A Well-founded Semantics for Basic Logic Programs with Arbitrary Abstract Constraint Atoms

Yisong Wang
cooperated with Fangzhen Lin, Mingyi Zhang and Jia-Huai You

KRW-2012 @ Guiyang
Logic programs with abstract constraint atoms:

- was firstly proposed by Marek and Truszczynski [MT04];
- has two **answer set** semantics [SPT07];
  - one is by reduction,
  - one is by completion.
- is general enough to capture
  - logic programs with aggregates [FPL11],
  - description logic programs [EW08],
  - disjunctive logic programs,
  - logic programs with nested expressions [LTT99].
Well-founded semantics of logic programs:

- is well-studied for normal logic programs [VRS91];
- has some nice properties, e.g,
  - well-founded atoms are in every answer sets,
  - no unfounded-founded atoms are in answer sets,
  - both are computable in polynomial time.
- has been proposed for various logic programs, e.g,
  - disjunctive logic programs [WZ05],
  - aggregate programs [PDB07],
  - description logic programs [ELIS11],
  - nondisjunctive hybrid MKNF knowledge base [KAH11].

It is unknown for logic programs with constraint atoms

Difficult: constraint atoms may be nonmonotonic!
A logic program with abstract constraint atoms $P$ is a finite set of rules of the form

$$A \leftarrow A_1, \ldots, A_k, \text{not } A_{k+1}, \ldots, \text{not } A_n$$

where $A$ and $A_i$'s are c-atoms of the form $(D, C)$, where

- $D$ is a finite set of atoms,
- $C \subseteq 2^D$ (the powerset of $D$).

Let $A = (D, C)$.

- The complement of $A$, written $\overline{A}$, is $(D, 2^D \setminus C)$.
- $A$ is elementary if it is of the form $(\{a\}, \{\{a\}\})$, written as $a$.

$P$ is basic if, $A$ is elementary for every rule of the form (1) in $P$. 

Let $S, M$ be sets of atoms, $A = (D, C)$.

- $M$ classically satisfies $A$, if $M \cap D \in C$,
- $M$ classically satisfies not $A$, if $M \cap D \notin C$,
- $A$ is monotonic whenever $M$ classically satisfies $A$ implies $M'$ classically satisfies $A$ for every $M' \supseteq M$,
- $S$ conditionally satisfies $A$ wrt $M$, if $S$ classically satisfies $A$ and $I \in C$ for every $I$ with $S \cap D \subseteq I \subseteq M \cap D$, denoted by $S \models_M A$.
- these notions are extended to sets of c-atoms.

Let $P$ be a basic positive program,

$$T_{(P,M)}(S) = \{ \text{Head}(r) | r \in P \text{ such that } S \models_M \text{Body}(r) \}.$$
Let $P$ be a basic program and $M$ a set of atoms.

- The complement of $P$, denoted by $\overline{P}$, is obtained from $P$ by replace $\text{not } A$ with $\overline{A}$.
- The reduct of $P$ wrt $M$, denoted by $P^M$, is obtained from $P$ by
  - removing all rules of the form (1) such that $M \models A_j$ ($k + 1 \leq j \leq n$),
  - eliminating remaining $\text{not } A$.
- $M$ is an c-answer set of $P$ if $M = \text{lfp}(T(\overline{P}, M))$.
- $M$ is an r-answer set of $P$ if $M = \text{lfp}(T(P^M, M))$. 
Let $S, J$ be two sets of atoms s.t $S \cap J = \emptyset$, $A = (A_d, A_c)$.

- **S-prefixed power set** $S \cup J$ denotes $\{S' | S \subseteq S' \subseteq S \cup J\}$.
- $S \cup J$ is maximal in $A$ if $S \cup J \subseteq A_c$ and no $S' \cup J' \subseteq A_c$ such that $S \cup J \subset S' \cup J'$.
- **Abstract representation**: $A^* = (A_d, A_c^*)$ where $A_c^*$ is the set of all maximal prefixed power sets in $A$.

**Example**

[Example 4 of [SPT07]] The aggregate program $P$ consists of

$$p(1); \ p(-1) \leftarrow p(2); \ p(2) \leftarrow \text{SUM}({X|p(X)}) \geq 1.$$  

The abstract representation is

$$\{\{p(1)\} \cup \{p(2)\}, \{p(2)\} \cup \{p(1), p(-1)\} \}.$$
A partial interpretation $I$:

- is a consistent set of literals of the form $a, \text{not } a$,
- $I^+ = \{a | a \in I\}$,
- $I^- = \{a | \text{not } a \in I\}$.

Let $A = (D, C)$ and $I$ a partial interpretation.

- $I$ satisfies $A$, written $I \models A$, if, for some $S \cup J \in C^*$, $S \subseteq I^+$ and $D \setminus (S \cup J) \subseteq I^-$.
- $I$ falsifies $A$, written $I \not\models A$, if, for every $S \cup J \in C^*$, $S \cap I^- \neq \emptyset$ or $D \setminus (S \cup J) \cap I^+ \neq \emptyset$.
- $I$ satisfies (resp. falsifies) not $A$ if $I$ falsifies (resp. satisfies) $A$.
- The two notions are extended to sets of literals and c-literals.
Let $P$ be a basic program and $I$ a partial interpretation. A set of atoms $U$ is an unfounded set of $P$ wrt $I$ iff, for any $a \in U$ and any $r \in P$ with $\text{Head}(r) = a$, either

(c-i) for some $A \in \text{Neg}(r)$, $I \models \text{not } A$ or
(c-ii) for some $A \in \text{Pos}(r)$, for any $S \cup J \in A^*_c$, either $U \cap S \neq \emptyset$ or $I \models S \cup \text{not } (A_d \setminus (S \cup J))$.

**Lemma**

Let $P$ be a basic logic program, $I$ a partial interpretation, and $U_1$ and $U_2$ two sets of atoms. If $U_1$ and $U_2$ are unfounded sets of $P$ wrt $I$ then $U_1 \cup U_2$ is also an unfounded set of $P$ wrt $I$. 
Let $P$ be a basic program and $I$ a partial interpretation. We define the operators $T_P$, $U_P$ and $W_P$ as follows.

- $T_P(I) = \{\text{Head}(r) | r \in P \text{ and } I \text{ satisfies Body}(r)\}$;
- $U_P(I) = \text{the greatest unfounded set of } P \text{ wrt } I$;
- $W_P(I) = T_P(I) \cup \text{not } U_P(I)$.

**Lemma**

The above three operators are monotonic.

**Definition**

The well-founded model of $P$ is $\text{lfp}(W_P)$.

The well-founded model of $P$ in the previous example is $\{p(1)\}$. 
Theorem

Let $P$ be a basic logic program and $I$ a total model of $P$. Then

- $I^+$ is an r-answer set of $P$ iff $W_P(I) = I$.
- $I^+$ is a c-answer set of $P$ iff $W_{\overline{P}}(I) = I$.

Theorem

Let $P$ be a basic logic program. We have

1. $[WFS(P)]^+ \subseteq (\bigcap rAS(P))$,
   $[WFS(P)]^+ \subseteq (\bigcap cAS(P))$.
2. $[WFS(P)]^- \cap (\bigcup rAS(P)) = \emptyset$,
   $[WFS(P)]^- \cap (\bigcup cAS(P)) = \emptyset$. 
Let $A = (D, C)$ and $I$ a partial interpretation. The simplification of $A$ wrt $I$, denoted $R(A, I)$, is the c-atom $(D', C')$ where

- $D' = D \setminus (I^+ \cup I^-)$ and
- $C' = \{ S \setminus I^+ | S \in C \text{ and } S \cap I^- = \emptyset \}$.

The simplification of a basic program $P$ under $WFS(P)$, denoted $R(P)$, is obtained from $P$ by

- eliminating every rule $r$ if either $Head(r) \in [WFS(P)]^+$ or $WFS(P)$ falsifies $Body(r)$,
- removing every c-literal which is satisfied by $WFS(P)$,
- replacing each remaining c-atom $A$ with $R(A, WFS(P))$. 
Example

Let’s recall the basic program:

\[ \{ p(1); \quad p(-1) \leftarrow p(2); \quad p(2) \leftarrow \text{SUM}(\{X | p(X)\}) \geq 1 \} \]

\[ \Downarrow \]

The abstract constraint encoding of the third rule is:

\[ p(2) \leftarrow (A_d, A_c), \text{As WFS}(P) = \{ p(1) \}, \text{it is simplified as:} \]

\[ \Downarrow \]

\[ p(2) \leftarrow (\{p(-1), p(2)\}, \emptyset, \{p(2)\}, \{p(-1), p(2)\}). \]

where \( A_d = \{p(-1), p(1), p(2)\}, \)

\( A_c = \)

\( \{\{p(1)\}, \{p(2)\}, \{p(1), p(2)\}, \{p(2), p(-1)\}, \{p(1), p(-1), p(2)\}\}. \)
Theorem

Let \( P \) be a basic logic program. A set \( M \) of atoms is an r-answer set of \( P \) iff \( X \) is an r-answer set of \( R(P) \) where
\[
X = M \setminus [WFS(P)]^+.
\]

Corollary

Let \( P \) be a basic logic program. A set \( M \) of atoms is a c-answer set of \( P \) iff \( X \) is a c-answer set of \( R(\overline{P}) \) where
\[
X = M \setminus [WFS(\overline{P})]^+.
\]
Example

Let $P$ be an aggregate program consisting of:

$$p(0) \leftarrow \text{not} \ \text{COUNT} (\{\langle 0 : p(0) \rangle, \langle 1 : p(1) \rangle \} \neq 1$$

If we take default negation as classical negation then the corresponding aggregate program $P'$ consists of

$$p(0) \leftarrow \neg \text{COUNT} (\{\langle 0 : p(0) \rangle, \langle 1 : p(1) \rangle \} \neq 1$$

In terms of [PDB07], we have that the ultimate well-founded model of $P'$ is $(\emptyset, \emptyset)$ which corresponds to the partial interpretation $\{\text{not } p(0), \text{not } p(1)\}$. 
According to [SPT07], while we take $P$ as the logic program with abstract constraint atoms $P''$ consisting of

$$p(0) \leftarrow \text{not} \left(\{p(0), p(1)\}, \emptyset, \{p(0), p(1)\}\right),$$

the well-founded model $P''$ is $\{\text{not} p(1)\}$. One can verify that the well-founded model of $\overline{P''}$ is $\{\text{not} p(0), \text{not} p(1)\}$. 
Example

Let $P$ be the aggregate program consisting of

\[ p(0) \leftarrow \text{not } \text{COUNT} \left( \{ Y : p(Y) \} \right) \leq 0. \]

$P$ corresponds to the basic logic program $P'$:

\[ \{ p(0) \leftarrow \text{not} \left( \{ p(0) \}, \{ \emptyset \} \right) \}. \]

Let $I = \emptyset$ and $X = \{ p(0) \}$. $X$ is an unfounded set for $P$ wrt $I$ in terms of [Fab05], $\emptyset$ is the unique unfounded set for $P$ wrt $I$, and the well-founded model of $P'$ is $\emptyset$. 

\[ Y \text{isong Wang cooperated with Fangzhen Lin, Mingyi Zhang and Jia-Huai You} \]
Example

Let $\mathcal{K} = (O, P)$ where $O = \emptyset$ and

$$P = \{ p(a) \leftarrow DL[S \odot p, S \ominus p; \neg S](a) \}. $$

The corresponding basic logic program is

$$\{ p(a) \leftarrow (\{ p(a) \}, \{ \emptyset, \{ \{ p(a) \} \} ) \}$$

whose well-founded model is $\{ p(a) \}$. According to [ELIS11], the dl-program is translated into $\mathcal{K}' = (O, P')$ where $P'$ consists of

$$p(a) \leftarrow DL[S \odot p, S \ominus p'; \neg S](a),$$

$$p'(a) \leftarrow \text{not } DL[S' \oplus p, S'](a),$$

whose well-founded model is $\emptyset$ according to [ELIS11].
We propose a well-founded semantics for basic logic programs, which shares the similar properties of normal logic programs, e.g.,

- is computable in polynomial time,
- respects the answer set semantics,
- can be used to simplify such logic programs,
- can be applied to other logic programs.

Future work:
- dealing with disjunction?
Thank you!
Thomas Eiter, Thomas Lukasiewicz, Giovambattista Ianni, and Roman Schindlauer.
Well-founded semantics for description logic programs in the semantic web.

Thomas Eiter and Kewen Wang.
Semantic forgetting in answer set programming.

Wolfgang Faber.
Unfounded sets for disjunctive logic programs with arbitrary aggregates.
In Logic Programming and Nonmonotonic Reasoning, 8th International Conference, LPNMR 2005, volume 3662 of


Tran Cao Son, Enrico Pontelli, and Phan Huy Tu. Answer sets for logic programs with arbitrary abstract constraint atoms.
