Answer Set Solver Backdoors

Emilia Oikarinen* Matti Järvisalo†

* HIIT & Aalto University, Finland
† HIIT & University of Helsinki, Finland

December 16, 2014 @ Computational Logic Day

In Proc. JELIA 2014, LNAI 8761, pp. 674-683

Aalto University HELSINKI INSTITUTE FOR INFORMATION TECHNOLOGY UNIVERSITY OF HELSINKI ACADEMY OF FINLAND

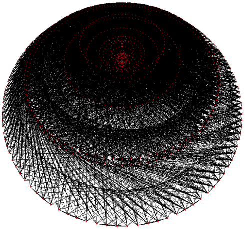
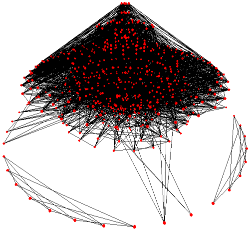
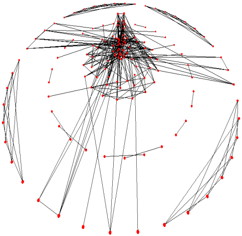
Motivation: ASP

- Answer set programming (ASP)
  - Rule-based constraint programming paradigm
  - Offers an expressive declarative language for conveniently modelling hard combinatorial problems
  - ... together with highly efficient solver technology for finding solutions (answer sets) to the rule-based constraint models.
- Efficient answer set solvers enable addressing a wide range of important real-world problems

Our understanding for the fundamental reasons for this success is limited

Motivation: Backdoors

A generic notion for providing insights to the surprising success of constraint solving remarkably large and complex real-world instances of combinatorial problems

Constraint graph (planning) . . . 5 vars assigned . . . 14 vars assigned

Backdoors to Problem Instances

Set B of variables such that a systematic search procedure needs to non-deterministically branch only on the variables in B in order to decide the instance.

Research Question & Contributions

Research Question

Do ASP solver techniques influence the existence of small backdoors?

- A search procedure having small backdoor to a problem instance can in principle decide the instance efficiently.

Contributions

1. Formalization of backdoors wrt three dimensions of ASP solver techniques
   - (i) Well-foundedness checking; (ii) no-good learning; (iii) branching.
2. Detailed analysis:
   relative size of backdoors for the $2^3 = 8$ solver abstractions
   - Extending earlier results for Boolean satisfiability solvers:
     Dimensions (i) and (iii) non-existing in SAT!

Solver Abstractions

Answer set existence for $\Pi \equiv$ SAT checking $\text{comp}(\Pi) \wedge L(\Pi)$

- $\text{comp}(\Pi)$: Clark's completion
  - Interpret `:-` as a logical equivalence $\leftrightarrow$.
- $L(\Pi)$: *loop formulas* of Π (worst-case exponential)
  - to rule out classical models not corresponding to answer sets.

Answer set solvers in three dimensions (X, Y, Z)

X: **Well-foundedness checking** over the loop formulas $L(\Pi)$

- **EWF**: eagerly after each decision
- **LWF**: lazily only after a satisfying assignment for $\text{comp}(\Pi)$

Y: **No-good learning**:

- **CL**: yes (a la the CDCL SAT algorithm)
- **noCL**: no (a la the DPLL SAT algorithm)

Z: **Branching**

- **B**: On atoms + bodies as atomic constructs
- **noB**: On atoms only

(X, Y, Z)-Abstractions and Solvers

The abstractions are closely related to implemented answer set solvers:

- DLV and Smodels relate most closely with $(\mathbf{EWF}, \mathbf{noCL}, \mathbf{noB})$
- Nomore++ with $(\mathbf{EWF}, \mathbf{noCL}, \mathbf{B})$
- $\texttt{Smodels}_{cc}$ with $(\mathbf{EWF}, \mathbf{CL}, \mathbf{noB})$;
- ASSAT, Cmodels, and SUP with $(\mathbf{LWF}, \mathbf{CL}, \mathbf{B})$
- Clasp, WASP, and SAG with $(\mathbf{EWF}, \mathbf{CL}, \mathbf{B})$

Answer Set Solver Backdoors

$(X, \mathbf{noCL}, Z)$-backdoors

Given a program Π, a subset $B \subseteq \text{atom}(\Pi) \cup \text{body}(\Pi)$ is a $(X, \mathbf{noCL}, Z)$-*backdoor* if

- for every truth assignment $\tau : B \rightarrow \{0, 1\}$,
  - $X = \mathbf{EWF}$: unit propagation on $\text{comp}(\Pi) \wedge L(\Pi)$
  - $X = \mathbf{LWF}$: unit propagation on $\text{comp}(\Pi)$

  returns a satisfying assignment for $\Pi|_\tau$ or concludes that $\Pi|_\tau$ is unsatisfiable.
- $Z = \mathbf{noB}$: $B \subseteq \text{atom}(\Pi)$.

Answer Set Solver Backdoors

$(X, \mathbf{CL}, Z)$-backdoors

A subset $B \subseteq \mathrm{atom}(\Pi) \cup \mathrm{body}(\Pi)$ is a $(X, \mathbf{CL}, Z)$-*backdoor* for Π if there exists a search tree exploration order for the $(X, \mathbf{CL}, Z)$-solver such that:

- The solver branches only on the variables in B.
- The solver uses unit propagation on $\mathrm{comp}(\Pi) \wedge L(\Pi)$ when all variables in B are assigned.
- The solver either finds a satisfying assignment for Π or proves Π unsatisfiable.
- $X = \mathbf{LWF}$: the solver uses $L(\Pi)$ for unit propagation only when the current assignment is complete over $\mathrm{atom}(\Pi) \cup \mathrm{body}(\Pi)$.
- $Z = \mathbf{noB}$: $B \subseteq \mathrm{atom}(\Pi)$.

Analysis: Results

- We compare the size of *smallest backdoors* w.r.t. different solver abstractions
- Results from SAT carry on to ASP using an encoding from CNF into ASP
- For dimensions non-existent in SAT solvers, in order to obtain separation we find program families which have different sizes of smallest backdoors
- UNSAT vs. SAT programs

UNSAT			LWF				EWF			
			noCL		CL		noCL		CL	
			noB	B	noB	B	noB	B	noB	B
LWF	noCL	noB	$\equiv$	$\geq$	$\geq$	$\geq$	$\geq$	$\geq$	$\geq$	$\geq$
		B	$\ll$	$\equiv$	?	$\geq$	$\ll$	$\geq$	?	$\geq$
	CL	noB	$\ll$	$\ll$	$\equiv$	$\geq$	$\ll$	$\ll$	$\geq$	$\geq$
		B	$\ll$	$\ll$	$\leq$	$\equiv$	$\ll$	$\ll$	?	$\geq$
EWF	noCL	noB	$\ll$	$\ll$	$\ll$	?	$\equiv$	$\geq$	$\geq$	$\geq$
		B	$\ll$	$\ll$	$\ll$	?	$\ll$	$\equiv$	?	$\geq$
	CL	noB	$\ll$	$\ll$	$\ll$	?	$\ll$	$\ll$	$\equiv$	$\geq$
		B	$\ll$	$\ll$	$\ll$	$\leq$	$\ll$	$\ll$	$\leq$	$\equiv$

A $\ll$ B *A* can have exponentially smaller backdoors than *B*

A $<$ B *A* can have smaller backdoors than *B*

A $\equiv$ B Sizes of backdoors the same for all programs

A $\leq$ B B $<$ A does not hold, and A $<$ B holds

A $\geq$ B Equivalent to B $\leq$ A

A ? B Relationship unknown

SAT			LWF				EWF			
			noCL		CL		noCL		CL	
			noB	B	noB	B	noB	B	noB	B
LWF	noCL	noB	$\equiv$	$\geq$	$\geq$	$\geq$	$\geq$	$\geq$	$\geq$	$\geq$
		B	$\ll$	$\equiv$	?	$\geq$	$\ll$	$\geq$	?	$\geq$
	CL	noB	$<$	$<$	$\equiv$	$\geq$	$<$	$<$	$\geq$	$\geq$
		B	$<$	$<$	$\leq$	$\equiv$	$<$	$<$	?	$\geq$
EWF	noCL	noB	$\ll$	$\ll$	$\ll$	$\ll$	$\equiv$	$\geq$	$\geq$	$\geq$
		B	$\ll$	$\ll$	$\ll$	$\ll$	$\ll$	$\equiv$	?	$\geq$
	CL	noB	$\ll$	$\ll$	$\ll$	$\ll$	$<$	$<$	$\equiv$	$\geq$
		B	$\ll$	$\ll$	$\ll$	$\ll$	$<$	$<$	$\leq$	$\equiv$

A $\ll$ B *A* can have exponentially smaller backdoors than *B*

A $<$ B *A* can have smaller backdoors than *B*

A $\equiv$ B Sizes of backdoors the same for all programs

A $\leq$ B B $<$ A does not hold, and A $<$ B holds

A $\geq$ B Equivalent to B $\leq$ A

A ? B Relationship unknown