ASP Propagation

Torsten Grote

Universität Potsdam - Institut für Informatik

May 10, 2007

Answerset Computation

$\texttt{answerset}(T, F, \Pi)$

begin
 $\mathcal{A} = atom(\Pi)$
 $(T, F) \leftarrow \texttt{propagation}(T, F, \Pi)$
 if $(T \cap F) \neq \emptyset$ **then** fail
 if $(T \cup F) = \mathcal{A}$ **then return**(T)
 $A \leftarrow \texttt{select}(\mathcal{A} \setminus (T \cup F), \Pi)$
 $\texttt{answerset}(T \cup \{A\}, F)$
 $\texttt{answerset}(T, F \cup \{A\})$
end

Before 2006

- no transparent approach
- publications deal only with implementation details
- lack of formal framework for characterizing ASP computation
- very difficult to
  - understand the functioning of solvers
  - compare different solvers
  - reveal their shortcomings
  - come up with improvements
- SAT approach doesn't have this deficiency

After 2006

- two proposals by Gebser and Schaub
- view answer set computations as derivations in an inference system
- don't look so much at logical equivalences in terms of models

First Proposal

- characterizing ASP inferences by unit propagation based on concepts from Constraint Processing
- nogoods already used in SAT solvers, can be used in ASP, too

Second Proposal

- using Tableau Calculi as inference system
- easy to understand concept known from proof theory
- declarative and fine-grained instrument for ASP operations and strategies

Tableau Calculi

General properties

- a tableau calculus is a set of tableau rules
- tableaux in most cases a binary tree
- root is a formula which is to proof
- it is extended to branches only by application of rules
- a branch is contradictory if it contains a and $\neg a$ for an $a \in atoms$
- a formula is invalid if every branch of its tableaux is contradictory

Rules for classical logic

$$(\wedge)\frac{a \wedge b}{\begin{matrix} a \\ b \end{matrix}} \qquad (\vee)\frac{a \vee b}{a \mid b} \qquad (\Rightarrow)\frac{a \Rightarrow b}{\neg a \mid b} \qquad (\neg 1)\frac{a}{\neg\neg a} \qquad (\neg 2)\frac{\neg\neg a}{a}$$

Properties of ASP tableaux

- root consists of all rules of logic program Π and all atoms
- nodes are signed propositions preceded by **T** for *true* and **F** for *false*
- branches are extended by special ASP tableau rules
- branch is complete if it is contradictory or closed under all rules
- every branch corresponds to a pair (Π, A) where A is an assignment
- a non contradictory complete branch contains an answer set
- an entire tableaux represents traversal of the search space

Atom-oriented tableau rules 1

$$(FTA)\ \frac{p \leftarrow l_1, \dots, l_n \quad \mathbf{T}\{l_1, \dots, l_n\}}{\mathbf{T}p} \qquad (BFA)\ \frac{p \leftarrow l_1, \dots, l_n \quad \mathbf{F}p}{\mathbf{F}\{l_1, \dots, l_n\}}$$

Atom-oriented nogoods 1

$$\Delta(p) = \{\{\mathbf{F}p, \mathbf{T}\beta_1\}, \dots, \{\mathbf{F}p, \mathbf{T}\beta_m\}\} \qquad body(p) = \{\beta_1, \dots, \beta_m\}$$

Atom-oriented tableau rules 2

$$(FFA)\ \frac{\mathbf{F}\beta_1, \dots, \mathbf{F}\beta_m}{\mathbf{F}p} \qquad (BTA)\ \frac{\mathbf{T}p \quad \mathbf{F}\beta_1, \dots, \mathbf{F}\beta_{i-1}, \mathbf{F}\beta_{i+1}, \dots, \mathbf{F}\beta_m}{\mathbf{T}\beta_i}$$

Atom-oriented nogoods 2

$$\delta(p) = \{\mathbf{T}p, \mathbf{F}\beta_1, \dots, \mathbf{F}\beta_m\}$$

Body-oriented tableau rules 1

$$(FTB)\,\frac{\begin{array}{c}p \leftarrow l_1, \dots, l_n\\ \mathbf{T}l_1, \dots, \mathbf{T}l_n\end{array}}{\mathbf{T}\{l_1, \dots, l_n\}} \qquad (BFB)\,\frac{\begin{array}{c}\mathbf{F}\{l_1, \dots, l_i, \dots, l_n\}\\ \mathbf{T}l_1, \dots, \mathbf{T}l_{i-1}, \mathbf{T}l_{i+1}, \dots, \mathbf{T}l_n\end{array}}{\mathbf{F}l_i}$$

Body-oriented nogoods 1

$$\delta(p) = \{\mathbf{F}\beta, \mathbf{T}l_1, \dots, \mathbf{T}l_n\} \qquad (\neg a \in \beta) \wedge (a = \bot) \Rightarrow \mathbf{T}l_a$$

Body-oriented tableau rules 2

$$(FFB)\,\frac{\begin{array}{c}p \leftarrow l_1, \dots, l_i, \dots, l_n\\ \mathbf{F}l_i\end{array}}{\mathbf{F}\{l_1, \dots, l_i, \dots, l_n\}} \qquad (BTB)\,\frac{\mathbf{T}\{l_1, \dots, l_i, \dots, l_n\}}{\mathbf{T}l_i}$$

Body-oriented nogoods 2

$$\Delta(p) = \{\{\mathbf{T}\beta, \mathbf{F}l_1\}, \dots, \{\mathbf{T}\beta, \mathbf{F}l_n\}\} \qquad (\neg a \in \beta) \wedge (a = \top) \Rightarrow \mathbf{F}l_a$$

The Cut rule

$$(Cut[X]) \ \frac{}{\mathbf{T}a \mid \mathbf{F}a} \ (a \in X \subseteq atom(\Pi) \cup body(\Pi))$$

Clark's Completion

- unit propagation on a program's completion is described by the previous rules
- $\mathcal{T}_{comp} = \{FTA, BFA, FFA, BTA, FTB, BFB, FFB, BTB\}$

Fitting's operator

- Φ applies only forward propagation and can be viewed as a set of tableau rules, too
- $\mathcal{T}_{\Phi} = \{FTA, FFA, FTB, FFB\}$

Well-founded rules

$$(WFN)\frac{\mathbf{F}\beta_1,\ldots,\mathbf{F}\beta_m}{\mathbf{F}p} \qquad (WFJ)\frac{\mathbf{T}p \quad \mathbf{F}\beta_1,\ldots,\mathbf{F}\beta_{i-1},\mathbf{F}\beta_{i+1},\ldots,\mathbf{F}\beta_m}{\mathbf{T}\beta_i}$$

$$(p \in GUS(\{r \in \Pi | body(r) \notin \{\beta_1,\ldots,\beta_m\}\}))$$

Loop rules

$$(FL)\frac{\mathbf{F}\beta_1,\ldots,\mathbf{F}\beta_m}{\mathbf{F}p} \qquad (BL)\frac{\mathbf{T}p \quad \mathbf{F}\beta_1,\ldots,\mathbf{F}\beta_{i-1},\mathbf{F}\beta_{i+1},\ldots,\mathbf{F}\beta_m}{\mathbf{T}\beta_i}$$

$$(p \in L, L \in Loop(\Pi), EB(L) = \{\beta_1,\ldots,\beta_m\})$$

Characterizing ASP solvers

Well-founded operator

- Ω is like Φ except that it replaces negation of single atoms with negation of unfounded sets
- $\mathcal{T}_{\Omega} = \{FTA, \mathbf{WFN}, FTB, FFB\}$

Existing ASP-Solvers

$$
\begin{aligned}
\mathcal{T}_{smodels} &= \mathcal{T}_{comp} \cup \{WFN, Cut[atom(\Pi)]\} \\
\mathcal{T}_{noMoRe} &= \mathcal{T}_{comp} \cup \{WFN, Cut[body(\Pi)]\} \\
\mathcal{T}_{nomore++} &= \mathcal{T}_{comp} \cup \{WFN, Cut[atom(\Pi) \cup body(\Pi)]\} \\
\mathcal{T}_{cmodels-1} &= \mathcal{T}_{comp} \cup \{Cut[atom(\Pi) \cup body(\Pi)]\} \\
\mathcal{T}_{assat} &= \mathcal{T}_{comp} \cup \{FL, Cut[atom(\Pi) \cup body(\Pi)]\}
\end{aligned}
$$

Questions?