \text{Obj} \mid \text{not } L \in X\}$. Hence, $\text{not pos}((\text{body}(r))^-) = (\text{body}(r))^+$. If r is a rule, then the rule $\text{head}(r) \leftarrow (\text{body}(r))^+$ is denoted by r^+ .

An answer set of a program can be defined as follows (only consistent answer sets are defined).

A total interpretation S is an *answer set* of a program P iff S^+ is the least model³ of the program $P^+ = \{r^+ \mid S \models (\text{body}(r))^- \}$. Note that an answer set S is usually represented by S^+ (this convention is sometimes used also in this paper).

The set of all answer sets of a program P is denoted by $AS(P)$. A program is called *coherent* iff it has an answer set.

Strict partial order is a binary relation, which is irreflexive, transitive and, consequently, asymmetric.

A *prioritized logic program* is usually defined as a triple (P, \prec, \mathcal{N}) , where P is a program, \prec a strict partial order on rules of P and a function \mathcal{N} assigns names to rules of P . If $r_1 \prec r_2$ it is said that r_2 is more preferred than r_1 .

We will not use \mathcal{N} . If a symbol r is used in this paper in order to denote a rule, then r is considered as the name of that rule (no different name $\mathcal{N}(r)$ is introduced).

3 Argumentation Structures

Our aim is to transfer a preference relation defined on rules to a preference relation on answer sets and, finally, to a notion of preferred answer sets. To that end argumentation structures are introduced. The basic argumentation structures correspond to rules. Some more general types of argumentation structures are derived from the basic argumentation structures. A special type of argumentation structures corresponds to answer sets.

Definition 1 (\ll_P , [10]) An objective literal L *depends* on a set of default literals $W \subseteq \text{Def}$ with respect to a program P ($L \ll_P W$)⁴ iff there is a sequence of rules $\langle r_1, \dots, r_k \rangle$, $k \geq 1$, $r_i \in P$ such that

- $\text{head}(r_k) = L$,
- $W \models \text{body}(r_1)$,
- for each i , $1 \leq i < k$, $W \cup \{\text{head}(r_1), \dots, \text{head}(r_i)\} \models \text{body}(r_{i+1})$.

The set $\{L \in \text{Lit} \mid L \ll_P W\} \cup W$ is denoted by $Cn_{\ll_P}(W)$.

$W \subseteq \text{Def}$ is *self-consistent* w.r.t. a program P iff $Cn_{\ll_P}(W)$ is consistent. \square

If $Z \subseteq \text{Obj}$, we will use sometimes the notation $Cn_{\ll_{P \cup Z}}(W)$, assuming that the program P is extended by the set of facts Z .

Definition 2 (Dependency structure) Let P be a program.

A self-consistent set $X \subseteq \text{Def}$ is called an *argument* w.r.t. the program P for a consistent set of objective literals Y , given a set of objective literals Z iff

1. $\text{pos}(X) \cap Z = \emptyset$,
2. $Y \subseteq Cn_{\ll_{P \cup Z}}(X)$.

We will use the notation $\langle Y \leftrightarrow X; Z \rangle$ and the triple denoted by it is called a *dependency structure* (w.r.t. P). \square

³ P^+ is treated as definite logic program, i.e., each objective literal of the form $\neg A$, where $A \in \text{At}$, is considered as a new atom.

⁴ $L \ll_P W$ could be defined as $T_P^\omega(W)$ and $Cn_{\ll_P}(W)$ as $T_P^\omega(W)$

If $Z = \emptyset$ also a shortened notation $\langle Y \leftrightarrow X \rangle$ can be used. We will omit sometimes the phrase “w.r.t. P ” and speak simply about dependency structures and arguments, if the corresponding program is clear from the context.

Basic argumentation structures comply with Definition 2 of dependency structures, if some conditions are satisfied:

Definition 3 (Basic argumentation structure) Let $r \in P$ be a rule such that

- $(body(r))^-$ is self-consistent and
- $pos((body(r))^-) \cap (body(r))^+ = \emptyset$.

Then $\mathcal{A} = \langle \{head(r)\} \leftrightarrow (body(r))^-; (body(r))^+ \rangle$ is called a *basic argumentation structure*. \square

Proposition 4 Each basic argumentation structure is a dependency structure. \square

We emphasize that only *self-consistent* arguments for *consistent* sets of objective literals are considered in this paper. Hence, programs as $P = \{p \leftarrow not\ p\}$ or $Q = \{p \leftarrow not\ q; \neg p \leftarrow not\ q\}$ are irrelevant for our constructions.

Some dependency structures can be derived from the basic argumentation structures. Only the dependency structures derived from the basic argumentation structures using derivation rules from Definition 5 are of interest in the rest of this paper, we will use the term *argumentation structure* for dependency structures derived from basic argumentation structures using derivation rules.

Derivation rules are motivated in Example 6.

Definition 5 (Derivation rules and argumentation structures) Each basic argumentation structure is an argumentation structure. Let P be a program.

R1 Let $r_1, r_2 \in P$, $\mathcal{A}_1 = \langle \{head(r_1)\} \leftrightarrow X_1; Z_1 \rangle$ and $\mathcal{A}_2 = \langle \{head(r_2)\} \leftrightarrow (body(r_2))^-; (body(r_2))^+ \rangle$ be argumentation structures, $head(r_2) \in Z_1$, $X_1 \cup (body(r_2))^- \cup Z_1 \cup (body(r_2))^+ \cup \{head(r_1)\}$ be consistent and $X_1 \cup (body(r_2))^-$ be self-consistent.

Then also $\mathcal{A}_3 = \langle head(r_1) \leftrightarrow X_1 \cup (body(r_2))^-; (Z_1 \setminus \{head(r_2)\}) \cup (body(r_2))^+ \rangle$ is an argumentation structure. We also write $\mathcal{A}_3 = u(\mathcal{A}_1, \mathcal{A}_2)$. We define u as a symmetric relation: $u(\mathcal{A}_1, \mathcal{A}_2) = u(\mathcal{A}_2, \mathcal{A}_1)$ ⁵

R2 Let $\mathcal{A}_1 = \langle Y_1 \leftrightarrow X_1 \rangle$ and $\mathcal{A}_2 = \langle Y_2 \leftrightarrow X_2 \rangle$ be argumentation structures and $X_1 \cup X_2 \cup Y_1 \cup Y_2$ be consistent and $X_1 \cup X_2$ be self-consistent.

Then also $\mathcal{A}_3 = \langle Y_1 \cup Y_2 \leftrightarrow X_1 \cup X_2 \rangle$ is an argumentation structure. We also write $\mathcal{A}_3 = \mathcal{A}_1 \cup \mathcal{A}_2$.

R3 Let $\mathcal{A}_1 = \langle Y_1 \leftrightarrow X_1 \rangle$ be an argumentation structure and $W \subseteq Def$.

If $X_1 \cup W \cup Y_1$ is consistent and $X_1 \cup W$ is self-consistent, then also $\mathcal{A}_2 = \langle Y_1 \leftrightarrow X_1 \cup W \rangle$ is an argumentation structure. We also write $\mathcal{A}_2 = \mathcal{A}_1 \cup W$. \square

Example 6 ([1]) Let a program P be given (P is used as a running example in this paper):

$$\begin{array}{ll} r_1 & b \leftarrow a, not\ \neg b \\ r_2 & \neg b \leftarrow not\ b \\ r_3 & a \leftarrow not\ \neg a. \end{array}$$

Suppose that $\prec = \{(r_2, r_1)\}$.

Consider the rule r_2 . The default negation $not\ b$ plays the role of a *defeasible argument*. If the argument can be consistently evaluated as true with respect to a program containing r_2 , then also $\neg b$ can (and must) be evaluated as true.

As regards the rule r_1 , default negation $not\ \neg b$ can be treated as an argument for b , if a is true, it is an example of a “conditional argument”.

The following basic argumentation structures correspond to the rules of P :

$\langle \{b\} \leftrightarrow \{not\ \neg b\}; \{a\} \rangle, \langle \{\neg b\} \leftrightarrow \{not\ b\} \rangle, \langle \{a\} \leftrightarrow \{not\ \neg a\} \rangle$. Let us denote them by $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3$, respectively.

⁵ Symmetry of u enables below a more succinct formulation of derivation rules Q1, Q2. The symbol u indicates that \mathcal{A}_3 is a result of an unfolding.

Some arguments can be treated as counterarguments against other arguments. If we accept the argument *not b* (with the consequence $\neg b$), it can be treated as a counterargument to *not $\neg b$* and, similarly, *not $\neg b$* (with the consequence b , if a is true) as a counterargument against *not b*. On the level of argumentation structures it can be said that \mathcal{A}_1 contradicts \mathcal{A}_2 and vice versa.

The preference relation can be directly transferred to *basic* argumentation structures, hence \mathcal{A}_1 is more preferred than \mathcal{A}_2 . Consequently, only the attack of \mathcal{A}_1 against \mathcal{A}_2 is relevant.

An example of a derived argumentation structure: \mathcal{A}_3 enables to “unfold” the condition a in \mathcal{A}_1 , the resulting argumentation structure can be expressed as $\mathcal{A}_4 = \langle \{b\} \leftrightarrow \{\text{not } \neg b, \text{not } \neg a\} \rangle$. Similarly, $\mathcal{A}_5 = \langle \{a, b\} \leftrightarrow \{\text{not } \neg b, \text{not } \neg a\} \rangle$ can be derived from \mathcal{A}_3 and \mathcal{A}_4 , $\mathcal{A}_5 = \mathcal{A}_3 \cup \mathcal{A}_4$.

We will also transfer the attack relation from the basic argumentation structures to the derived argumentation structures.

Observe that some argumentation structures correspond to answer sets. \mathcal{A}_5 corresponds to the answer set $\{a, b\}$ and $\mathcal{A}_6 = \langle \{a, \neg b\} \leftrightarrow \{\text{not } b, \text{not } \neg a\} \rangle$ to $\{a, \neg b\}$. Notice that $\mathcal{A}_6 = \mathcal{A}_2 \cup \mathcal{A}_3$. The attack relation enables to select the preferred answer set. This will be discussed later in Example 19. \square

Proposition 7 *Each argumentation structure is a dependency structure.*

Proof. We have to show that an application of R1, R2 and R3 preserves properties of dependency structures.

R1 Since $S_1 = X_1 \cup (\text{body}(r_2))^- \cup Z_1 \cup (\text{body}(r_2))^+ \cup \{\text{head}(r_1)\}$ is consistent $S_2 = X_1 \cup (\text{body}(r_2))^- \cup (Z_1 \setminus \{\text{head}(r_2)\}) \cup (\text{body}(r_2))^+ \subseteq S_1$ is also consistent. It means $\text{pos}(X_1 \cup (\text{body}(r_2))^-) \cap ((Z_1 \setminus \{\text{head}(r_2)\}) \cup (\text{body}(r_2))^+) = \emptyset$.

Let $Q = P \cup (Z_1 \setminus \{\text{head}(r_2)\}) \cup (\text{body}(r_2))^+$ and $w = \text{head}(r_2) \leftarrow$.

From $\text{head}(r_2) \in \text{Cn}_{\ll_{P \cup (\text{body}(r_2))^+}}((\text{body}(r_2))^-)$ we have sequence of rules $R_2 = \langle q_1, q_2, \dots, q_m \rangle$ where $m > 0$ and $q_m = r_2$.

From $\text{head}(r_1) \in \text{Cn}_{\ll_{P \cup Z_1}}(X_1)$ we have sequence of rules $R_1 = \langle p_1, p_2, \dots, p_n \rangle$ where $n > 0$ and $p_n = r_1$. We assume there is at most one occurrence of w in R_1 . Otherwise we can remove all but leftmost one. Note that since $r_2 \in P$ there is a possibility to satisfy $\text{body}(r_1)$ in a different way than using w .

If $w \in R_1$ then we have $p_i = w$ for some $1 \leq i < n$. We construct sequence

$R_3 = \langle q_1, q_2, \dots, q_m, p_1, p_2, \dots, p_{i-1}, p_{i+1}, \dots, p_n \rangle$. If $w \notin R_1$ we construct sequence

$R_3 = \langle q_1, q_2, \dots, q_m, p_1, p_2, \dots, p_n \rangle$. In both cases R_3 satisfy conditions from definition 1 for assumption $X_1 \cup (\text{body}(r_2))^-$.

Since rules in R3 are from program Q we have $\text{head}(r_1) \in \text{Cn}_{\ll_Q}(X_1 \cup (\text{body}(r_2))^-)$.

R3 $Z_2 = \emptyset$ hence $\text{pos}(X_1 \cup W) \cap Z_2 = \emptyset$. We have $Y_1 \subseteq \text{Cn}_{\ll_P}(X_1)$. So for every $y \in Y_1$ there is a sequence R of rules that satisfy Definition 1 for assumption X_1 . Same sequence satisfy definition 1 for superset assumption $X_1 \cup W$. Hence $y \in \text{Cn}_{\ll_P}(X_1 \cup W)$ and $Y_1 \subseteq \text{Cn}_{\ll_P}(X_1 \cup W)$.

R2 $Z_3 = \emptyset$ hence $\text{pos}(X_1 \cup X_2) \cap Z_3 = \emptyset$. We have $Y_1 \subseteq \text{Cn}_{\ll_P}(X_1)$ hence $Y_1 \subseteq \text{Cn}_{\ll_P}(X_1 \cup X_2)$. We also have $Y_2 \subseteq \text{Cn}_{\ll_P}(X_2)$ hence $Y_2 \subseteq \text{Cn}_{\ll_P}(X_1 \cup X_2)$. Therefore $Y_1 \cup Y_2 \subseteq \text{Cn}_{\ll_P}(X_1 \cup X_2)$.

A *derivation* of an argumentation structure \mathcal{A} (w.r.t. P) is a sequence $\langle \mathcal{A}_1, \mathcal{A}_2, \dots, \mathcal{A}_k \rangle$ of argumentation structures (w.r.t. P) such that \mathcal{A}_1 is a basic argumentation structure, $\mathcal{A} = \mathcal{A}_k$ and each \mathcal{A}_i , $1 < i \leq k$, is either a basic argumentation structure or it is obtained by R1 or R2 or R3 from preceding argumentation structures.

4 Attacks

Our approach to preferred answer sets is based on a solution of conflicts between argumentation structures. We distinguish three steps towards that goal. *Contradictions* between argumentation structures represent the elementary step. Rule preference and contradiction between basic argumentation structures are used to form an attack relation. Consider two basic argumentation structures \mathcal{A}_1 and \mathcal{A}_2 . If \mathcal{A}_1 contradicts \mathcal{A}_2 and corresponds to a more preferred rule, then it *attacks* \mathcal{A}_2 . Attacks are propagated to other argumentation structures using derivation rules. Attacks between argumentation structures depend on how argumentation structures are derived. Hence, we need a more context-independent notion and we define a relation of *blocking* between argumentation structures. The complement of blocking (warranting) is used in the definition of preferred argumentation structures.

Definition 8 Consider argumentation structures $\mathcal{A} = \langle Y_1 \leftrightarrow X_1; Z_1 \rangle$ and $\mathcal{B} = \langle Y_2 \leftrightarrow X_2; Z_2 \rangle$.

If there is a literal $L \in Y_1$ such that $\text{not } L \in X_2$, it is said that the argument X_1 *contradicts* the argument X_2 and the argumentation structure \mathcal{A} *contradicts* the argumentation structure \mathcal{B} .

It is said that X_1 is a *counterargument* to X_2 . \square

The basic argumentation structures corresponding to the facts of the given program are not contradicted.

Let $r_1 = a \leftarrow$ be a fact and $\text{not } a \in (\text{body}(r_2))^-$. Then any $W \subseteq \text{Def}$ s.t. $(\text{body}(r_2))^- \subseteq W$ is not self-consistent and, therefore, it is not an argument.

Example 9 In Example 6, \mathcal{A}_1 contradicts \mathcal{A}_2 and \mathcal{A}_2 contradicts \mathcal{A}_1 .

Only some counterarguments are interesting: the rule r_1 is more preferred than the rule r_2 , therefore the counterargument of \mathcal{A}_2 against \mathcal{A}_1 should not be “effectual”. We are going to introduce a notion of *attack* in order to denote “effectual” counterarguments. \square

Similarly as for the case of argumentation structures, the basic attacks are defined first. A terminological convention: if \mathcal{A}_1 attacks \mathcal{A}_2 , it is said that the pair $(\mathcal{A}_1, \mathcal{A}_2)$ is an attack.

Definition 10 Let $r_2 \prec r_1$ and let $\mathcal{A}_1 = \langle \{\text{head}(r_1)\} \leftrightarrow (\text{body}(r_1))^-; (\text{body}(r_1))^+ \rangle$ contradicts $\mathcal{A}_2 = \langle \{\text{head}(r_2)\} \leftrightarrow (\text{body}(r_2))^-; (\text{body}(r_2))^+ \rangle$.

Then \mathcal{A}_1 *attacks* \mathcal{A}_2 and it is said that this attack is *basic*. \square

Attacks between argumentation structures “inherited” (propagated) from basic attacks are defined in terms of derivation rules. The rules of that inheritance are motivated and defined below.

Example 11 Let us continue with Example 6.

Consider the basic argumentation structures $\mathcal{A}_1 = \langle \{b\} \leftrightarrow \{\text{not } \neg b\}; \{a\} \rangle$, $\mathcal{A}_2 = \langle \{-b\} \leftrightarrow \{\text{not } b\} \rangle$, $\mathcal{A}_3 = \langle \{a\} \leftrightarrow \{\text{not } \neg a\} \rangle$ and the derived argumentation structures $\mathcal{A}_4 = \langle \{b\} \leftrightarrow \{\text{not } \neg b, \text{not } \neg a\} \rangle$, $\mathcal{A}_5 = \langle \{b, a\} \leftrightarrow \{\text{not } \neg b, \text{not } \neg a\} \rangle$, $\mathcal{A}_6 = \langle \{-b, a\} \leftrightarrow \{\text{not } b, \text{not } \neg a\} \rangle$.

$(\mathcal{A}_1, \mathcal{A}_2)$ is the only basic attack (the more preferred \mathcal{A}_1 attacks the less preferred \mathcal{A}_2).

Derivations of the attacks of $(\mathcal{A}_4, \mathcal{A}_2)$ and $(\mathcal{A}_5, \mathcal{A}_2)$ could be motivated as follows. \mathcal{A}_4 is derived from \mathcal{A}_1 and \mathcal{A}_3 using R1, the attack of \mathcal{A}_1 against \mathcal{A}_2 should be propagated to the attack $(\mathcal{A}_4, \mathcal{A}_2)$. Note that \mathcal{A}_3 is not attacked.

Now, \mathcal{A}_5 is derived from \mathcal{A}_3 and \mathcal{A}_4 . Again, the attack of \mathcal{A}_4 against \mathcal{A}_2 should be inherited by $(\mathcal{A}_5, \mathcal{A}_2)$.

Similarly, \mathcal{A}_6 is derived from attacked \mathcal{A}_2 . The attacks against \mathcal{A}_2 are transferred to the attacks against \mathcal{A}_6 . The attack $(\mathcal{A}_5, \mathcal{A}_6)$ is a crucial one, a selection of preferred answer set is based on it; compare with Example 19.

On the contrary, \mathcal{A}_2 contradicts \mathcal{A}_4 and \mathcal{A}_5 , but it is based on a less preferred rule, hence those contradictions are not considered as attacks. \square

First we define two rules, Q1 and Q2, which specify inheritance of attacks “via unfolding” - use of the rule R1. Second, two rules Q3 and Q4 derive attacks when the attacking or attacked side is joined with another argumentation structure - use of the rule R2. Finally, rules Q5 and Q6 derive attacks, if attacking or attacked side is extended by an assumption - use of the rule R3. Some asymmetries between pairs Q1, Q2 and Q3, Q4 will be discussed below, see Example 22.

Definition 12 (Attack derivation rules) Basic attacks are attacks.

Q1 Let $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3$ be argumentation structures such that:

- \mathcal{A}_1 attacks \mathcal{A}_2 ,
- \mathcal{A}_3 does not attack \mathcal{A}_1 , and
- $u(\mathcal{A}_2, \mathcal{A}_3)$ is argumentation structure.

Then \mathcal{A}_1 attacks $u(\mathcal{A}_2, \mathcal{A}_3)$.

Q2 Let $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3$ be argumentation structures such that:

- \mathcal{A}_1 attacks \mathcal{A}_2 ,
- \mathcal{A}_3 is not attacked, and

– $u(\mathcal{A}_1, \mathcal{A}_3)$ is argumentation structure.

Then $u(\mathcal{A}_1, \mathcal{A}_3)$ attacks \mathcal{A}_2 .

Q3 Let \mathcal{A}_1 and \mathcal{A}_3 be argumentation structures of the form $\langle X \leftrightarrow Y \rangle$ and \mathcal{A}_2 be an argumentation structure. Suppose that:

- \mathcal{A}_1 attacks \mathcal{A}_2 ,
- \mathcal{A}_3 is not attacked and
- $\mathcal{A}_1 \cup \mathcal{A}_3$ is argumentation structure.

Then $\mathcal{A}_1 \cup \mathcal{A}_3$ attacks \mathcal{A}_2 .

Q4 Let \mathcal{A}_1 be an argumentation structure and $\mathcal{A}_2, \mathcal{A}_3$ be argumentation structures of the form $\langle X \leftrightarrow Y \rangle$ such that:

- \mathcal{A}_1 attacks \mathcal{A}_2 ,
- \mathcal{A}_3 does not attack \mathcal{A}_1 , and
- $\mathcal{A}_2 \cup \mathcal{A}_3$ is argumentation structure.

Then \mathcal{A}_1 attacks $\mathcal{A}_2 \cup \mathcal{A}_3$.

Q5 Let $\mathcal{A}_1 = \langle X \leftrightarrow Y \rangle$ and \mathcal{A}_2 be argumentation structures. Let $W \subseteq Def$. Suppose that:

- \mathcal{A}_1 attacks \mathcal{A}_2 , and
- $\mathcal{A}_1 \cup W = \langle X \leftrightarrow Y \cup W \rangle$ is argumentation structure.

Then $\mathcal{A}_1 \cup W$ attacks \mathcal{A}_2 .

Q6 Let \mathcal{A}_1 and $\mathcal{A}_2 = \langle X \leftrightarrow Y \rangle$ be argumentation structures. Let $W \subseteq Def$. Suppose that:

- \mathcal{A}_1 attacks \mathcal{A}_2 , and
- $\langle X \leftrightarrow Y \cup W \rangle = \mathcal{A}_2 \cup W$ is argumentation structure.

Then \mathcal{A}_1 attacks $\mathcal{A}_2 \cup W$.

There are no other attacks except those specified above. \square

Definition 13 A *derivation of an attack* \mathcal{X} is a sequence $\mathcal{X}_1, \dots, \mathcal{X}_k$, where $\mathcal{X} = \mathcal{X}_k$, each \mathcal{X}_i is an attack (a pair of attacking and attacked argumentation structures), \mathcal{X}_1 is a basic attack and each $\mathcal{X}_i, 1 < i \leq k$ is either a basic attack or it is derived from the previous attacks using rules Q1, Q2, Q3, Q4, Q5, Q6.

Derivations of argumentation structures and of attacks blend together. Example 15 shows that a pair of argumentation structures $(\mathcal{B}, \mathcal{A})$ is an attack w.r.t. a derivation, but it is not an attack w.r.t. another derivation. Let us start with a definition.

Definition 14 Let a program P and an answer set S be given. Let be $R = \{r \in P \mid body(r) \subseteq S\}$. It is said that R is the set of all *generating rules* of S^+ . \square

Example 15 Let P be

| | |
|-------|-----------------------|
| r_1 | $a \leftarrow not\ b$ |
| r_2 | $b \leftarrow not\ a$ |
| r_3 | $a \leftarrow not\ c$ |
| r_4 | $c \leftarrow b$. |

$\leftarrow = \{(r_1, r_2)\}$.

There are two answer sets of P : $S_1 = \{a\}$ and $S_2 = \{b, c\}$. The corresponding argumentation structures are $\mathcal{A} = \langle \{a\} \leftrightarrow \{not\ b, not\ c\} \rangle$ and $\mathcal{B} = \langle \{b, c\} \leftrightarrow \{not\ a\} \rangle$, respectively.

There are two derivations of \mathcal{A} . Both derivations start from a basic argumentation structure and R3 is used. The first is the sequence $\mathcal{A}_1, \mathcal{A}$ and the second is $\mathcal{A}_3, \mathcal{A}$, where $\mathcal{A}_1 = \langle \{a\} \leftrightarrow \{not\ b\} \rangle$ and $\mathcal{A}_3 = \langle \{a\} \leftrightarrow \{not\ c\} \rangle$.

If the sequence $\mathcal{A}_1, \mathcal{A}$ is considered, an attack against \mathcal{A} is derivable. Let be $\mathcal{A}_2 = \langle \{b\} \leftrightarrow \{not\ a\} \rangle$, $\mathcal{A}_4 = \langle \{c\} \leftrightarrow \emptyset; \{b\} \rangle$. The corresponding attack derivation is as follows:

$(\mathcal{A}_2, \mathcal{A}_1), (u(\mathcal{A}_4, \mathcal{A}_2), \mathcal{A}_1), (\mathcal{B}, \mathcal{A}_1), (\mathcal{B}, \mathcal{A})$, where Q2, Q3 and Q6 are used.

The only basic attack of our example is $(\mathcal{A}_2, \mathcal{A}_1)$. Hence, the second derivation $\mathcal{A}_3, \mathcal{A}$ of \mathcal{A} cannot be attacked.

The derivations of \mathcal{A} correspond to two sets of rules generating S_1 , i.e., $R_1 = \{r_1\}$, and $R_2 = \{r_3\}$. R_1 contains an attacked rule, while R_2 does not contain such a rule. We accept that if there is a set of rules generating an answer set S s.t. no rule is attacked by a rule generating another answer set, then we can consider S as a preferred one.

We transfer this rather credulous approach to derivations of preferred argumentation structures. \square

Definition 16 (Complete arguments) An argumentation structure $\langle Y \leftrightarrow X \rangle$ is called *complete* iff for each literal $L \in Obj$ it holds that $L \in Y$ or *not* $L \in X$. \square

Definition 17 (Warranted and blocked derivations) Let $\sigma = \mathcal{A}_1, \dots, \mathcal{A}_k$ be a derivation of an argumentation structure \mathcal{A} , where $\mathcal{A} = \mathcal{A}_k$.

It is said that σ is *blocked* iff there is a derivation τ of the attack $(\mathcal{B}, \mathcal{A})$, where \mathcal{B} is a complete argumentation structure and each member of τ contains an \mathcal{A}_i as a second component.

A derivation is *warranted* if it is not blocked. \square

Definition 18 (Warranted and blocked argumentation structures) An argumentation structure \mathcal{A} is warranted iff there is a warranted derivation of \mathcal{A} .

\mathcal{A} is blocked iff each derivation of \mathcal{A} is blocked. \square

5 Preferred answer sets

Example 19 Consider our running example, where we have complete argumentation structures $\mathcal{A}_5 = \langle \{b, a\} \leftrightarrow \{\text{not } \neg b, \text{not } \neg a\} \rangle$, $\mathcal{A}_6 = \langle \{-b, a\} \leftrightarrow \{\text{not } \neg a, \text{not } b\} \rangle$.

We will prefer \mathcal{A}_5 over \mathcal{A}_6 . \mathcal{A}_6 is blocked by \mathcal{A}_5 . On the other hand, \mathcal{A}_5 is not blocked.

Consequently, we will consider $\{a, b\}$ as a preferred answer set of the given prioritized logic program.

\square

Definition 20 (Preferred answer set) An argumentation structure is *preferred* iff it is complete and warranted.

$Y \cup X$ is a *preferred answer set* iff $\langle Y \leftrightarrow X \rangle$ is a preferred argumentation structure. \square

Notice that our notion of preferred answer set is rather a credulous one, it is based on the notion of warranted derivation, i.e., at least one derivation of a preferred answer set should not be blocked.

The following example shows that the argumentation structure corresponding to the only answer set of a program is preferred, even if it is attacked (by an argumentation structure which is not complete).

Example 21

| | |
|-------|------------------------------|
| r_1 | $b \leftarrow \text{not } a$ |
| r_2 | $a \leftarrow \text{not } b$ |
| r_3 | $c \leftarrow a$ |
| r_4 | $c \leftarrow \text{not } c$ |

$\prec = \{(r_1, r_2), (r_3, r_4)\}$.

Let the basic argumentation structures be denoted by \mathcal{A}_i , $i = 1, \dots, 4$. $(\mathcal{A}_1, \mathcal{A}_2)$, $(\mathcal{A}_3, \mathcal{A}_4)$ are the basic attacks. \mathcal{A}_1 attacks $\mathcal{A}_5 = \langle \{c\} \leftrightarrow \{\text{not } b\} \rangle$ according to the rule Q1 and \mathcal{A}_1 attacks $\mathcal{A}_6 = \langle \{c, a\} \leftrightarrow \{\text{not } b\} \rangle$ according to the rule Q4.

Remind that according to Definition 17 a derivation can be blocked only by a complete argumentation structure and an argumentation structure is blocked iff each its derivation is blocked. Consequently, the complete argumentation structure \mathcal{A}_6 is not blocked by another complete argumentation structure (there is no such structure) and, consequently, it is the preferred argumentation structure.

We distinguish between attacking and blocking. A blocked argumentation structure is attacked by a *complete* argumentation structure. Preferred argumentation structures are not blocked. \square

Next example explains asymmetries between Q1, Q2 and Q3, Q4. The main idea is as follows. We are more cautious when an attacking argumentation structure is derived (Q2, Q3) and we require that a ‘‘parent’’ of the attacking argumentation structure is not attacked at all. On the other hand, a scheme of derivation rules Q1 and Q4 is as follows: \mathcal{A}_1 attacks \mathcal{A}_2 , \mathcal{A} is a derived argumentation structure from the attacked \mathcal{A}_2 and an argumentation structure \mathcal{A}_3 . In order to derive an attack of \mathcal{A}_1 against \mathcal{A} it is sufficient to assume that \mathcal{A}_3 does not attack \mathcal{A}_1 . However, there are some problems with this design decision.

Example 22 Consider a program P :

$$\begin{array}{ll} r_1 & a_1 \leftarrow \text{not } a_3, \text{not } d_2 \\ r_2 & d_1 \leftarrow \text{not } a_3, \text{not } d_2 \\ r_3 & a_2 \leftarrow \text{not } a_1, \text{not } d_3 \\ r_4 & d_2 \leftarrow \text{not } a_1, \text{not } d_3 \\ r_5 & a_3 \leftarrow \text{not } a_2, \text{not } d_1 \\ r_6 & d_3 \leftarrow \text{not } a_2, \text{not } d_1 \end{array}$$

$\prec = \{(r_1, r_4), (r_3, r_5), (r_6, r_2)\}$.

Notice that a complete argumentation structure $\mathcal{B}_1 = \langle \{a_1, d_1\} \leftarrow \{\text{not } a_3, \text{not } d_2\} \rangle$ is derived from \mathcal{A}_1 corresponding to r_1 and from \mathcal{A}_2 corresponding to r_2 , similarly $\mathcal{B}_2 = \langle \{a_2, d_2\} \leftarrow \{\text{not } a_1, \text{not } d_3\} \rangle$ is derived from \mathcal{A}_3 corresponding to r_3 and from \mathcal{A}_4 corresponding to r_4 and $\mathcal{B}_3 = \langle \{a_3, d_3\} \leftarrow \{\text{not } a_2, \text{not } d_1\} \rangle$ is derived from \mathcal{A}_5 corresponding to r_5 and from \mathcal{A}_6 corresponding to r_6 . $\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3$ are all complete argumentation structures of our example.

Suppose that Q3 does not contain condition that a “parent” of the attacking argumentation structure is not attacked at all. Then we get that each complete argumentation structure is blocked, consequently, no preferred answer set can be selected. But we are extremely interested in a selection of a preferred answer set.

As a consequence, we are too liberal in selecting preferred answer sets: Consider program

$$\begin{array}{ll} r_1 & a \leftarrow \\ r_2 & b \leftarrow \text{not } a \\ r_3 & c \leftarrow \text{not } b \\ r_4 & b \leftarrow \text{not } c \end{array}$$

$\prec = \{(r_2, r_3), (r_3, r_4)\}$.

We get that both $S_1 = \{a, c\}$ and $S_2 = \{a, b\}$ are preferred answer sets, but S_2 is not an intuitive selection. The reason is that both argumentation structures (let us denote them by \mathcal{A}_1 and \mathcal{A}_2) corresponding to S_1 and S_2 , respectively, are attacked and rule Q3 cannot be applied. Hence, each derivation of \mathcal{A}_1 and \mathcal{A}_2 is warranted. A weaker version of Q3 would enable to repair this, however, it is a too high price for us. \square

Theorem 23 *If S is a preferred answer set of (P, \prec, \mathcal{N}) , then S is an answer set of P .*

6 Principles

The principles (partially) specify what it means that an order on answer sets corresponds to the given order on rules. The first two principles below are dependent on [1]. Principle III reproduces an idea of Proposition 6.1 from [1]. The Principles of [1] are originally expressed in an abstract way for the general case of nonmonotonic prioritized defeasible rules. We restrict the discussion (and the wording) of the Principles to the case of logic programs and answer sets.

Let P be a program and $r_1, r_2 \in P$. It is said that r_2 is *attacked* by r_1 (r_1 attacks r_2) iff $\text{not } \text{head}(r_1) \in (\text{body}(r_2))^-$.

Definition 24 Let a prioritized logic program (P, \prec, \mathcal{N}) be given. Let R_1, R_2 be sets of generating rules for some answer sets of P . It is said that R_1 *attacks* R_2 iff there is $r_1 \in R_1, r_2 \in R_2$ such that $r_2 \prec r_1$ and r_1 attacks r_2 .

Definition 25 Let a prioritized logic program (P, \prec, \mathcal{N}) be given. Let R be a set of generating rules of some answer set of P . It is said that R is a *warranted set of generating rules* iff there is no set Q of generating rules of some answer set of P such that Q attacks R . \square

Principle I in its original formulation does not hold for our approach. A terminological remark – words associated to our approach (attack, warranted) are used in presented formulations of Principles. But remind definitions 24 and 25 – the notions are defined for generating sets of rules independently on our approach. We have considered the same principles also in another approach, see [14] and also [13]. It is defined directly on generating sets, and uses neither argumentation structures nor derivation rules.

In all principles below it is assumed that a prioritized logic program (P, \prec, \mathcal{N}) is given.⁶

Principle I Let A_1 and A_2 be two answer sets of the program P . Let $R \subset P$ be a set of rules and $d_1, d_2 \in P \setminus R$ are rules. Let A_1^+, A_2^+ be generated by the rules $R \cup \{d_1\}$ and $R \cup \{d_2\}$, respectively. If d_1 is preferred over d_2 and each set of generating rules of A_2^+ is attacked by a warranted set of generating rules of some answer set of P , then A_2 is not a preferred answer set of (P, \prec, \mathcal{N}) . \square

Our formulation of Principle I differs from the original formulation in [1] – the condition “each set of generating rules of A_2^+ is attacked by a warranted set of generating rules of some answer set of P ” is added because of the credulous stance to warranted derivations accepted in this paper

We do not accept the following principle. See discussion below.

Principle II Let A be a preferred answer set of a prioritized logic program (P, \prec, \mathcal{N}) and r be a rule such that $(\text{body}(r))^+ \not\subseteq A^+$. Then A is a preferred answer set of $(P \cup \{r\}, \prec', \mathcal{N}')$, whenever \prec' agrees with \prec on rules in P and \mathcal{N}' extends \mathcal{N} with the name r . \square

We believe that the possibility to always select a preferred answer set from a non-empty set of standard answer sets is of critical importance.

Principle III Let $\mathcal{B} \neq \emptyset$ be the set of all answer sets of P . Then there is a selection function Σ s.t. $\Sigma(\mathcal{B})$ is the set of all preferred answer sets of (P, \prec, \mathcal{N}) , where $\emptyset \neq \Sigma(\mathcal{B}) \subseteq \mathcal{B}$. \square

It is shown in [1], Proposition 6.1, that Principle II is incompatible with Principle III, if the notion of preferred answer set from [1] is accepted:

Example 26 ([1]) Consider program P , whose single standard answer set is $S = \{b\}$ and the rule (1) is preferred over the rule (2).

$$c \leftarrow \text{not } b \quad (1)$$

$$b \leftarrow \text{not } a \quad (2)$$

S is not a preferred answer set in the framework of [1]. Assume that S , the only standard answer set of P , is selected – according to the Principle III – as the preferred answer set of (P, \prec, \mathcal{N}) .⁷ Let P' be $P \cup \{a \leftarrow c\}$ and $a \leftarrow c$ be preferred over the both rules 1 and 2. P' has two standard answer sets, S and $T = \{a, c\}$. Note that $\{c\} \not\subseteq S^+$. Hence, S should be the preferred answer set of P' according to the Principle II. However, in the framework of [1] the only preferred answer set of $(P', \prec', \mathcal{N}')$ is T . This selection of preferred answer set satisfies clear intuitions – T is generated by the two most preferred rules.

In our approach the complete argumentation structure $\mathcal{A}_4 = \langle \{a, c\} \leftrightarrow \{\text{not } b\} \rangle$ blocks the complete argumentation structure $\mathcal{A}_5 = \langle \{b\} \leftrightarrow \{\text{not } a, \text{not } c\} \rangle$, hence, \mathcal{A}_4 is preferred and $\{a, c\}$ is the preferred answer set.

Principle III is of crucial value according to our view. A more detailed justification of our decision not to accept Principle II is presented in [11]. We mention here only that we select preferred answer sets of P' from a broader variety of possibilities. Consequently, no condition satisfied by a subset of those possibilities should constrain the selection of preferred answer set from the extended set of possibilities. \square

Principle II is not accepted also in [8]. According to [4] descriptive approaches do not satisfy this principle in general.

Principle IV expresses when an answer set is a preferred one. We consider it as an expression of a descriptive approach to preferred answer set specification, as we understand it and accept in this paper.

Principle IV Let S be answer set of P . Suppose that S is generated by a set of rules R . If R is a warranted set of generating rules then S is a preferred answer set. \square

⁶ The original formulation of principles by [1] is as follows.

Principle I. Let B_1 and B_2 be two belief sets of a prioritized theory $(T; \prec)$ generated by the (ground) rules $R \cup d_1$ and $R \cup d_2$, where $d_1, d_2 \notin R$, respectively. If d_1 is preferred over d_2 , then B_2 is not a (maximally) preferred belief set of T .

Principle II. Let B be a preferred belief set of a prioritized theory $(T; \prec)$ and r a (ground) rule such that at least one prerequisite of r is not in B . Then B is a preferred belief set of $(T \cup \{r\}; \prec')$ whenever \prec' agrees with \prec on priorities among rules in T .

⁷ Observe that the only derived complete argumentation structure is $\langle \{b\} \leftrightarrow \{\text{not } a, \text{not } c\} \rangle$. Hence, $\{b\}$ is a preferred answer set of (P, \prec, \mathcal{N}) in our framework.

As regards a choice of principles, we accept the position of [1]: even if somebody does not accept a set of principles for preferential reasoning, those (and similar) principles are still of interest as they may be used for classifying different patterns of reasoning.

7 Discussion

The question whether derivation rules for attacks are sufficient and necessary arises in a natural way. Our only response to the question in this paper is that Principles I, III, IV are satisfied, when we use notions of attack, blocking and warranting introduced in this paper. We proceed to theorems about satisfaction of principles.

Theorem 27 *Principle III is satisfied. Let $\mathcal{P} = (P, \prec, \mathcal{N})$ be a prioritized logic program and $AS(P) \neq \emptyset$. Then there is a preferred answer set of \mathcal{P} .*

Proof. **Case 1** is trivial – if a program P has only one answer set S , then no complete argumentation structure blocks $\langle S^+ \leftrightarrow S^- \rangle$.

Case 2. Let a program P has only two answer sets S_1 and S_2 . Let the corresponding complete argumentation structures be $\mathcal{A}_1 = \langle S_1^+ \leftrightarrow S_1^- \rangle$ and $\mathcal{A}_2 = \langle S_2^+ \leftrightarrow S_2^- \rangle$, respectively. Suppose that \mathcal{A}_1 and \mathcal{A}_2 block each other.

It means that each derivation of both is blocked by the other complete argumentation structure. Consider all derivations of \mathcal{A}_1 (which should be blocked by \mathcal{A}_2). Hence, each derivation σ_i contains an argumentation structure \mathcal{B}_i attacked by \mathcal{A}_2 , i.e., $\mathcal{X} = (\mathcal{A}_2, \mathcal{B}_i)$ is an attack. Each derivation of \mathcal{X} should start from a basic attack and ends with $(\mathcal{A}_2, \mathcal{B}_i)$.

If \mathcal{X} is a basic attack, then the only generating set of rules of S_2 contains only one rule $r = S_2^+ \leftarrow S_2^-$, where $S^+ = \{head(r)\}$. We assume that there is a rule r_1 s.t. $r_1 \prec r$ and *not* $head(r) \in (body(r_1))^-$. On the other hand, \mathcal{A}_1 blocks \mathcal{A}_2 and there is an $r_2 \in P$ which is among the generating rules of S_1 , $r \prec r_2$ and *not* $head(r_2) \in (body(r))^-$.

Notice that $\langle head(r_2) \leftrightarrow (body(r_2))^-; (body(r_2))^+ \rangle$ attacks \mathcal{A}_2 . If $(body(r_2))^+ \neq \emptyset$, then a derivation of attack $(\mathcal{A}_2, \mathcal{A}_1)$ has to use Q1 and $\langle head(r_2) \leftrightarrow (body(r_2))^-; (body(r_2))^+ \rangle$. But Q1 is not applicable – attacking argumentation structure should be not attacked. Similarly, if $(body(r_2))^+ = \emptyset$, Q4 should be used, but Q4 is not applicable because of the same reason.

Assume that \mathcal{X} is not a basic attack. Then there is a basic attack as follows.

Let be $\mathcal{R}_1 = \langle head(r_1) \leftrightarrow (body(r_1))^-; (body(r_1))^+ \rangle$,

$\mathcal{R}_2 = \langle head(r_2) \leftrightarrow (body(r_2))^-; (body(r_2))^+ \rangle$, where

$head(r_2) \in S_2, head(r_1) \in S_1, r_1 \prec r_2$, *not* $head(r_2) \in (body(r_1))^-$ and, consequently, $(\mathcal{R}_2, \mathcal{R}_1)$ is a basic attack.

We will prove that if each derivation of \mathcal{A}_2 is blocked by \mathcal{A}_1 , then it is impossible to derive the attack $(\mathcal{A}_2, \mathcal{A}_1)$.

Let the basic attack $(\mathcal{R}_2, \mathcal{R}_1)$ be given. A derivation of $(\mathcal{A}_2, \mathcal{A}_1)$ from the basic attack should contain rules Q2 or Q3 or Q4 or Q5 in order to proceed from \mathcal{R}_2 to \mathcal{A}_2 (\mathcal{X} is not a basic attack). A derivation of \mathcal{A}_2 using R1, R2, R3 could be reconstructed from this. The derivation is blocked. Therefore, Q2, Q3, Q4 and Q5 are not applicable and the derivation of $(\mathcal{A}_2, \mathcal{A}_1)$ is impossible.

Case 3. Let be $AS(P) = \{S_1, \dots, S_k\}$, $k \geq 3$. Assume that the corresponding complete argumentation structures are $\mathcal{A}_i, i = 1, \dots, k$. Suppose that each of them is blocked. Let us denote the set $\{\mathcal{A}_i \mid i = 1, \dots, k\}$ by O .

Suppose that the set $N \subseteq O$ contains only blocked, but not blocking complete argumentation structures (each $\mathcal{A} \in N$ is blocked and not blocking). If $O \setminus N$ contains only basic argumentation structures then the preference relation \prec is cyclic. Let $M \subseteq O$ be the set of complete argumentation structures which block an argumentation structure and they are not basic argumentation structures.

We will show that there is $\mathcal{A} \in M$ which is not blocked.

We assumed to the contrary that each complete argumentation structure in M is blocked (and blocking simultaneously). If the cardinality of M is 2, Case 2 applies.

Let \mathcal{A}_1 be in M , i.e., \mathcal{A}_1 is a not basic argumentation structure. Assume (without loss of generality) that each derivation of \mathcal{A}_1 is blocked and \mathcal{A}_1 blocks a derivation of \mathcal{A}_3 . We have to show that an attack $(\mathcal{A}_1, \mathcal{A}_3)$ is not derivable.

Consider a derivation of the attack $(\mathcal{A}_1, \mathcal{A}_3)$ and reconstruct the corresponding derivation of \mathcal{A}_1 . Suppose that \mathcal{A}_2 (again without loss of generality) blocks this derivation of \mathcal{A}_1 .

Hence, \mathcal{A}_2 attacks an argumentation structure \mathcal{B} in the derivation of \mathcal{A}_1 . It follows that some argumentation structure in a derivation of \mathcal{A}_2 attacks a basic argumentation structure in the derivation of \mathcal{A}_1 . Consequently, neither rules Q1 and Q4, nor rules Q2 and Q3 are applicable in a derivation of the attack $(\mathcal{A}_1, \mathcal{A}_3)$. Therefore, it is not derivable.

Let $R \in \mathcal{R}$ be attacked by a warranted set Q of generating rules for some answer set of P . Since Q is warranted, there is a warranted derivation of complete argumentation structure \mathcal{B} corresponding to Q . There is also a derivation of complete argumentation structure \mathcal{A} corresponding to R . Q attacks R , so there is a basic argumentation structure \mathcal{C} from the derivation of \mathcal{A} attacked by \mathcal{D} from derivation of \mathcal{B} . Q is warranted, rules Q2 and Q3 are applicable and hence attack $(\mathcal{D}, \mathcal{C})$ is propagated to attack $(\mathcal{B}, \mathcal{C})$. It follows that derivation of \mathcal{A} is blocked.

Theorem 28 *Principle I is satisfied. Let $\mathcal{P} = (P, \prec, \mathcal{N})$ be a prioritized logic program, A_1 and A_2 be two answer sets of P . Let $R \subset P$ be a set of rules and $d_1, d_2 \in P \setminus R$ are rules, d_1 is preferred over d_2 . Let A_1^+, A_2^+ be generated by the rules $R \cup \{d_1\}$ and $R \cup \{d_2\}$, respectively. Assume that each set of generating rules of A_2^+ is attacked by a warranted set of generating rules of some answer set of P .*

Then A_2 is not a preferred answer set of (P, \prec, \mathcal{N}) .

Proof. It is assumed that each set of generating rules of A_2 is attacked by a warranted set of generating rules of some answer set of P . A_2 is not a preferred answer set of (P, \prec, \mathcal{N}) .

Theorem 29 *Principle IV is satisfied. Let $\mathcal{P} = (P, \prec, \mathcal{N})$ be a prioritized logic program and S be an answer set of P . Suppose that S is generated by a warranted set of rules R .*

Then S is a preferred answer set.

Proof. Let R be a set of rules generating an answer set S . If R is a warranted set of generating rules, then there is a derivation of the argumentation structure $\langle S^+ \leftrightarrow S^- \rangle$ which is warranted.

8 Conclusions

An argumentation framework has been constructed, which enables transferring attacks of rules to attacks of argumentation structures and, consequently, to warranted complete argumentation structures. Preferred answer sets correspond to warranted complete argumentation structures. This construction enables a selection of a preferred answer set whenever there is a non-empty set of standard answer sets of a program.

We did not accept the second principle from [1] and we needed to modify their first principle. On the other hand, new principles, which reflect the role of blocking, have been proposed. We stress the role of blocking – in our approach, rules can be blocked by more preferred rules, but the rules which are not blocked are handled in a declarative style.

Among goals for our future research are first of all a thorough analysis of properties and weaknesses of the presented approach (supported by an implementation of the derivation rules) and a detailed comparison to other approaches.

Finally, we have to mention the main differences between the preliminary version [11] and this paper. A more subtle set of attack derivation rules is introduced. A new assumption in Q3 (\mathcal{A}_3 is not attacked) changed the set of attacked derivations and, consequently, our semantics. A new and more adequate notion of warranted and blocked argumentation structure is introduced, which is based on new concepts of warranted and blocked derivations. Consequently, the notion of preferred answer set is changed. A connection of attacks between argumentation structures to different derivations of argumentation structures was not expressed in [11]. More precise and appropriate formulations of Principles IV and I are presented.

Acknowledgements: We are grateful to anonymous referees for very valuable comments and proposals. This paper was supported by the grant 1/0689/10 of VEGA.

References

1. G. Brewka and T. Eiter. Preferred answer sets for extended logic programs. *Artificial Intelligence*, 109(1-2):297–356, 1999.
2. M. Caminada and L. Amgoud. On the evaluation of argumentation formalisms. *Artificial Intelligence*, 171(5-6):286–310, 2007.
3. J. Delgrande, T. Schaub, and H. Tompits. A framework for compiling preferences in logic programs. *Theory and Practice of Logic Programming*, 3(2):129–187, 2003.
4. J. Delgrande, T. Schaub, H. Tompits, and K. Wang. A classification and survey of preference handling approaches in nonmonotonic reasoning. *Computational Intelligence*, 20(2):308–334, 2004.
5. J. P. Delgrande and T. Schaub. Expressing preferences in default logic. *Artificial Intelligence*, 123(1-2):41–87, 2000.
6. Y. Dimopoulos and A. Torres. Graph theoretical structures in logic programs and default theories. *Theor. Comput. Sci.*, 170(1-2):209–244, 1996.
7. A. J. García and G. R. Simari. Defeasible logic programming: An argumentative approach. *TPLP*, 4(1-2):95–138, 2004.
8. C. Sakama and K. Inoue. Prioritized logic programming and its application to commonsense reasoning. *Artificial Intelligence*, 123(1-2):185–222, 2000.
9. T. Schaub and K. Wang. A comparative study of logic programs with preference. In *IJCAI*, pages 597–602, 2001.
10. J. Šefrānek. Rethinking semantics of dynamic logic programs. In *Proc. of the Workshop NMR*, 2006.
11. J. Šefrānek. Preferred answer sets supported by arguments. In *Proc. of the Workshop NMR*, 2008.
12. J. Šefrānek and A. Šimko. Warranted derivation of preferred answer sets, <http://kedrigern.dcs.fmph.uniba.sk/reports/>. Technical Report TR-2011-027, Comenius University, Faculty of Mathematics, Physics, and Informatics, 2011.
13. A. Šimko. Accepting the natural order of rules in a logic program with preferences. In *Proceedings of the Doctoral Consortium at ICLP 2011.*, 2011.
14. A. Šimko. Preferred answer sets - banned generating set approach. In *submitted*, 2011.
15. K. Wang, L. Zhou, and F. Lin. Alternating fixpoint theory for logic programs with priority. In *Computational Logic*, pages 164–178, 2000.