FAKULTÄT FÜR !NFORMATIK Faculty of Informatics

kbs Knowledge-Based Systems Group

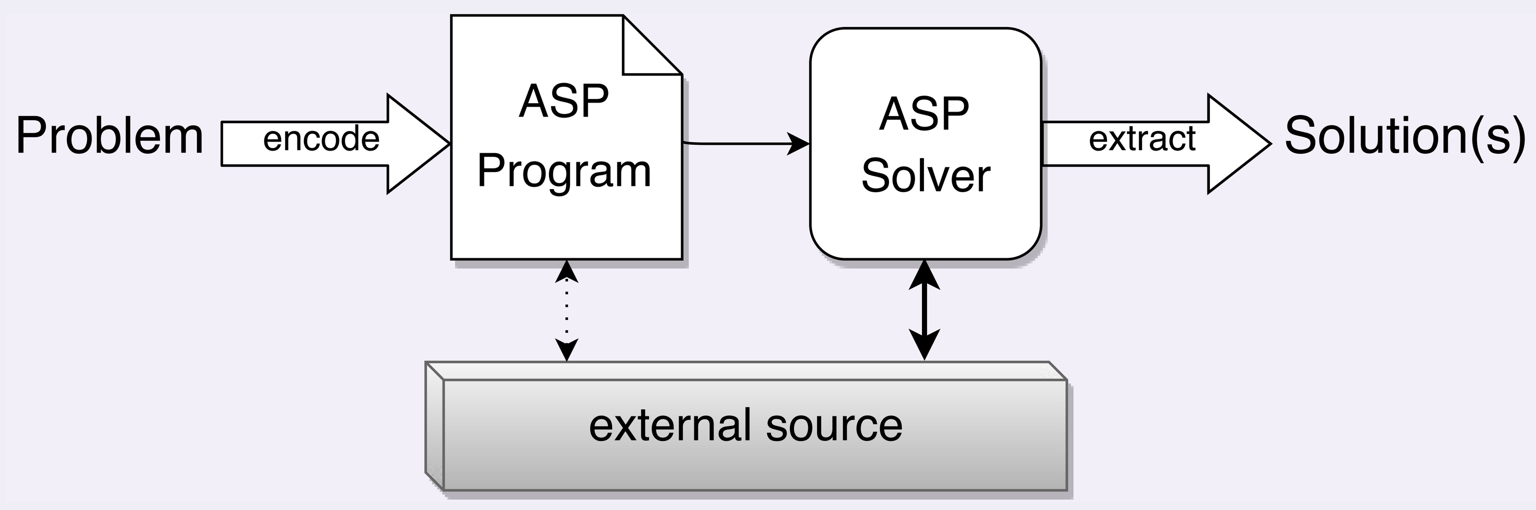
Twenty-fifth International Joint Conference on Artificial Intelligence (IJCAI-16), New York, USA, July 9-15, 2016

Exploiting Partial Assignments for Efficient Evaluation of Answer Set Programs with External Source Access

Thomas Eiter Tobias Kaminski Christoph Redl Antonius Weinzierl

{eiter,kaminski,redl,weinzierl}@tuwien.ac.at

1. Motivation



- HEX-programs extend ASP by external sources
- Similar to SMT for SAT, but external source is black box
- Rule bodies may contain external atoms of the form $\&g[\mathbf{p}](\mathbf{c})$
  - g is an external predicate name
  - $\mathbf{p} = p_1, \dots, p_k$ are input predicate names or constants
  - $\mathbf{c} = c_1, \dots, c_l$ are output terms

Semantics: Boolean oracle function $f_{\&g}$ s.t. $\&g[\mathbf{p}](\mathbf{c})$ is true iff $f_{\&g}(\mathbf{A}, \mathbf{p}, \mathbf{c})$, w.r.t. assignment $\mathbf{A}$

- Basic evaluation:
  1. replace $\&g[\mathbf{p}](\mathbf{c})$ by $e_{\&g[\mathbf{p}]}(\mathbf{c})$, add $e_{\&g[\mathbf{p}]}(\mathbf{c}) \lor ne_{\&g[\mathbf{p}]}(\mathbf{c})$
  2. run CDNL solver (e.g. Clasp)
  3. check guess for $\&g[\mathbf{p}](\mathbf{c})$ when $\mathbf{p}$ decided
  4. learn io-nogoods when evaluating external atoms to avoid wrong guesses

Challenge: External sources cannot guide the solver effectively, they are **black boxes** evaluated under **complete** assignments!

Example

Oracle function for checking if size of predicate extension $\geq n$:

$$f_{\&geq}(\mathbf{A}, p, n) = \begin{cases} \mathbf{T} & \text{if } |\{\mathbf{T}p(x,y) \in \mathbf{A}\}| \geq n \\ \mathbf{F} & \text{otherwise} \end{cases}$$

HEX-program:

$$vertex(a). \quad vertex(b).$$
$$a(X,Y) \lor na(X,Y) \leftarrow vertex(X), vertex(Y).$$
$$\leftarrow e_{\&geq[a,2]}().$$
$$e_{\&geq[a,2]}() \lor ne_{\&geq[a,2]}() \leftarrow$$

$$\mathbf{A}: \{\mathbf{F}e_{\&geq[a,2]}(), \mathbf{T}a(a,b), \mathbf{F}a(b,a), \mathbf{T}a(a,a), \mathbf{F}a(b,b)\}$$
$$\text{Learn}: \{\mathbf{F}e_{\&geq[a,2]}(), \mathbf{T}a(a,b), \mathbf{F}a(b,a), \mathbf{T}a(a,a), \mathbf{F}a(b,b)\}$$

2. Main Contributions

Extension from two-valued to three-valued assignments, enables:

1. Early evaluation of external sources
2. External theory learning producing smaller nogoods
3. Nogood minimization techniques

New techniques applicable by user without expert knowledge
Benchmarks show effectiveness of techniques

References

Thomas Eiter, Giovambattista Ianni, Roman Schindlauer, and Hans Tompits: *"A Uniform Integration of Higher- Order Reasoning and External Evaluations in Answer-Set Programming"*, IJCAI, 2005.

Thomas Eiter, Michael Fink, Thomas Krennwallner, and Christoph Redl: *"Conflict-driven ASP solving with external sources"*, TPLP, 2012.

Robert Nieuwenhuis, Albert Oliveras, and Cesare Tinelli: *"Solving SAT and SAT Modulo Theories: From an abstract Davis–Putnam–Logemann–Loveland procedure to DPLL(T)"*, ACM Journal, 2006.

3. Extension to Partial Assignments

Partial assignment over atoms $\mathcal{A}$ is set $\mathbf{A}$ of signed literals $\mathbf{T}a$, $\mathbf{F}a$ and $\mathbf{U}a$ s.t. for all $a \in \mathcal{A}$ exactly one of $\mathbf{T}a \in \mathbf{A}$, $\mathbf{F}a \in \mathbf{A}$ or $\mathbf{U}a \in \mathbf{A}$ holds.

A *three-valued oracle function* $f_{\&g}$ for $\&g[\mathbf{p}](\mathbf{c})$ is a function such that $f_{\&g}(\mathbf{A}, \mathbf{p}, \mathbf{c}) \in \{\mathbf{T}, \mathbf{F}, \mathbf{U}\}$ for a partial assignment $\mathbf{A}$ and all possible values of $\mathbf{p}$ and $\mathbf{c}$.

A three-valued oracle function $f_{\&g}$ is *assignment-monotonic* if $f_{\&g}(\mathbf{A}, \mathbf{p}, \mathbf{c}) = X$, $X \in \{\mathbf{T}, \mathbf{F}\}$, implies $f_{\&g}(\mathbf{A}', \mathbf{p}, \mathbf{c}) = X$ for all assignments $\mathbf{A}' \succeq \mathbf{A}$.

4. Nogood Learning with Partial Assignments

Nogood learning: Nogood only containing the decided part of a partial assignment learned as soon as oracle function evaluates to $\mathbf{T}$ or $\mathbf{F}$
Partial nogoods often significantly smaller

Nogood minimization: Given an io-nogood N, its minimized nogoods are

$$minimize(N) = \{N' \subseteq N \mid N' \text{ is an io-nogood}, f_{\&g}(N'', \mathbf{p}, \mathbf{c}) = \mathbf{U} \text{ for all } N'' \subsetneq N'_I\}.$$

Nogoods with same input part can be minimized simultaneously

Example

Extension to three-valued oracle function:

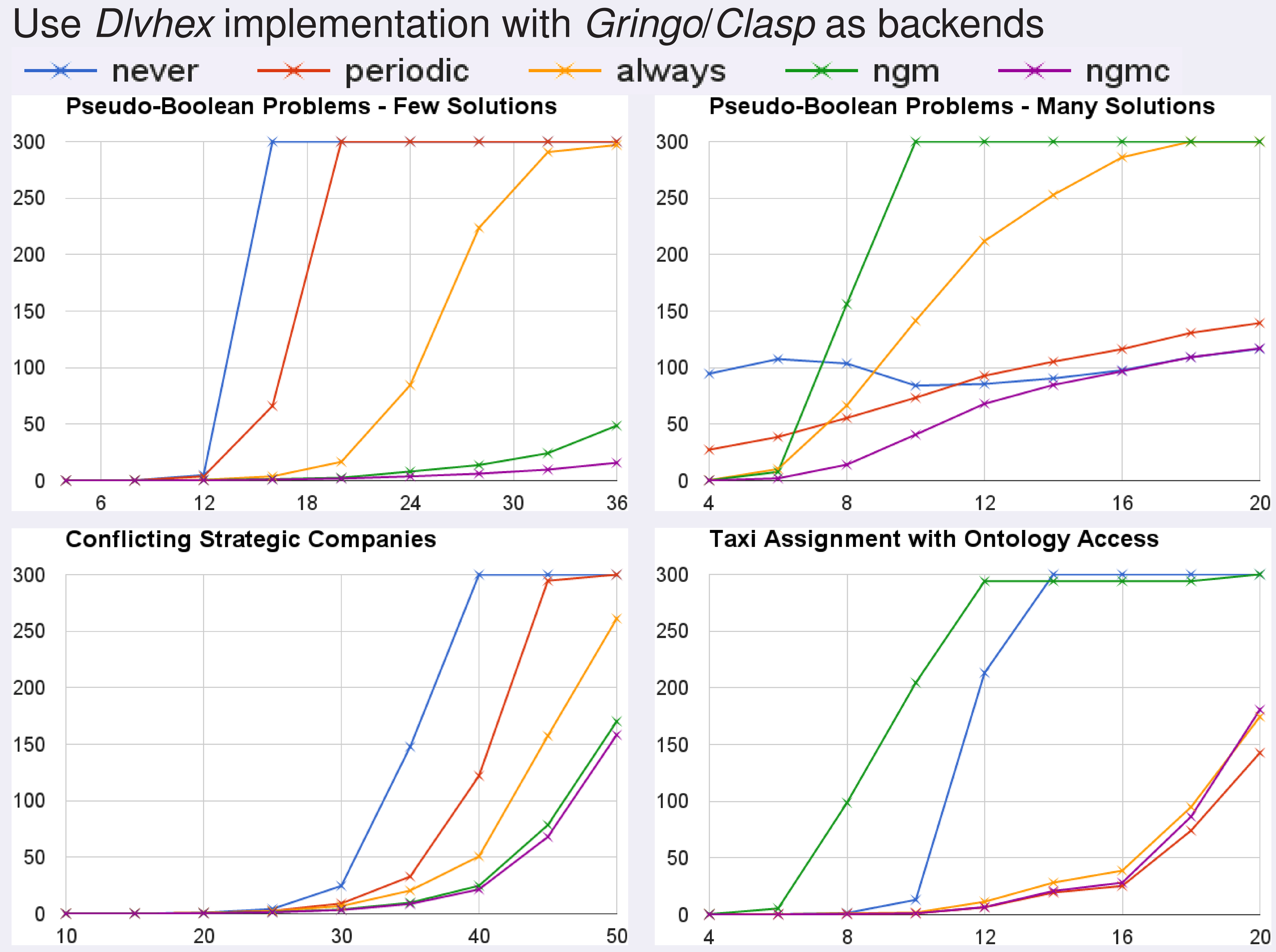
$$f_{\&geq}(\mathbf{A}, arc, n) = \begin{cases} \mathbf{T} & \text{if } |\{\mathbf{T}arc(X,Y) \in \mathbf{A}\}| \geq n \\ \mathbf{U} & \text{if } |\{\mathbf{T}arc(X,Y), \mathbf{U}arc(X,Y) \in \mathbf{A}\}| \geq n \\ \mathbf{F} & \text{otherwise} \end{cases}$$

External source can already be checked under partial assignment:

$$\mathbf{A}: \{\mathbf{F}e_{\&geq[a,2]}(), \mathbf{T}a(a,b), \mathbf{F}a(b,a), \mathbf{T}a(a,a), \mathbf{U}a(b,b)\}$$
$$\text{Learn}: \{\mathbf{F}e_{\&geq[a,2]}(), \mathbf{T}a(a,b), \mathbf{F}a(b,a), \mathbf{T}a(a,a)\}$$
$$\text{Learn minimal}: \{\mathbf{F}e_{\&geq[a,2]}(), \mathbf{T}a(a,b), \mathbf{T}a(a,a)\}$$

5. Empirical Evaluation

Use *DlvhexCVhex* implementation with *Gringo/Clasp* as backends



average runtime over 50 instances (sec.) vs. instance size, (timeout 300 sec.)

- Significant improvements if not very many answer sets
- Tradeoff: time for evaluating external atom $\leftrightarrow$ information gain
- Benefit of nogood minimization depends on size of nogoods

FWF Der Wissenschaftsfonds.