Towards MaxSAT-Based Proof Systems

A Practical Perspective

Joao Marques-Silva

Joint work M.L. Bonet, S. Buss, A. Ignatiev and A. Morgado

University of Lisbon

Workshop on Theory and Practice of Satisfiability Solving

CMO, Oaxaca, México

August 2018

The SAT disruption

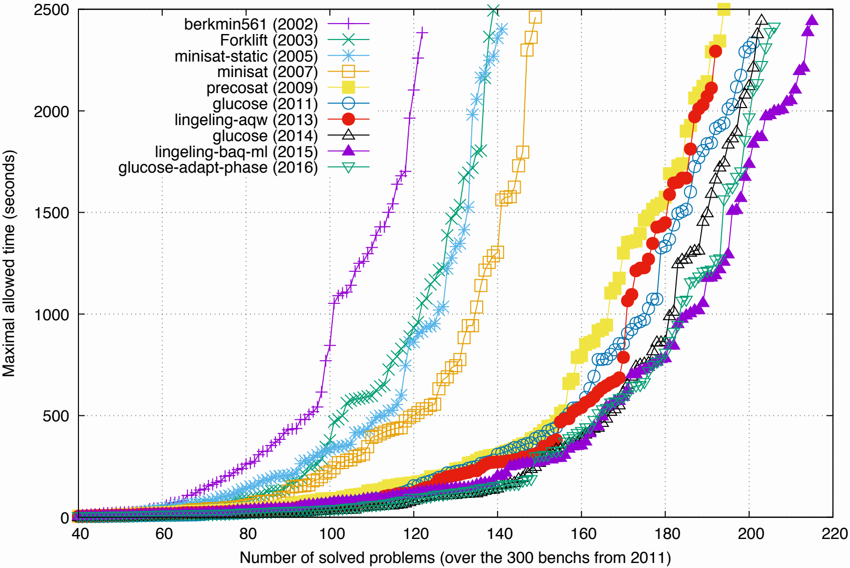
- Key breakthroughs in mid 90s and early 00s

The SAT disruption

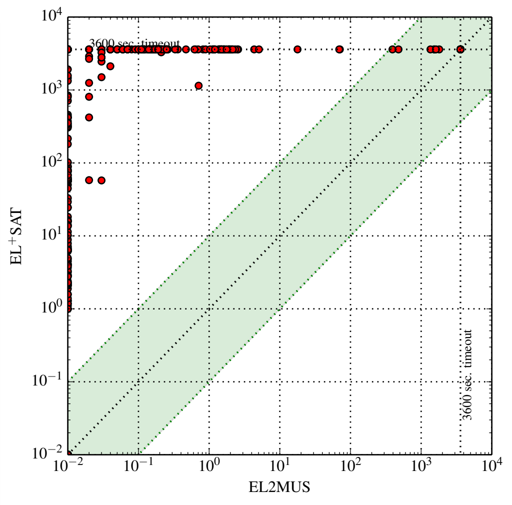
- Key breakthroughs in mid 90s and early 00s
  - **CDCL** SAT solving enabled many successes over the years
  - Hundreds (thousands?) of practical applications

SAT solver evolution

[Source: Simon 2015]



SAT can make the difference – axiom pinpointing



- Instances: $\mathcal{EL}^+$ medical ontologies

How significant is SAT solving?

How significant is SAT solving?

Comm. ACM 2010

DOI:10.1145/1839676.1839677

Moshe Y. Vardi

On P, NP, and Computational Complexity

Today's SAT solvers, which enjoy wide industrial usage, routinely solve SAT instances with over one *million* variables. How can a scary **NP**-complete problem be so easy? What is going on?

The answer is that one must read complexity-theoretic claims carefully. Classical **NP**-completeness theory is about *worst-case* complexity.

My point here is not to criticize complexity theory. It is a beautiful theory that has yielded deep insights over the last 50 years, as well as posed fundamental, tantalizing problems, such as the **P** vs. **NP** problem. But an important role of theory is to shed light on practice, and there we have large gaps. We need, I believe, a richer and broader complexity theory, a theory that would explain both the difficulty and the easiness of problems like SAT. More theory, please!

Moshe Y. Vardi, EDITOR-IN-CHIEF

How significant is SAT solving? And SAT oracles?

Comm. ACM 2010

DOI:10.1145/1839676.1839677

Moshe Y. Vardi

On P, NP, and Computational Complexity

Today's SAT solvers, which enjoy wide industrial usage, routinely solve SAT instances with over one *million* variables. How can a scary **NP**-complete problem be so easy? What is going on?

The answer is that one must read complexity-theoretic claims carefully. Classical **NP**-completeness theory is about *worst-case* complexity.

My point here is not to criticize complexity theory. It is a beautiful theory that has yielded deep insights over the last 50 years, as well as posed fundamental, tantalizing problems, such as the **P** vs. **NP** problem. But an important role of theory is to shed light on practice, and there we have large gaps. We need, I believe, a richer and broader complexity theory, a theory that would explain both the difficulty and the easiness of problems like SAT. More theory, please!

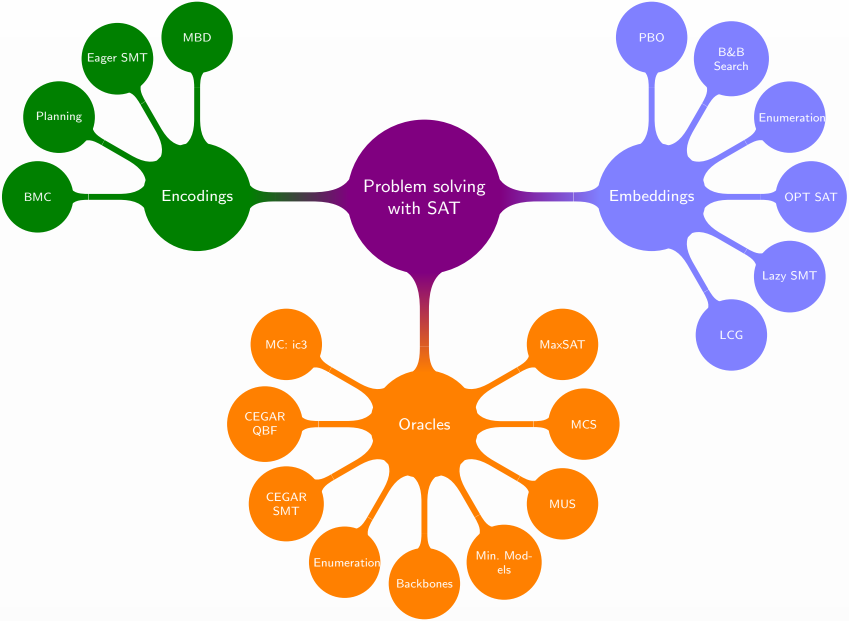
Moshe Y. Vardi, EDITOR-IN-CHIEF

When you have a big hammer, look for nails!

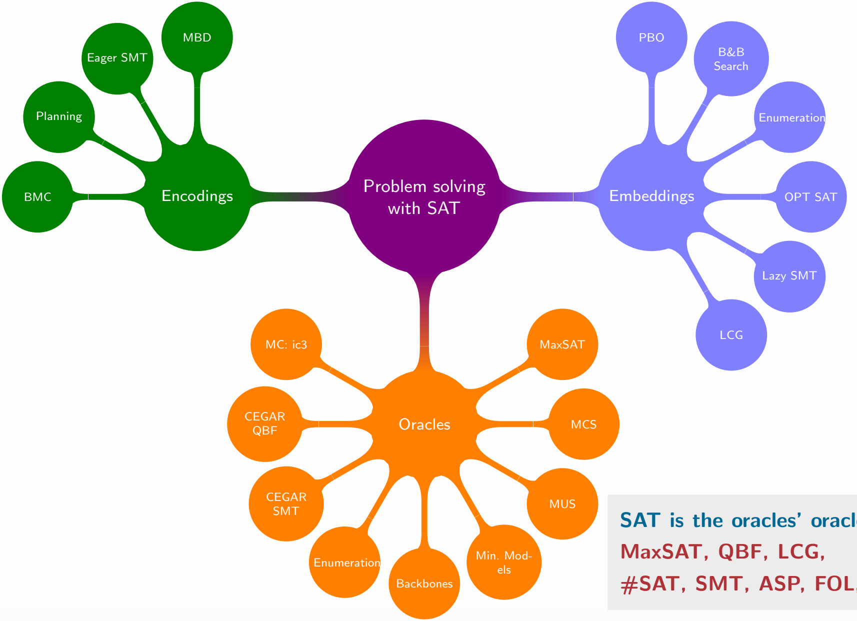
© M. Vardi

© Comm. ACM 2010

SAT is ubiquitous in problem solving



SAT is ubiquitous in problem solving



SAT is the oracles' oracle:
MaxSAT, QBF, LCG, #SAT, SMT, ASP, FOL, ...

What is Maximum Satisfiability (MaxSAT)?

$x_6 \vee x_2$	$\neg x_6 \vee x_2$	$\neg x_2 \vee x_1$	$\neg x_1$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4$	$\neg x_4 \vee x_5$
$x_7 \vee x_5$	$\neg x_7 \vee x_5$	$\neg x_5 \vee x_3$	$\neg x_3$

What is Maximum Satisfiability (MaxSAT)?

$\boxed{x_6 \vee x_2}$	$\neg x_6 \vee x_2$	$\neg x_2 \vee x_1$	$\neg x_1$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4$	$\boxed{\neg x_4 \vee x_5}$
$\boxed{x_7 \vee x_5}$	$\neg x_7 \vee x_5$	$\neg x_5 \vee x_3$	$\neg x_3$

- Given unsatisfiable formula find largest satisfiable subset of clauses

What is Maximum Satisfiability (MaxSAT)?

$x_6 \vee x_2$	$\neg x_6 \vee x_2$	$\neg x_2 \vee x_1$	$\boxed{\neg x_1}$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4$	$\neg x_4 \vee x_5$
$\boxed{x_7 \vee x_5}$	$\neg x_7 \vee x_5$	$\neg x_5 \vee x_3$	$\neg x_3$

- Given unsatisfiable formula, find largest satisfiable subset of clauses

What is Maximum Satisfiability (MaxSAT)?

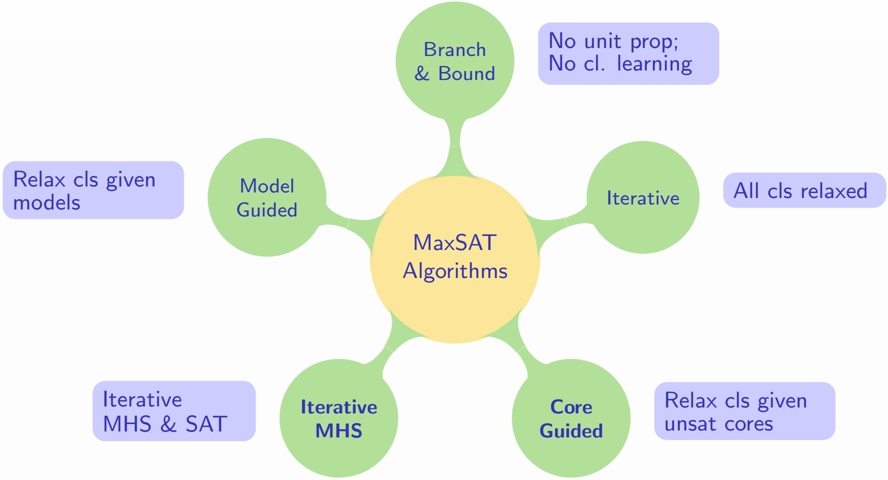
$x_6 \vee x_2$	$\neg x_6 \vee x_2$	$\neg x_2 \vee x_1$	$\neg x_1$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4$	$\neg x_4 \vee x_5$
$x_7 \vee x_5$	$\neg x_7 \vee x_5$	$\neg x_5 \vee x_3$	$\neg x_3$

- Given unsatisfiable formula, find largest satisfiable subset of clauses

MaxSAT Variants		Hard Clauses? No	Hard Clauses? Yes
Weights?	No	Plain	Partial
	Yes	Weighted	Weighted Partial

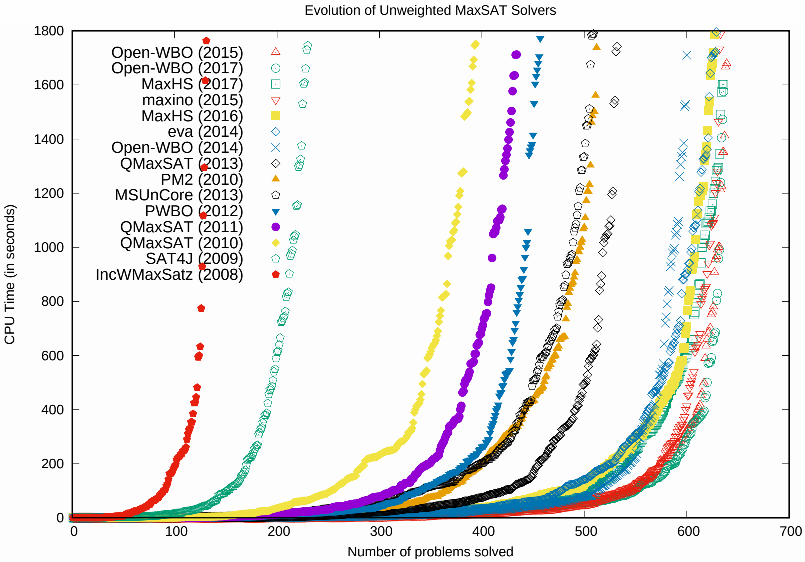
- **Many** practical applications [e.g. SZGN17]

Many MaxSAT approaches



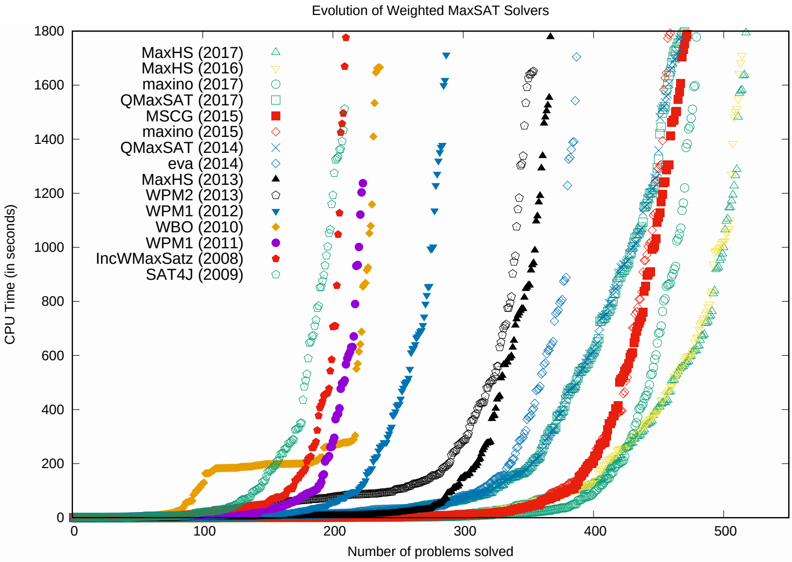
- For practical (**industrial**) instances: **core-guided & MHS** approaches are the most effective [MaxSAT17]

MaxSAT (r)evolution – unweighted instances 2008-2017



Source: [MaxSAT 2017 organizers]

MaxSAT (r)evolution – weighted instances 2008-2017



Source: [MaxSAT 2017 organizers]

What about in 2018?

What about in 2018? – complete tracks

Source: [MaxSAT 2017 organizers]

Unweighted			Weighted		
Solver	#Solved	Time (Avg)	Solver	#Solved	Time (Avg)
RC2-B	421	126.32	**RC2-B**	421	256.02
RC2-A	416	138.98	RC2-A	416	267.55
maxino	405	137.50	MaxHS	390	274.87
MaxHS	386	178.06	Pacose	390	348.98
Open-WBO-Gluc	382	171.54	QMaxSAT	381	320.78

What about in 2018? – complete tracks

Source: [MaxSAT 2017 organizers]

Unweighted			Weighted		
Solver	#Solved	Time (Avg)	Solver	#Solved	Time (Avg)
RC2-B	421	126.32	**RC2-B**	421	256.02
RC2-A	416	138.98	RC2-A	416	267.55
maxino	405	137.50	MaxHS	390	274.87
MaxHS	386	178.06	Pacose	390	348.98
Open-WBO-Gluc	382	171.54	QMaxSAT	381	320.78

- **Note:** RC2 is a variant of a **2014** algorithm, with some practical optimizations
  - **Core-guided**, based on lower-bound refinement [FM06,MSP07]
  - Exploits **soft** cardinality constraints [MDMS14]
  - Inspired by OLL algorithm, first used in ASP optimization [AKMS12]

What about in 2018? – complete tracks

Source: [MaxSAT 2017 organizers]

Unweighted			Weighted		
Solver	#Solved	Time (Avg)	Solver	#Solved	Time (Avg)
RC2-B	421	126.32	**RC2-B**	421	256.02
RC2-A	416	138.98	RC2-A	416	267.55
maxino	405	137.50	MaxHS	390	274.87
MaxHS	386	178.06	Pacose	390	348.98
Open-WBO-Gluc	382	171.54	QMaxSAT	381	320.78

- **Note:** RC2 is a variant of a **2014** algorithm, with some practical optimizations, **and implemented with PySAT**
  - **Core-guided**, based on lower-bound refinement [FM06,MSP07]
  - Exploits **soft** cardinality constraints [MDMS14]
  - Inspired by OLL algorithm, first used in ASP optimization [AKMS12]

Outline

MaxSAT Solving

Horn MaxSAT

PHP Refutations in Polynomial Time

Outline

MSU3 core-guided algorithm

(M.-S.&Planes,CoRR'07)

$x_6 \vee x_2$	$\neg x_6 \vee x_2$	$\neg x_2 \vee x_1$	$\neg x_1$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4$	$\neg x_4 \vee x_5$
$x_7 \vee x_5$	$\neg x_7 \vee x_5$	$\neg x_5 \vee x_3$	$\neg x_3$

Example CNF formula

MSU3 core-guided algorithm

(M.-S.&Planes,CoRR'07)

$x_6 \vee x_2$	$\neg x_6 \vee x_2$	$\boxed{\neg x_2 \vee x_1}$	$\boxed{\neg x_1}$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$\boxed{x_2 \vee x_4}$	$\boxed{\neg x_4 \vee x_5}$
$x_7 \vee x_5$	$\neg x_7 \vee x_5$	$\boxed{\neg x_5 \vee x_3}$	$\boxed{\neg x_3}$

Formula is UNSAT; $\mathrm{OPT} \leq |\varphi| - 1$; Get unsat core

MSU3 core-guided algorithm

(M.-S.&Planes,CoRR'07)

$x_6 \vee x_2$	$\neg x_6 \vee x_2$	$\neg x_2 \vee x_1 \vee r_1$	$\neg x_1 \vee r_2$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4 \vee r_3$	$\neg x_4 \vee x_5 \vee r_4$
$x_7 \vee x_5$	$\neg x_7 \vee x_5$	$\neg x_5 \vee x_3 \vee r_5$	$\neg x_3 \vee r_6$

$\sum_{i=1}^{6} r_i \leq 1$

Add relaxation variables and AtMost$k$, $k = 1$, constraint

MSU3 core-guided algorithm

(M.-S.&Planes,CoRR'07)

$x_6 \vee x_2$	$\neg x_6 \vee x_2$	$\neg x_2 \vee x_1 \vee r_1$	$\neg x_1 \vee r_2$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4 \vee r_3$	$\neg x_4 \vee x_5 \vee r_4$
$x_7 \vee x_5$	$\neg x_7 \vee x_5$	$\neg x_5 \vee x_3 \vee r_5$	$\neg x_3 \vee r_6$

$\sum_{i=1}^{6} r_i \leq 1$

Formula is (again) UNSAT; OPT $\leq |\varphi| - 2$; Get unsat core

MSU3 core-guided algorithm

(M.-S.&Planes,CoRR'07)

$x_6 \vee x_2 \vee r_7$ $\quad \neg x_6 \vee x_2 \vee r_8$ $\quad \neg x_2 \vee x_1 \vee r_1$ $\quad \neg x_1 \vee r_2$

$\neg x_6 \vee x_8$ $\quad x_6 \vee \neg x_8$ $\quad x_2 \vee x_4 \vee r_3$ $\quad \neg x_4 \vee x_5 \vee r_4$

$x_7 \vee x_5 \vee r_9$ $\quad \neg x_7 \vee x_5 \vee r_{10}$ $\quad \neg x_5 \vee x_3 \vee r_5$ $\quad \neg x_3 \vee r_6$

$\sum_{i=1}^{10} r_i \leq 2$

Add new relaxation variables and update AtMost$k$, k=2, constraint

MSU3 core-guided algorithm

(M.-S.&Planes,CoRR'07)

$x_6 \vee x_2 \vee r_7$	$\neg x_6 \vee x_2 \vee r_8$	$\neg x_2 \vee x_1 \vee r_1$	$\neg x_1 \vee r_2$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4 \vee r_3$	$\neg x_4 \vee x_5 \vee r_4$
$x_7 \vee x_5 \vee r_9$	$\neg x_7 \vee x_5 \vee r_{10}$	$\neg x_5 \vee x_3 \vee r_5$	$\neg x_3 \vee r_6$

$\sum_{i=1}^{10} r_i \leq 2$

Instance is now SAT

MSU3 core-guided algorithm

(M.-S.&Planes,CoRR'07)

$x_6 \vee x_2 \vee r_7$ $\quad \neg x_6 \vee x_2 \vee r_8$ $\quad \neg x_2 \vee x_1 \vee r_1$ $\quad \neg x_1 \vee r_2$

$\neg x_6 \vee x_8$ $\quad x_6 \vee \neg x_8$ $\quad x_2 \vee x_4 \vee r_3$ $\quad \neg x_4 \vee x_5 \vee r_4$

$x_7 \vee x_5 \vee r_9$ $\quad \neg x_7 \vee x_5 \vee r_{10}$ $\quad \neg x_5 \vee x_3 \vee r_5$ $\quad \neg x_3 \vee r_6$

$\sum_{i=1}^{10} r_i \leq 2$

MaxSAT solution is $|\varphi| - \mathcal{I} = 12 - 2 = 10$

MSU3 core-guided algorithm

(M.-S.&Planes,CoRR'07)

Builds on FM06 seminal work ...

$x_6 \vee x_2 \vee r_7$	$\neg x_6 \vee x_2 \vee r_8$	$\neg x_2 \vee x_1 \vee r_1$	$\neg x_1 \vee r_2$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4 \vee r_3$	$\neg x_4 \vee x_5 \vee r_4$
$x_7 \vee x_5 \vee r_9$	$\neg x_7 \vee x_5 \vee r_{10}$	$\neg x_5 \vee x_3 \vee r_5$	$\neg x_3 \vee r_6$

$\sum_{i=1}^{10} r_i \leq 2$

MaxSAT solution is $|\varphi| - \mathcal{I} = 12 - 2 = 10$

AtMost$k$/PB constraints used

Relaxed soft clauses become **hard**

MSU3 core-guided algorithm

(M.-S.&Planes,CoRR'07)

Builds on FM06 seminal work ...

$x_6 \vee x_2 \vee r_7$ $\quad \neg x_6 \vee x_2 \vee r_8$ $\quad \neg x_2 \vee x_1 \vee r_1$ $\quad \neg x_1 \vee r_2$

$\neg x_6 \vee x_8$ $\quad x_6 \vee \neg x_8$ $\quad x_2 \vee x_4 \vee r_3$ $\quad \neg x_4 \vee x_5 \vee r_4$

$x_7 \vee x_5 \vee r_9$ $\quad \neg x_7 \vee x_5 \vee r_{10}$ $\quad \neg x_5 \vee x_3 \vee r_5$ $\quad \neg x_3 \vee r_6$

$\sum_{i=1}^{10} r_i \leq 2$

MaxSAT solution is $|\varphi| - \mathcal{I} = 12 - 2 = 10$

AtMost$k$/PB constraints used

Some clauses not relaxed

Relaxed soft clauses become **hard**

MSU3 core-guided algorithm

(M.-S.&Planes,CoRR'07)

Builds on FM06 seminal work ...

$x_6 \vee x_2 \vee r_7$ $\quad \neg x_6 \vee x_2 \vee r_8$ $\quad \neg x_2 \vee x_1 \vee r_1$ $\quad \neg x_1 \vee r_2$

$\neg x_6 \vee x_8$ $\quad x_6 \vee \neg x_8$ $\quad x_2 \vee x_4 \vee r_3$ $\quad \neg x_4 \vee x_5 \vee r_4$

$x_7 \vee$ ... $\vee r_6$

$\sum_{i=1}^{10} r_i \leq 2$

Note: # of SAT oracle calls grows linear with solution cost!

MaxSAT solution is $|\varphi| - \mathcal{I} = 12 - 2 = 10$

AtMost$k$/PB constraints used

Some clauses not relaxed

Relaxed soft clauses become **hard**

Outline

MaxSAT Solving

- Core Guided with MSU3 – Example
- Core Guided with RC2 – Example
- MaxHS – Example
- MaxHS – Algorithm

Horn MaxSAT

PHP Refutations in Polynomial Time

Soft cardinality constraints

(Morgado,Dodaro&M.-S.,CP'14)

$x_6 \vee x_2$	$\neg x_6 \vee x_2$	$\neg x_2 \vee x_1$	$\neg x_1$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4$	$\neg x_4 \vee x_5$
$x_7 \vee x_5$	$\neg x_7 \vee x_5$	$\neg x_5 \vee x_3$	$\neg x_3$

Example CNF formula

Soft cardinality constraints

(Morgado,Dodaro&M.-S.,CP'14)

$x_6 \vee x_2$	$\neg x_6 \vee x_2$	$\neg x_2 \vee x_1$	$\neg x_1$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4$	$\neg x_4 \vee x_5$
$x_7 \vee x_5$	$\neg x_7 \vee x_5$	$\neg x_5 \vee x_3$	$\neg x_3$

Formula is UNSAT; OPT $\leq |\varphi| - 1$; Get unsat core

Soft cardinality constraints

(Morgado,Dodaro&M.-S.,CP'14)

$x_6 \vee x_2$	$\neg x_6 \vee x_2$	$\neg x_2 \vee x_1 \vee r_1$	$\neg x_1 \vee r_2$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4 \vee r_3$	$\neg x_4 \vee x_5 \vee r_4$
$x_7 \vee x_5$	$\neg x_7 \vee x_5$	$\neg x_5 \vee x_3 \vee r_5$	$\neg x_3 \vee r_6$
$S_1 \leq 1$			

Aux sums: $S_1 = \sum_{i=1}^{6} r_i$;

Add relaxation variables and AtMost1 constraint

Soft cardinality constraints

(Morgado,Dodaro&M.-S.,CP'14)

$x_6 \vee x_2$ $\quad \neg x_6 \vee x_2$ $\quad \neg x_2 \vee x_1 \vee r_1$ $\quad \neg x_1 \vee r_2$

$\neg x_6 \vee x_8$ $\quad x_6 \vee \neg x_8$ $\quad x_2 \vee x_4 \vee r_3$ $\quad \neg x_4 \vee x_5 \vee r_4$

$x_7 \vee x_5$ $\quad \neg x_7 \vee x_5$ $\quad \neg x_5 \vee x_3 \vee r_5$ $\quad \neg x_3 \vee r_6$

$S_1 \leq 1$

Aux sums: $S_1 = \sum_{i=1}^{6} r_i$;

Formula is (again) UNSAT; OPT $\leq |\varphi| - 2$; Get unsat core

Soft cardinality constraints

(Morgado,Dodaro&M.-S.,CP'14)

$x_6 \vee x_2 \vee r_7$ $\quad$ $\neg x_6 \vee x_2 \vee r_8$ $\quad$ $\neg x_2 \vee x_1 \vee r_1$ $\quad$ $\neg x_1 \vee r_2$

$\neg x_6 \vee x_8$ $\quad$ $x_6 \vee \neg x_8$ $\quad$ $x_2 \vee x_4 \vee r_3$ $\quad$ $\neg x_4 \vee x_5 \vee r_4$

$x_7 \vee x_5 \vee r_9$ $\quad$ $\neg x_7 \vee x_5 \vee r_{10}$ $\quad$ $\neg x_5 \vee x_3 \vee r_5$ $\quad$ $\neg x_3 \vee r_6$

$S_1 \leq 2 \qquad S_2' + \neg(S_1 \leq 1) \leq 1$

Aux sums: $S_1 = \sum_{i=1}^{6} r_i$; $S_2' = \sum_{i=7}^{10} r_i$; $S_2 = S_2' + \neg(S_1 \leq 1)$

Add new relaxation variables (S_2'), update AtMost$k$ constraint and add new AtMost1 constraint

Soft cardinality constraints

(Morgado,Dodaro&M.-S.,CP'14)

$x_6 \vee x_2 \vee r_7$	$\neg x_6 \vee x_2 \vee r_8$	$\neg x_2 \vee x_1 \vee r_1$	$\neg x_1 \vee r_2$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4 \vee r_3$	$\neg x_4 \vee x_5 \vee r_4$
$x_7 \vee x_5 \vee r_9$	$\neg x_7 \vee x_5 \vee r_{10}$	$\neg x_5 \vee x_3 \vee r_5$	$\neg x_3 \vee r_6$

$\boxed{S_1 \leq 2}$ $\boxed{S_2' + \neg(S_1 \leq 1) \leq 1}$

$S_1 \geq 2 \rightarrow S_2' = 0$
$S_1 \leq 1 \rightarrow S_2' \leq 1$

Aux sums: $S_1 = \sum_{i=1}^{6} r_i$; $S_2' = \sum_{i=7}^{10} r_i$; $S_2 = S_2' + \neg(S_1 \leq 1)$

Add new relaxation variables (S_2'), update AtMost$k$ constraint and add new AtMost1 constraint

Soft cardinality constraints

(Morgado,Dodaro&M.-S.,CP'14)

$x_6 \vee x_2 \vee r_7$	$\neg x_6 \vee x_2 \vee r_8$	$\neg x_2 \vee x_1 \vee r_1$	$\neg x_1 \vee r_2$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4 \vee r_3$	$\neg x_4 \vee x_5 \vee r_4$
$x_7 \vee x_5 \vee r_9$	$\neg x_7 \vee x_5 \vee r_{10}$	$\neg x_5 \vee x_3 \vee r_5$	$\neg x_3 \vee r_6$
$S_1 \leq 2$	$S_2' + \neg(S_1 \leq 1) \leq 1$		

Aux sums: $S_1 = \sum_{i=1}^{6} r_i$; $S_2' = \sum_{i=7}^{10} r_i$; $S_2 = S_2' + \neg(S_1 \leq 1)$

Instance is now SAT

Soft cardinality constraints

(Morgado,Dodaro&M.-S.,CP'14)

$x_6 \vee x_2 \vee r_7$	$\neg x_6 \vee x_2 \vee r_8$	$\neg x_2 \vee x_1 \vee r_1$	$\neg x_1 \vee r_2$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4 \vee r_3$	$\neg x_4 \vee x_5 \vee r_4$
$x_7 \vee x_5 \vee r_9$	$\neg x_7 \vee x_5 \vee r_{10}$	$\neg x_5 \vee x_3 \vee r_5$	$\neg x_3 \vee r_6$

$S_1 \leq 2 \qquad S_2' + \neg(S_1 \leq 1) \leq 1$

Aux sums: $S_1 = \sum_{i=1}^{6} r_i$; $S_2' = \sum_{i=7}^{10} r_i$; $S_2 = S_2' + \neg(S_1 \leq 1)$

MaxSAT solution is $|\varphi| - \mathcal{I} = 12 - 2 = 10$

Soft cardinality constraints

(Morgado,Dodaro&M.-S.,CP'14)

Builds on other algorithms: FM06, MSP07, ...

$x_6 \vee x_2 \vee r_7$	$\neg x_6 \vee x_2 \vee r_8$	$\neg x_2 \vee x_1 \vee r_1$	$\neg x_1 \vee r_2$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4 \vee r_3$	$\neg x_4 \vee x_5 \vee r_4$
$x_7 \vee x_5 \vee r_9$	$\neg x_7 \vee x_5 \vee r_{10}$	$\neg x_5 \vee x_3 \vee r_5$	$\neg x_3 \vee r_6$

$S_1 \leq 2 \qquad S'_2 + \neg(S_1 \leq 1) \leq 1$

Aux sums: $S_1 = \sum_{i=1}^{6} r_i$; $S'_2 = \sum_{i=7}^{10} r_i$; $S_2 = S'_2 + \neg(S_1 \leq 1)$

MaxSAT solution is $|\varphi| - \mathcal{I} = 12 - 2 = 10$

Only AtMost$k$ constraints used

Sums reused with $\neq$ RHSs

Relaxed soft clauses become **hard**

Outline

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$ $\quad c_2 = \neg x_6 \vee x_2$ $\quad c_3 = \neg x_2 \vee x_1$ $\quad c_4 = \neg x_1$

$c_5 = \neg x_6 \vee x_8$ $\quad c_6 = x_6 \vee \neg x_8$ $\quad c_7 = x_2 \vee x_4$ $\quad c_8 = \neg x_4 \vee x_5$

$c_9 = x_7 \vee x_5$ $\quad c_{10} = \neg x_7 \vee x_5$ $\quad c_{11} = \neg x_5 \vee x_3$ $\quad c_{12} = \neg x_3$

$$\mathcal{K} = \emptyset$$

- Find MHS of $\mathcal{K}$:

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$$c_1 = x_6 \vee x_2 \qquad c_2 = \neg x_6 \vee x_2 \qquad c_3 = \neg x_2 \vee x_1 \qquad c_4 = \neg x_1$$

$$c_5 = \neg x_6 \vee x_8 \qquad c_6 = x_6 \vee \neg x_8 \qquad c_7 = x_2 \vee x_4 \qquad c_8 = \neg x_4 \vee x_5$$

$$c_9 = x_7 \vee x_5 \qquad c_{10} = \neg x_7 \vee x_5 \qquad c_{11} = \neg x_5 \vee x_3 \qquad c_{12} = \neg x_3$$

$$\mathcal{K} = \emptyset$$

- Find MHS of $\mathcal{K}$: $\emptyset$

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$ $\quad c_2 = \neg x_6 \vee x_2$ $\quad c_3 = \neg x_2 \vee x_1$ $\quad c_4 = \neg x_1$

$c_5 = \neg x_6 \vee x_8$ $\quad c_6 = x_6 \vee \neg x_8$ $\quad c_7 = x_2 \vee x_4$ $\quad c_8 = \neg x_4 \vee x_5$

$c_9 = x_7 \vee x_5$ $\quad c_{10} = \neg x_7 \vee x_5$ $\quad c_{11} = \neg x_5 \vee x_3$ $\quad c_{12} = \neg x_3$

$$\mathcal{K} = \emptyset$$

- Find MHS of $\mathcal{K}$: $\emptyset$
- SAT($\mathcal{F} \setminus \emptyset$)?

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$ $\quad c_2 = \neg x_6 \vee x_2$ $\quad c_3 = \neg x_2 \vee x_1$ $\quad c_4 = \neg x_1$

$c_5 = \neg x_6 \vee x_8$ $\quad c_6 = x_6 \vee \neg x_8$ $\quad c_7 = x_2 \vee x_4$ $\quad c_8 = \neg x_4 \vee x_5$

$c_9 = x_7 \vee x_5$ $\quad c_{10} = \neg x_7 \vee x_5$ $\quad c_{11} = \neg x_5 \vee x_3$ $\quad c_{12} = \neg x_3$

$$\mathcal{K} = \emptyset$$

- Find MHS of $\mathcal{K}$: $\emptyset$
- SAT($\mathcal{F} \setminus \emptyset$)? No

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$ $\quad c_2 = \neg x_6 \vee x_2$ $\quad c_3 = \neg x_2 \vee x_1$ $\quad c_4 = \neg x_1$

$c_5 = \neg x_6 \vee x_8$ $\quad c_6 = x_6 \vee \neg x_8$ $\quad c_7 = x_2 \vee x_4$ $\quad c_8 = \neg x_4 \vee x_5$

$c_9 = x_7 \vee x_5$ $\quad c_{10} = \neg x_7 \vee x_5$ $\quad c_{11} = \neg x_5 \vee x_3$ $\quad c_{12} = \neg x_3$

$$\mathcal{K} = \emptyset$$

- Find MHS of $\mathcal{K}$: $\emptyset$
- SAT($\mathcal{F} \setminus \emptyset$)? No
- Core of $\mathcal{F}$: $\{c_1, c_2, c_3, c_4\}$

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$	$c_2 = \neg x_6 \vee x_2$	$c_3 = \neg x_2 \vee x_1$	$c_4 = \neg x_1$
$c_5 = \neg x_6 \vee x_8$	$c_6 = x_6 \vee \neg x_8$	$c_7 = x_2 \vee x_4$	$c_8 = \neg x_4 \vee x_5$
$c_9 = x_7 \vee x_5$	$c_{10} = \neg x_7 \vee x_5$	$c_{11} = \neg x_5 \vee x_3$	$c_{12} = \neg x_3$

$$\mathcal{K} = \{\{c_1, c_2, c_3, c_4\}\}$$

- Find MHS of $\mathcal{K}$: $\emptyset$
- SAT($\mathcal{F} \setminus \emptyset$)? No
- Core of $\mathcal{F}$: $\{c_1, c_2, c_3, c_4\}$. Update $\mathcal{K}$

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$	$c_2 = \neg x_6 \vee x_2$	$c_3 = \neg x_2 \vee x_1$	$c_4 = \neg x_1$
$c_5 = \neg x_6 \vee x_8$	$c_6 = x_6 \vee \neg x_8$	$c_7 = x_2 \vee x_4$	$c_8 = \neg x_4 \vee x_5$
$c_9 = x_7 \vee x_5$	$c_{10} = \neg x_7 \vee x_5$	$c_{11} = \neg x_5 \vee x_3$	$c_{12} = \neg x_3$

$$\mathcal{K} = \{\{c_1, c_2, c_3, c_4\}\}$$

- Find MHS of $\mathcal{K}$:

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$ $\quad c_2 = \neg x_6 \vee x_2$ $\quad c_3 = \neg x_2 \vee x_1$ $\quad c_4 = \neg x_1$

$c_5 = \neg x_6 \vee x_8$ $\quad c_6 = x_6 \vee \neg x_8$ $\quad c_7 = x_2 \vee x_4$ $\quad c_8 = \neg x_4 \vee x_5$

$c_9 = x_7 \vee x_5$ $\quad c_{10} = \neg x_7 \vee x_5$ $\quad c_{11} = \neg x_5 \vee x_3$ $\quad c_{12} = \neg x_3$

$$\mathcal{K} = \{\{c_1, c_2, c_3, c_4\}\}$$

- Find MHS of $\mathcal{K}$: E.g. $\{c_1\}$

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$$c_1 = x_6 \vee x_2 \qquad c_2 = \neg x_6 \vee x_2 \qquad c_3 = \neg x_2 \vee x_1 \qquad c_4 = \neg x_1$$

$$c_5 = \neg x_6 \vee x_8 \qquad c_6 = x_6 \vee \neg x_8 \qquad c_7 = x_2 \vee x_4 \qquad c_8 = \neg x_4 \vee x_5$$

$$c_9 = x_7 \vee x_5 \qquad c_{10} = \neg x_7 \vee x_5 \qquad c_{11} = \neg x_5 \vee x_3 \qquad c_{12} = \neg x_3$$

$$\mathcal{K} = \{\{c_1, c_2, c_3, c_4\}\}$$

- Find MHS of $\mathcal{K}$: E.g. $\{c_1\}$
- SAT($\mathcal{F} \setminus \{c_1\}$)?

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$	$c_2 = \neg x_6 \vee x_2$	$c_3 = \neg x_2 \vee x_1$	$c_4 = \neg x_1$
$c_5 = \neg x_6 \vee x_8$	$c_6 = x_6 \vee \neg x_8$	$c_7 = x_2 \vee x_4$	$c_8 = \neg x_4 \vee x_5$
$c_9 = x_7 \vee x_5$	$c_{10} = \neg x_7 \vee x_5$	$c_{11} = \neg x_5 \vee x_3$	$c_{12} = \neg x_3$

$$\mathcal{K} = \{\{c_1, c_2, c_3, c_4\}\}$$

- Find MHS of $\mathcal{K}$: E.g. $\{c_1\}$
- SAT($\mathcal{F} \setminus \{c_1\}$)? No

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$ $\quad c_2 = \neg x_6 \vee x_2$ $\quad c_3 = \neg x_2 \vee x_1$ $\quad c_4 = \neg x_1$

$c_5 = \neg x_6 \vee x_8$ $\quad c_6 = x_6 \vee \neg x_8$ $\quad c_7 = x_2 \vee x_4$ $\quad c_8 = \neg x_4 \vee x_5$

$c_9 = x_7 \vee x_5$ $\quad c_{10} = \neg x_7 \vee x_5$ $\quad c_{11} = \neg x_5 \vee x_3$ $\quad c_{12} = \neg x_3$

$$\mathcal{K} = \{\{c_1, c_2, c_3, c_4\}\}$$

- Find MHS of $\mathcal{K}$: E.g. $\{c_1\}$
- SAT($\mathcal{F} \setminus \{c_1\}$)? No
- Core of $\mathcal{F}$: $\{c_9, c_{10}, c_{11}, c_{12}\}$

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$ $\quad c_2 = \neg x_6 \vee x_2$ $\quad c_3 = \neg x_2 \vee x_1$ $\quad c_4 = \neg x_1$

$c_5 = \neg x_6 \vee x_8$ $\quad c_6 = x_6 \vee \neg x_8$ $\quad c_7 = x_2 \vee x_4$ $\quad c_8 = \neg x_4 \vee x_5$

$c_9 = x_7 \vee x_5$ $\quad c_{10} = \neg x_7 \vee x_5$ $\quad c_{11} = \neg x_5 \vee x_3$ $\quad c_{12} = \neg x_3$

$$\mathcal{K} = \{\{c_1, c_2, c_3, c_4\}, \{c_9, c_{10}, c_{11}, c_{12}\}\}$$

- Find MHS of $\mathcal{K}$: E.g. $\{c_1\}$
- SAT($\mathcal{F} \setminus \{c_1\}$)? No
- Core of $\mathcal{F}$: $\{c_9, c_{10}, c_{11}, c_{12}\}$. Update $\mathcal{K}$

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$ $\quad c_2 = \neg x_6 \vee x_2$ $\quad c_3 = \neg x_2 \vee x_1$ $\quad c_4 = \neg x_1$

$c_5 = \neg x_6 \vee x_8$ $\quad c_6 = x_6 \vee \neg x_8$ $\quad c_7 = x_2 \vee x_4$ $\quad c_8 = \neg x_4 \vee x_5$

$c_9 = x_7 \vee x_5$ $\quad c_{10} = \neg x_7 \vee x_5$ $\quad c_{11} = \neg x_5 \vee x_3$ $\quad c_{12} = \neg x_3$

$$\mathcal{K} = \{\{c_1, c_2, c_3, c_4\}, \{c_9, c_{10}, c_{11}, c_{12}\}\}$$

- Find MHS of $\mathcal{K}$:

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$$c_1 = x_6 \vee x_2 \qquad c_2 = \neg x_6 \vee x_2 \qquad c_3 = \neg x_2 \vee x_1 \qquad c_4 = \neg x_1$$

$$c_5 = \neg x_6 \vee x_8 \qquad c_6 = x_6 \vee \neg x_8 \qquad c_7 = x_2 \vee x_4 \qquad c_8 = \neg x_4 \vee x_5$$

$$c_9 = x_7 \vee x_5 \qquad c_{10} = \neg x_7 \vee x_5 \qquad c_{11} = \neg x_5 \vee x_3 \qquad c_{12} = \neg x_3$$

$$\mathcal{K} = \{\{c_1, c_2, c_3, c_4\}, \{c_9, c_{10}, c_{11}, c_{12}\}\}$$

- Find MHS of $\mathcal{K}$: E.g. $\{c_1, c_9\}$

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$ $\quad c_2 = \neg x_6 \vee x_2$ $\quad c_3 = \neg x_2 \vee x_1$ $\quad c_4 = \neg x_1$

$c_5 = \neg x_6 \vee x_8$ $\quad c_6 = x_6 \vee \neg x_8$ $\quad c_7 = x_2 \vee x_4$ $\quad c_8 = \neg x_4 \vee x_5$

$c_9 = x_7 \vee x_5$ $\quad c_{10} = \neg x_7 \vee x_5$ $\quad c_{11} = \neg x_5 \vee x_3$ $\quad c_{12} = \neg x_3$

$$\mathcal{K} = \{\{c_1, c_2, c_3, c_4\}, \{c_9, c_{10}, c_{11}, c_{12}\}\}$$

- Find MHS of $\mathcal{K}$: E.g. $\{c_1, c_9\}$
- SAT$(\mathcal{F} \setminus \{c_1, c_9\})$?

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$$c_1 = x_6 \vee x_2 \qquad c_2 = \neg x_6 \vee x_2 \qquad c_3 = \neg x_2 \vee x_1 \qquad c_4 = \neg x_1$$

$$c_5 = \neg x_6 \vee x_8 \qquad c_6 = x_6 \vee \neg x_8 \qquad c_7 = x_2 \vee x_4 \qquad c_8 = \neg x_4 \vee x_5$$

$$c_9 = x_7 \vee x_5 \qquad c_{10} = \neg x_7 \vee x_5 \qquad c_{11} = \neg x_5 \vee x_3 \qquad c_{12} = \neg x_3$$

$$\mathcal{K} = \{\{c_1, c_2, c_3, c_4\}, \{c_9, c_{10}, c_{11}, c_{12}\}\}$$

- Find MHS of $\mathcal{K}$: E.g. $\{c_1, c_9\}$
- SAT$(\mathcal{F} \setminus \{c_1, c_9\})$? No

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$ $\quad c_2 = \neg x_6 \vee x_2$ $\quad c_3 = \neg x_2 \vee x_1$ $\quad c_4 = \neg x_1$

$c_5 = \neg x_6 \vee x_8$ $\quad c_6 = x_6 \vee \neg x_8$ $\quad c_7 = x_2 \vee x_4$ $\quad c_8 = \neg x_4 \vee x_5$

$c_9 = x_7 \vee x_5$ $\quad c_{10} = \neg x_7 \vee x_5$ $\quad c_{11} = \neg x_5 \vee x_3$ $\quad c_{12} = \neg x_3$

$$\mathcal{K} = \{\{c_1, c_2, c_3, c_4\}, \{c_9, c_{10}, c_{11}, c_{12}\}\}$$

- Find MHS of $\mathcal{K}$: E.g. $\{c_1, c_9\}$
- SAT($\mathcal{F} \setminus \{c_1, c_9\}$)? No
- Core of $\mathcal{F}$: $\{c_3, c_4, c_7, c_8, c_{11}, c_{12}\}$

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$	$c_2 = \neg x_6 \vee x_2$	$c_3 = \neg x_2 \vee x_1$	$c_4 = \neg x_1$
$c_5 = \neg x_6 \vee x_8$	$c_6 = x_6 \vee \neg x_8$	$c_7 = x_2 \vee x_4$	$c_8 = \neg x_4 \vee x_5$
$c_9 = x_7 \vee x_5$	$c_{10} = \neg x_7 \vee x_5$	$c_{11} = \neg x_5 \vee x_3$	$c_{12} = \neg x_3$

$$\mathcal{K} = \{\{c_1, c_2, c_3, c_4\}, \{c_9, c_{10}, c_{11}, c_{12}\}, \{c_3, c_4, c_7, c_8, c_{11}, c_{12}\}\}$$

- Find MHS of $\mathcal{K}$: E.g. $\{c_1, c_9\}$
- SAT($\mathcal{F} \setminus \{c_1, c_9\}$)? No
- Core of $\mathcal{F}$: $\{c_3, c_4, c_7, c_8, c_{11}, c_{12}\}$. Update $\mathcal{K}$

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$ $\quad c_2 = \neg x_6 \vee x_2$ $\quad c_3 = \neg x_2 \vee x_1$ $\quad c_4 = \neg x_1$

$c_5 = \neg x_6 \vee x_8$ $\quad c_6 = x_6 \vee \neg x_8$ $\quad c_7 = x_2 \vee x_4$ $\quad c_8 = \neg x_4 \vee x_5$

$c_9 = x_7 \vee x_5$ $\quad c_{10} = \neg x_7 \vee x_5$ $\quad c_{11} = \neg x_5 \vee x_3$ $\quad c_{12} = \neg x_3$

$$\mathcal{K} = \{\{c_1, c_2, c_3, c_4\}, \{c_9, c_{10}, c_{11}, c_{12}\}, \{c_3, c_4, c_7, c_8, c_{11}, c_{12}\}\}$$

- Find MHS of $\mathcal{K}$:

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$ $\quad c_2 = \neg x_6 \vee x_2$ $\quad c_3 = \neg x_2 \vee x_1$ $\quad c_4 = \neg x_1$

$c_5 = \neg x_6 \vee x_8$ $\quad c_6 = x_6 \vee \neg x_8$ $\quad c_7 = x_2 \vee x_4$ $\quad c_8 = \neg x_4 \vee x_5$

$c_9 = x_7 \vee x_5$ $\quad c_{10} = \neg x_7 \vee x_5$ $\quad c_{11} = \neg x_5 \vee x_3$ $\quad c_{12} = \neg x_3$

$$\mathcal{K} = \{\{c_1, c_2, c_3, c_4\}, \{c_9, c_{10}, c_{11}, c_{12}\}, \{c_3, c_4, c_7, c_8, c_{11}, c_{12}\}\}$$

- Find MHS of $\mathcal{K}$: E.g. $\{c_4, c_9\}$

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$ $\quad c_2 = \neg x_6 \vee x_2$ $\quad c_3 = \neg x_2 \vee x_1$ $\quad c_4 = \neg x_1$

$c_5 = \neg x_6 \vee x_8$ $\quad c_6 = x_6 \vee \neg x_8$ $\quad c_7 = x_2 \vee x_4$ $\quad c_8 = \neg x_4 \vee x_5$

$c_9 = x_7 \vee x_5$ $\quad c_{10} = \neg x_7 \vee x_5$ $\quad c_{11} = \neg x_5 \vee x_3$ $\quad c_{12} = \neg x_3$

$$\mathcal{K} = \{\{c_1, c_2, c_3, c_4\}, \{c_9, c_{10}, c_{11}, c_{12}\}, \{c_3, c_4, c_7, c_8, c_{11}, c_{12}\}\}$$

- Find MHS of $\mathcal{K}$: E.g. $\{c_4, c_9\}$
- SAT$(\mathcal{F} \setminus \{c_4, c_9\})$?

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$	$c_2 = \neg x_6 \vee x_2$	$c_3 = \neg x_2 \vee x_1$	$c_4 = \neg x_1$
$c_5 = \neg x_6 \vee x_8$	$c_6 = x_6 \vee \neg x_8$	$c_7 = x_2 \vee x_4$	$c_8 = \neg x_4 \vee x_5$
$c_9 = x_7 \vee x_5$	$c_{10} = \neg x_7 \vee x_5$	$c_{11} = \neg x_5 \vee x_3$	$c_{12} = \neg x_3$

$$\mathcal{K} = \{\{c_1, c_2, c_3, c_4\}, \{c_9, c_{10}, c_{11}, c_{12}\}, \{c_3, c_4, c_7, c_8, c_{11}, c_{12}\}\}$$

- Find MHS of $\mathcal{K}$: E.g. $\{c_4, c_9\}$
- SAT($\mathcal{F} \setminus \{c_4, c_9\}$)? Yes, e.g. $x_1 = x_2 = 1, x_3 = x_4 = x_5 = x_6 = x_7 = x_8 = 0$

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$	$c_2 = \neg x_6 \vee x_2$	$c_3 = \neg x_2 \vee x_1$	$c_4 = \neg x_1$
$c_5 = \neg x_6 \vee x_8$	$c_6 = x_6 \vee \neg x_8$	$c_7 = x_2 \vee x_4$	$c_8 = \neg x_4 \vee x_5$
$c_9 = x_7 \vee x_5$	$c_{10} = \neg x_7 \vee x_5$	$c_{11} = \neg x_5 \vee x_3$	$c_{12} = \neg x_3$

$$\mathcal{K} = \{\{c_1, c_2, c_3, c_4\}, \{c_9, c_{10}, c_{11}, c_{12}\}, \{c_3, c_4, c_7, c_8, c_{11}, c_{12}\}\}$$

- Find MHS of $\mathcal{K}$: E.g. $\{c_4, c_9\}$
- SAT($\mathcal{F} \setminus \{c_4, c_9\}$)? Yes, e.g. $x_1 = x_2 = 1, x_3 = x_4 = x_5 = x_6 = x_7 = x_8 = 0$
- Terminate & return 2

MaxSAT with Minimum Hitting Sets (MHS)

(Davies&Bacchus,CP'11)

$c_1 = x_6 \vee x_2$	$c_2 = \neg x_6 \vee x_2$	$c_3 = \neg x_2 \vee x_1$	$c_4 = \neg x_1$
$c_5 = \neg x_6 \vee x_8$	$c_6 = x_6 \vee \neg x_8$	$c_7 = x_2 \vee x_4$	$c_8 = \neg x_4 \vee x_5$
$c_9 = x_7 \vee x_5$	$c_{10} = \neg x_7 \vee x_5$	$c_{11} = \neg x_5 \vee x_3$	$c_{12} = \neg x_3$

$$\mathcal{K} = \{\{c_1, c_2, c_3, c_4\}, \{c_9, c_{10}, c_{11}, c_{12}\}, \{c_3, c_4, c_7, c_8, c_{11}, c_{12}\}\}$$

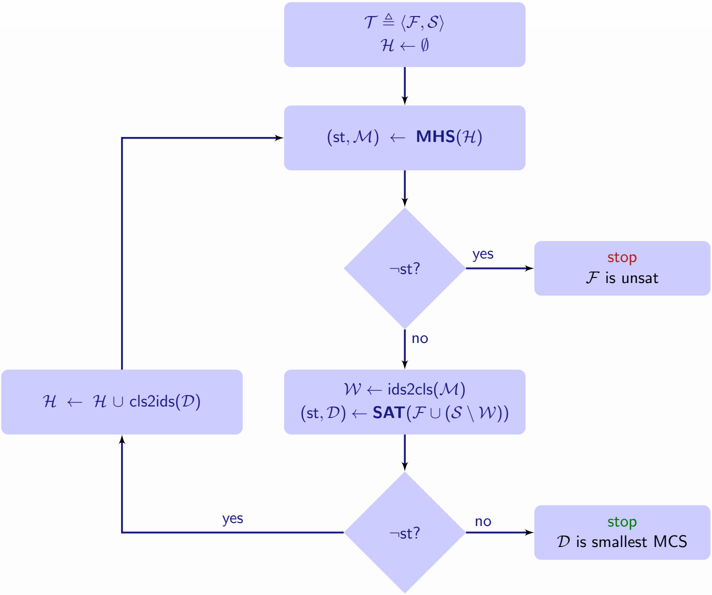
- Find MHS of $\mathcal{K}$: E.g. $\{c_4, c_9\}$
- SAT$(\mathcal{F} \setminus \{c_4, c_9\})$? Yes, e.g. $x_1 = x_2 = 1, x_3 = x_4 = x_5 = x_6 = x_7 = x_8 = 0$
- Terminate & return 2

Possibly many MHSes, with one SAT oracle call for each MHS!

Outline

The MaxHS algorithm

(Davies&Bacchus,CP'11)

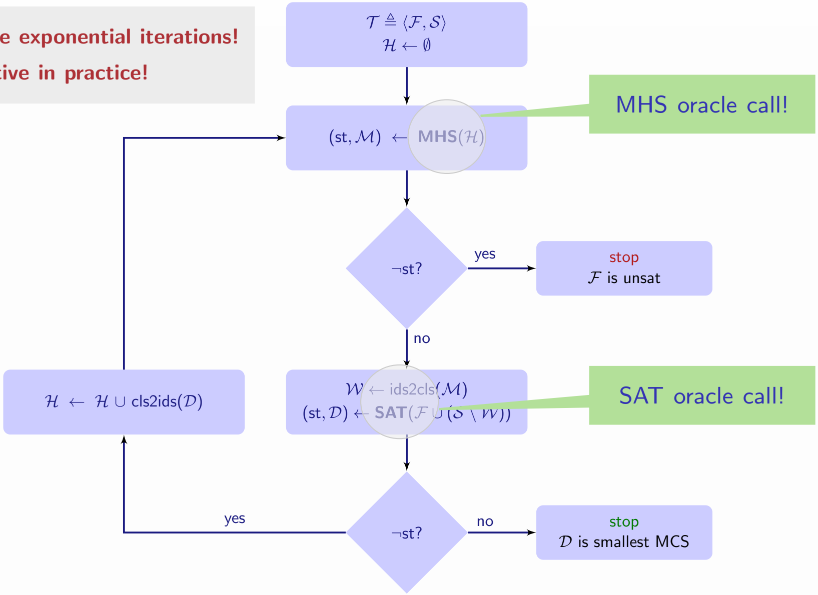


The MaxHS algorithm

(Davies&Bacchus,CP'11)

Worst-case exponential iterations!

But effective in practice!



Outline

MaxSAT Solving

Horn MaxSAT

PHP Refutations in Polynomial Time

Recap Horn MaxSAT

- What is Horn MaxSAT?
  - All soft clauses are Horn
    - Most often, unit soft clauses
  - All hard clauses are Horn

Recap Horn MaxSAT

- What is Horn MaxSAT?
  - All soft clauses are Horn
    - Most often, unit soft clauses
  - All hard clauses are Horn

- How hard is Horn MaxSAT?
  - Horn MaxSAT is NP-hard [JS87]
  - Decision K-HornSAT is NP-complete [JS87]
    - **By definition, any problem in NP is reducible to K-HornSAT**
    - But ...

Why use Horn MaxSAT?

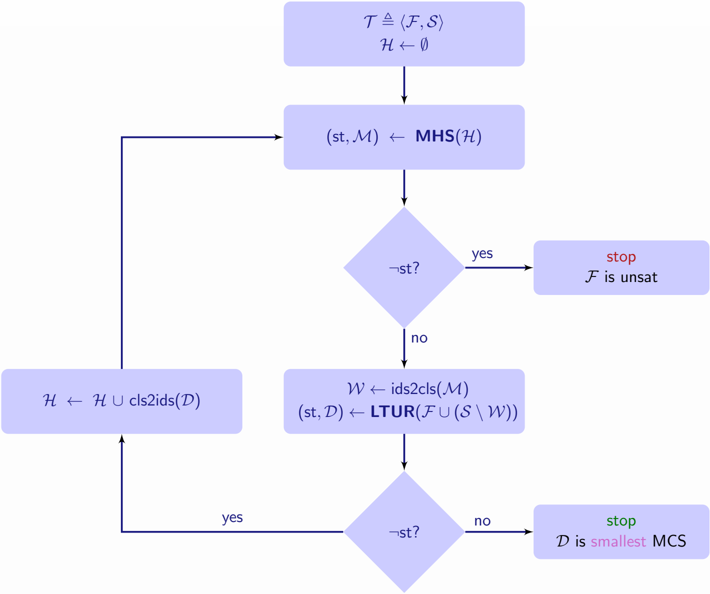
- **Practical perspective:**
  - MaxSAT with MHSes is **very efficient** in practice
  - For Horn MaxSAT, we can replace SAT call (worst-case exponential) with LTUR call (worst-case **linear**)

Why use Horn MaxSAT?

- **Practical perspective:**
  - MaxSAT with MHSes is **very efficient** in practice
  - For Horn MaxSAT, we can replace SAT call (worst-case exponential) with LTUR call (worst-case **linear**)

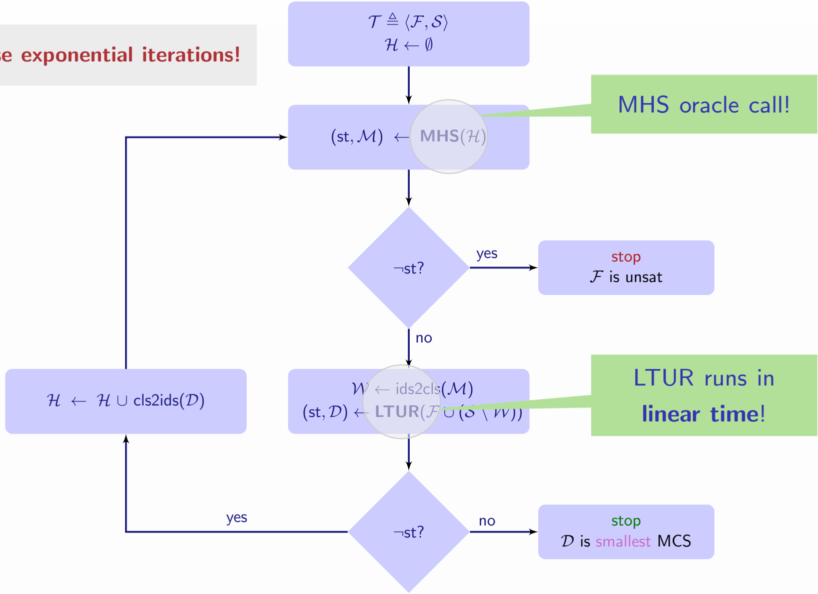
- **Theoretical perspective:**
  - Reducing SAT to Horn MaxSAT & applying a MaxSAT algorithm yields new proof system(s)
    - MaxSAT resolution
    - Core-guided algorithm(s)
    - MaxHS-like algorithms
    - ...
  - Reducing PHP to SAT and then to Horn MaxSAT admits polynomial time refutations for **some** MaxSAT algorithms

A Horn MaxHS algorithm



A Horn MaxHS algorithm

Worst-case exponential iterations!



What can we solve with Horn MaxSAT?

[IMMS17,MSIM17]

SAT $\leq_P$ Horn MaxSAT	and so CSP, ASP, SMT*, ...
CSP $\leq_P$ Horn MaxSAT	direct, besides CSP $\leq_P$ SAT
PHP $\leq_P$ Horn MaxSAT	direct, besides PHP $\leq_P$ SAT
MaxClique $\leq_P$ Horn MaxSAT	and so MinVC, MaxIS
MinHS $\leq_P$ Horn MaxSAT	and so MaxSP
MinDS $\leq_P$ Horn MaxSAT	

What can we solve with Horn MaxSAT?

[IMMS17,MSIM17]

SAT $\leq_P$ Horn MaxSAT	and so CSP, ASP, SMT*, ...
CSP $\leq_P$ Horn MaxSAT	direct, besides CSP $\leq_P$ SAT
PHP $\leq_P$ Horn MaxSAT	direct, besides PHP $\leq_P$ SAT
MaxClique $\leq_P$ Horn MaxSAT	and so MinVC, MaxIS
MinHS $\leq_P$ Horn MaxSAT	and so MaxSP
MinDS $\leq_P$ Horn MaxSAT	

- **Most** encodings of cardinality constraints are Horn
  - Sequential counters; totalizers; sorting networks; (pairwise) (cardinality networks); bitwise (for AtMost1) [S05,ES06,ANORC11,...]

What can we solve with Horn MaxSAT?

[IMMS17,MSIM17]

SAT $\leq_P$ Horn MaxSAT	and so CSP, ASP, SMT*, ...
CSP $\leq_P$ Horn MaxSAT	direct, besides CSP $\leq_P$ SAT
PHP $\leq_P$ Horn MaxSAT	direct, besides PHP $\leq_P$ SAT
MaxClique $\leq_P$ Horn MaxSAT	and so MinVC, MaxIS
MinHS $\leq_P$ Horn MaxSAT	and so MaxSP
MinDS $\leq_P$ Horn MaxSAT	

- **Most** encodings of cardinality constraints are Horn
  - Sequential counters; totalizers; sorting networks; (pairwise) (cardinality networks); bitwise (for AtMost1) [S05,ES06,ANORC11,...]
- **Some** encodings of pseudo-Boolean constraints are Horn
  - Local polynomial watchdog (LPW) [BBR09]

What can we solve with Horn MaxSAT?

[IMMS17,MSIM17]

SAT $\leq_P$ Horn MaxSAT — and so CSP, ASP, SMT*, ...

CSP $\leq_P$ Horn MaxSAT — direct, besides CSP $\leq_P$ SAT

PHP $\leq_P$ Horn MaxSAT — direct, besides PHP $\leq_P$ SAT

MaxClique $\leq_P$ Horn MaxSAT — and so MinVC, MaxIS

MinHS $\leq_P$ Horn MaxSAT — and so MaxSP

MinDS $\leq_P$ Horn MaxSAT

- **Most** encodings of cardinality constraints are Horn
  - Sequential counters; totalizers; sorting networks; (pairwise) (cardinality networks); bitwise (for AtMost1) [S05,ES06,ANORC11,...]
- **Some** encodings of pseudo-Boolean constraints are Horn
  - Local polynomial watchdog (LPW) [BBR09]

Knapsack $\leq_P$ Horn MaxSAT

What can we solve with Horn MaxSAT?

[IMMS17,MSIM17]

SAT $\leq_P$ Horn MaxSAT	and so CSP, ASP, SMT*, ...
CSP $\leq_P$ Horn MaxSAT	direct, besides CSP $\leq_P$ SAT
PHP $\leq_P$ Horn MaxSAT	direct, besides PHP $\leq_P$ SAT
MaxClique $\leq_P$ Horn MaxSAT	and so MinVC, MaxIS
MinHS $\leq_P$ Horn MaxSAT	and so MaxSP
MinDS $\leq_P$ Horn MaxSAT	

- **Most** encodings of cardinality constraints are Horn
  - Sequential counters; totalizers; sorting networks; (pairwise) (cardinality networks); bitwise (for AtMost1) [S05,ES06,ANORC11,...]
- **Some** encodings of pseudo-Boolean constraints are Horn
  - Local polynomial watchdog (LPW) [BBR09]

Knapsack $\leq_P$ Horn MaxSAT

- **Horn MaxSAT: enables general-purpose problem solving**

Outline

SAT reduces to Horn MaxSAT

$$\mathcal{F} \triangleq (x_1 \vee \neg x_2 \vee x_3) \wedge (x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_3)$$

SAT reduces to Horn MaxSAT

$$\mathcal{F} \triangleq (x_1 \vee \neg x_2 \vee x_3) \wedge (x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_3)$$

- For each x_i, create new variables p_i (for $x_i = 1$) and n_i (for $x_i = 0$)
- p_i and n_i cannot both be assigned 1:
  - Add hard clause $(\neg p_i \vee \neg n_i)$

SAT reduces to Horn MaxSAT

$$\mathcal{F} \triangleq (x_1 \lor \neg x_2 \lor x_3) \land (x_2 \lor x_3) \land (\neg x_1 \lor \neg x_3)$$

- For each x_i, create new variables p_i (for $x_i = 1$) and n_i (for $x_i = 0$)
- p_i and n_i cannot both be assigned 1:
  - Add hard clause $(\neg p_i \lor \neg n_i)$
- Reencode original clauses (as hard clauses):
  - Literal x_i replaced by $\neg n_i$
  - Literal $\neg x_i$ replaced by $\neg p_i$
- Goal is to assign value 1 to each variable, if possible:
  - Add soft clauses (p_i) and (n_i)

SAT reduces to Horn MaxSAT

$$\mathcal{F} \triangleq (x_1 \vee \neg x_2 \vee x_3) \wedge (x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_3)$$

- For each x_i, create new variables p_i (for $x_i = 1$) and n_i (for $x_i = 0$)
- p_i and n_i cannot both be assigned 1:
  - Add hard clause $(\neg p_i \vee \neg n_i)$
- Reencode original clauses (as hard clauses):
  - Literal x_i replaced by $\neg n_i$
  - Literal $\neg x_i$ replaced by $\neg p_i$
- Goal is to assign value 1 to each variable, if possible:
  - Add soft clauses (p_i) and (n_i)

- All clauses are Horn

SAT reduces to Horn MaxSAT

$$\mathcal{F} \triangleq (x_1 \vee \neg x_2 \vee x_3) \wedge (x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_3)$$

- For each x_i, create new variables p_i (for $x_i = 1$) and n_i (for $x_i = 0$)
- p_i and n_i cannot both be assigned 1:
  - Add hard clause $(\neg p_i \vee \neg n_i)$
- Reencode original clauses (as hard clauses):
  - Literal x_i replaced by $\neg n_i$
  - Literal $\neg x_i$ replaced by $\neg p_i$
- Goal is to assign value 1 to each variable, if possible:
  - Add soft clauses (p_i) and (n_i)

- All clauses are Horn

- Original formula is satisfiable iff Horn MaxSAT formula can satisfy n soft clauses (and the hard clauses)
  - I.e., satisfying n soft clauses represents assignment to the n variables consistent with the original clauses **!**

SAT reduces to Horn MaxSAT (Cont.)

$$\mathcal{F} \triangleq (x_1 \vee \neg x_2 \vee x_3) \wedge (x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_3)$$

- Example:
  - New variables: $p_1, p_2, p_3, n_1, n_2, n_3$
  - Filter impossible assignments: $\{(\neg p_1 \vee \neg n_1), (\neg p_2 \vee \neg n_2), (\neg p_3 \vee \neg n_3)\}$
  - Original clauses reencoded: $(\neg n_1 \vee \neg p_2 \vee \neg n_3) \wedge (\neg n_2 \vee \neg n_3) \wedge (\neg p_1 \vee \neg p_3)$
  - Soft clauses: $\{(p_1), (p_2), (p_3), (n_1), (n_2), (n_3)\}$

SAT reduces to Horn MaxSAT (Cont.)

$$\mathcal{F} \triangleq (x_1 \vee \neg x_2 \vee x_3) \wedge (x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_3)$$

- Example:
  - New variables: $p_1, p_2, p_3, n_1, n_2, n_3$
  - Filter impossible assignments: $\{(\neg p_1 \vee \neg n_1), (\neg p_2 \vee \neg n_2), (\neg p_3 \vee \neg n_3)\}$
  - Original clauses reencoded: $(\neg n_1 \vee \neg p_2 \vee \neg n_3) \wedge (\neg n_2 \vee \neg n_3) \wedge (\neg p_1 \vee \neg p_3)$
  - Soft clauses: $\{(p_1), (p_2), (p_3), (n_1), (n_2), (n_3)\}$
- Encoding is a variant of the dual-rail encoding, used since the mid 80s [BBBCS87]

Pigeonhole formulas – propositional encoding PHP_m^{m+1}

- Variables:
  - $x_{ij} = 1$ iff the i^{th} pigeon is placed in the j^{th} hole, $1 \leq i \leq m+1$, $1 \leq j \leq m$

Pigeonhole formulas – propositional encoding PHP_m^{m+1}

- Variables:
  - $x_{ij} = 1$ iff the i^{th} pigeon is placed in the j^{th} hole, $1 \leq i \leq m+1$, $1 \leq j \leq m$
- Constraints:
  - Each pigeon **must** be placed in at least one hole, and each hole **must not** have more than one pigeon

$$\bigwedge_{i=1}^{m+1} \mathrm{AtLeast1}(x_{i1}, \ldots, x_{im}) \wedge \bigwedge_{j=1}^{m} \mathrm{AtMost1}(x_{1j}, \ldots, x_{m+1\,j})$$

Pigeonhole formulas – propositional encoding PHP_m^{m+1}

- Variables:
  - $x_{ij} = 1$ iff the i^{th} pigeon is placed in the j^{th} hole, $1 \leq i \leq m+1$, $1 \leq j \leq m$
- Constraints:
  - Each pigeon **must** be placed in at least one hole, and each hole **must not** have more than one pigeon

$$\bigwedge_{i=1}^{m+1} \mathrm{AtLeast1}(x_{i1}, \ldots, x_{im}) \wedge \bigwedge_{j=1}^{m} \mathrm{AtMost1}(x_{1j}, \ldots, x_{m+1\,j})$$

- Example encoding, with pairwise encoding for AtMost1 constraint:

Constraint	Clause(s)
$\wedge_{i=1}^{m+1}\mathrm{AtLeast1}(x_{i1}, \ldots, x_{im})$	$(x_{i1} \vee \ldots \vee x_{im})$
$\wedge_{j=1}^{m}\mathrm{AtMost1}(x_{1j}, \ldots, x_{m+1\,j})$	$\wedge_{r=2}^{m+1} \wedge_{s=1}^{r-1} (\neg x_{rj} \vee \neg x_{sj})$

PHP as Horn MaxSAT

- New variables n_{ij} and p_{ij}, for each x_{ij}, $1 \leq i \leq m+1, 1 \leq j \leq m$
- The soft clauses $\mathcal{S}$, with $|\mathcal{S}| = 2m(m+1)$, are given by

$$\begin{array}{ll}\{ & (n_{11}), \ldots, (n_{1m}), \ldots, (n_{m+1\,1}), \ldots, (n_{m+1\,m}), \\ & (p_{11}), \ldots, (p_{1m}), \ldots, (p_{m+1\,1}), \ldots, (p_{m+1\,m}) \quad \}\end{array}$$

PHP as Horn MaxSAT

- New variables n_{ij} and p_{ij}, for each x_{ij}, $1 \leq i \leq m+1, 1 \leq j \leq m$
- The soft clauses $\mathcal{S}$, with $|\mathcal{S}| = 2m(m+1)$, are given by

$$\begin{array}{ll}\{ & (n_{11}), \ldots, (n_{1m}), \ldots, (n_{m+1\,1}), \ldots, (n_{m+1\,m}), \\ & (p_{11}), \ldots, (p_{1m}), \ldots, (p_{m+1\,1}), \ldots, (p_{m+1\,m}) \quad \}\end{array}$$

- Clauses in $\mathcal{P}$: $\mathcal{P} = \{(\neg n_{ij} \vee \neg p_{ij}) \,|\, 1 \leq i \leq m+1, 1 \leq j \leq m\}$

PHP as Horn MaxSAT

- New variables n_{ij} and p_{ij}, for each x_{ij}, $1 \leq i \leq m+1, 1 \leq j \leq m$
- The soft clauses $\mathcal{S}$, with $|\mathcal{S}| = 2m(m+1)$, are given by

$$\begin{aligned} \{ \quad & (n_{11}), \ldots, (n_{1m}), \ldots, (n_{m+1\,1}), \ldots, (n_{m+1\,m}), \\ & (p_{11}), \ldots, (p_{1m}), \ldots, (p_{m+1\,1}), \ldots, (p_{m+1\,m}) \quad \} \end{aligned}$$

- Clauses in $\mathcal{P}$: $\mathcal{P} = \{(\neg n_{ij} \vee \neg p_{ij}) \,|\, 1 \leq i \leq m+1, 1 \leq j \leq m\}$
- AtLeast1 constraints encoded as $\mathcal{L}_i, 1 \leq i \leq m+1$
- AtMost1 constraints encoded as $\mathcal{M}_j, 1 \leq j \leq m$

PHP as Horn MaxSAT

- New variables n_{ij} and p_{ij}, for each x_{ij}, $1 \leq i \leq m+1, 1 \leq j \leq m$
- The soft clauses $\mathcal{S}$, with $|\mathcal{S}| = 2m(m+1)$, are given by

$$\begin{array}{ll}\{ & (n_{11}), \ldots, (n_{1m}), \ldots, (n_{m+1\,1}), \ldots, (n_{m+1\,m}), \\ & (p_{11}), \ldots, (p_{1m}), \ldots, (p_{m+1\,1}), \ldots, (p_{m+1\,m}) \quad \}\end{array}$$

- Clauses in $\mathcal{P}$: $\mathcal{P} = \{(\neg n_{ij} \vee \neg p_{ij}) \,|\, 1 \leq i \leq m+1, 1 \leq j \leq m\}$
- AtLeast1 constraints encoded as $\mathcal{L}_i, 1 \leq i \leq m+1$
- AtMost1 constraints encoded as $\mathcal{M}_j, 1 \leq j \leq m$
- Full reduction of PHP to Horn MaxSAT

$$\langle \mathcal{H}, \mathcal{S} \rangle = \left\langle \wedge_{i=1}^{m+1} \mathcal{L}_i \wedge \wedge_{j=1}^{m} \mathcal{M}_j \wedge \mathcal{P}, \mathcal{S} \right\rangle$$

PHP as Horn MaxSAT

- New variables n_{ij} and p_{ij}, for each x_{ij}, $1 \leq i \leq m+1, 1 \leq j \leq m$
- The soft clauses $\mathcal{S}$, with $|\mathcal{S}| = 2m(m+1)$, are given by

$$\begin{array}{ll} \{ & (n_{11}), \ldots, (n_{1m}), \ldots, (n_{m+1\,1}), \ldots, (n_{m+1\,m}), \\ & (p_{11}), \ldots, (p_{1m}), \ldots, (p_{m+1\,1}), \ldots, (p_{m+1\,m}) \quad \} \end{array}$$

- Clauses in $\mathcal{P}$: $\mathcal{P} = \{(\neg n_{ij} \vee \neg p_{ij}) \,|\, 1 \leq i \leq m+1, 1 \leq j \leq m\}$
- AtLeast1 constraints encoded as $\mathcal{L}_i, 1 \leq i \leq m+1$
- AtMost1 constraints encoded as $\mathcal{M}_j, 1 \leq j \leq m$
- Full reduction of PHP to Horn MaxSAT

$$\langle \mathcal{H}, \mathcal{S} \rangle = \left\langle \wedge_{i=1}^{m+1} \mathcal{L}_i \wedge \wedge_{j=1}^{m} \mathcal{M}_j \wedge \mathcal{P}, \mathcal{S} \right\rangle$$

- **No** more than $m(m+1)$ clauses can be satisfied, due to $\mathcal{P}$
- PHP_m^{m+1} is satisfiable iff there exists an assignment that satisfies the hard clauses $\mathcal{H}$ and $m(m+1)$ soft clauses from $\mathcal{S}$

PHP as Horn MaxSAT II

- Clauses in each $\mathcal{L}_i$ and in each $\mathcal{M}_j$, with pairwise encoding

Original Constraint	Encoded To	Clauses
$\wedge_{i=1}^{m+1}\text{AtLeast1}(x_{i1},\ldots,x_{im})$	$\mathcal{L}_i$	$(\neg n_{i1} \vee \ldots \vee \neg n_{im})$
$\wedge_{j=1}^{m}\text{AtMost1}(x_{1j},\ldots,x_{m+1,j})$	$\mathcal{M}_j$	$\wedge_{r=2}^{m+1} \wedge_{s=1}^{r-1} (\neg p_{rj} \vee \neg p_{sj})$

PHP as Horn MaxSAT II

- Clauses in each $\mathcal{L}_i$ and in each $\mathcal{M}_j$, with pairwise encoding

Original Constraint	Encoded To	Clauses
$\wedge_{i=1}^{m+1}\text{AtLeast1}(x_{i1},\ldots,x_{im})$	$\mathcal{L}_i$	$(\neg n_{i1} \vee \ldots \vee \neg n_{im})$
$\wedge_{j=1}^{m}\text{AtMost1}(x_{1j},\ldots,x_{m+1,j})$	$\mathcal{M}_j$	$\wedge_{r=2}^{m+1} \wedge_{s=1}^{r-1} (\neg p_{rj} \vee \neg p_{sj})$

- **Note:** constraints with key structural properties:

Constraint	Variables
$\mathcal{L}_i$	$(\neg n_{i1} \vee \ldots \vee \neg n_{im})$
$\mathcal{L}_k$	$(\neg n_{k1} \vee \ldots \vee \neg n_{km})$
$\mathcal{M}_j$	$\wedge_{r=2}^{m+1} \wedge_{s=1}^{r-1} (\neg p_{rj} \vee \neg p_{sj})$
$\mathcal{M}_l$	$\wedge_{r=2}^{m+1} \wedge_{s=1}^{r-1} (\neg p_{rl} \vee \neg p_{sl})$

  - Variables in each $\mathcal{L}_i$ **disjoint** from **any** other $\mathcal{L}_k$ and $\mathcal{M}_j$, $k \neq i$
  - Variables in each $\mathcal{M}_j$ **disjoint** from **any** other $\mathcal{M}_l$, $l \neq j$

Outline

Some results from our SAT'17 paper

Claim 1

Core-guided MaxSAT (e.g. MSU3) produces a lower bound on the number of falsified clauses $\geq m(m+1)+1$ in polynomial time

Claim 2

MaxSAT resolution produces a lower bound on the number of falsified clauses $\geq m(m+1)+1$ in polynomial time

Remark

Horn MaxSAT encoding enables polynomial time refutations of the unsatisfiability of PHP instances, using CDCL SAT solvers

Proof of claim 1 – outline

1. Assume MSU3 MaxSAT algorithm
   - **Note:** Suffices to analyze disjoint sets separately

Proof of claim 1 – outline

1. Assume MSU3 MaxSAT algorithm
   – **Note:** Suffices to analyze disjoint sets separately
2. Relate soft clauses with each $\mathcal{L}_i$ and each $\mathcal{M}_j$
   – **Recall**: each constraint disjoint from the others (but not from $\mathcal{P}$)

Proof of claim 1 – outline

1. Assume MSU3 MaxSAT algorithm
   - **Note:** Suffices to analyze disjoint sets separately
2. Relate soft clauses with each $\mathcal{L}_i$ and each $\mathcal{M}_j$
   - **Recall**: each constraint disjoint from the others (but not from $\mathcal{P}$)
3. Derive large enough lower bound on # of falsified clauses:

Constr. type	# falsified cls	# constr	In total
$\mathcal{L}_i$	1	$i = 1, \ldots, m+1$	$m+1$
$\mathcal{M}_j$	m	$j = 1, \ldots, m$	$m \cdot m$
			$m(m+1)+1$

Proof of claim 1 – outline

1. Assume MSU3 MaxSAT algorithm
   - **Note:** Suffices to analyze disjoint sets separately
2. Relate soft clauses with each $\mathcal{L}_i$ and each $\mathcal{M}_j$
   - **Recall**: each constraint disjoint from the others (but not from $\mathcal{P}$)
3. Derive large enough lower bound on # of falsified clauses:

Constr. type	# falsified cls	# constr	In total
$\mathcal{L}_i$	1	$i = 1, \ldots, m+1$	$m+1$
$\mathcal{M}_j$	m	$j = 1, \ldots, m$	$m \cdot m$
			$m(m+1)+1$

4. Each increase in the value of the lower bound obtained by unit propagation (UP)
   - In total: polynomial number of (linear time) UP runs

Proof of claim 1 – unit propagation steps I

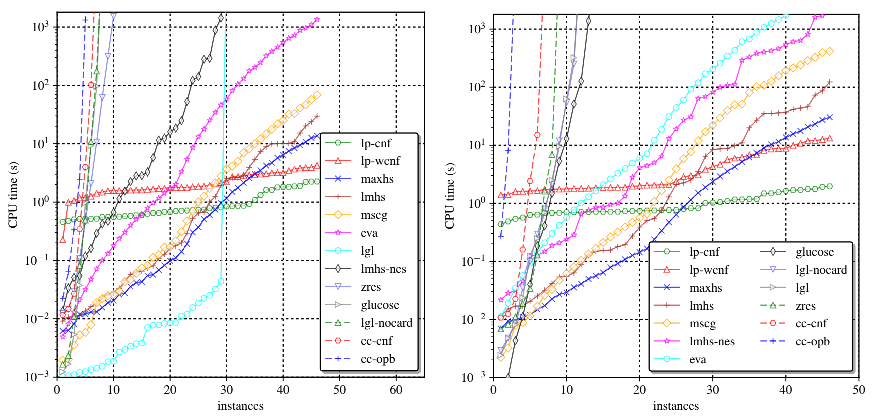
Constr	Hard cls	Soft cls	Relaxed clauses	Updated AtMost$k$ constr	LB incr
$\mathcal{L}_i$	$(\neg n_{i1} \vee \ldots \vee \neg n_{im})$	$(n_{i1}), \ldots, (n_{im})$	$(s_{il} \vee n_{i1})$, $1 \leq l \leq m$	$\sum_{l=1}^{m} s_{il} \leq 1$	1
$\mathcal{M}_j$	$(\neg p_{1j} \vee \neg p_{2j})$	$(p_{1j}), (p_{2j})$	$(r_{1j} \vee p_{1j})$, $(r_{2j} \vee p_{2j})$	$\sum_{l=1}^{2} r_{lj} \leq 1$	1
$\mathcal{M}_j$	$(\neg p_{1j} \vee \neg p_{3j})$, $(\neg p_{2j} \vee \neg p_{3j})$, $(r_{1j} \vee p_{1j})$, $(r_{2j} \vee p_{2j})$, $\sum_{l=1}^{2} r_{lj} \leq 1$	(p_{3j})	$(r_{3j} \vee p_{3j})$	$\sum_{l=1}^{3} r_{lj} \leq 2$	1
		$\ldots$			
$\mathcal{M}_j$	$(\neg p_{1j} \vee \neg p_{m+1j}), \ldots,$ $(\neg p_{mj} \vee \neg p_{m+1j})$, $(r_{1j} \vee p_{1j}), \ldots,$ $(r_{mj} \vee p_{mj})$, $\sum_{l=1}^{m} r_{lj} \leq m-1$	(p_{m+1j})	$(r_{m+1j} \vee p_{m+1j})$	$\sum_{l=1}^{m+1} r_{lj} \leq m$	1

Proof of claim 1 – unit propagation steps II

Clauses	Unit Propagation
$(p_{k+1\,j})$	$p_{k+1\,j} = 1$
$(\neg p_{1j} \vee \neg p_{k+1\,j}), \ldots, (\neg p_{kj} \vee \neg p_{k+1\,j})$	$p_{1j} = \ldots = p_{kj} = 0$
$(r_{1j} \vee p_{1j}), \ldots, (r_{kj} \vee p_{kj})$	$r_{1j} = \ldots = r_{kj} = 1$
$\sum_{l=1}^{k} r_{lj} \leq k-1$	$\left(\sum_{l=1}^{k} r_{lj} \leq k-1\right) \vdash_1 \perp$

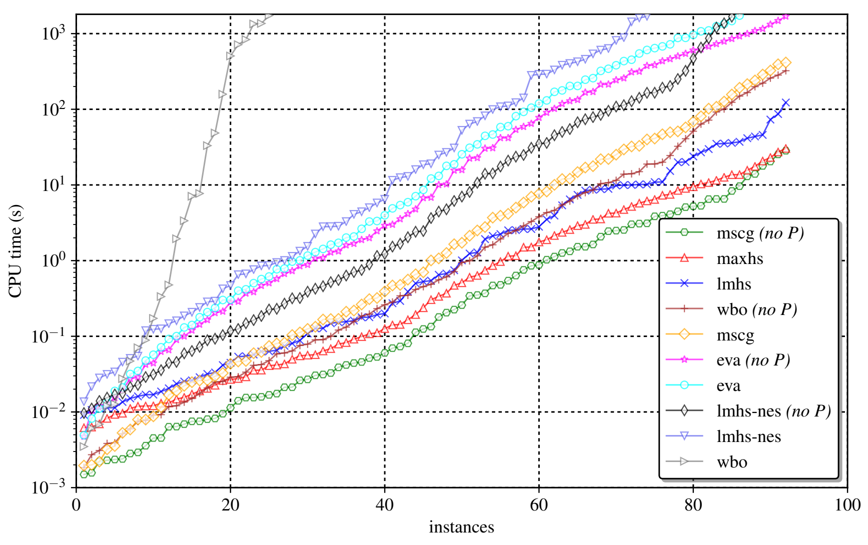
- Key points:
  - For each $\mathcal{L}_i$, UP raises LB by 1
  - For each $\mathcal{M}_j$, UP raises LB by m
  - In total, UP raises LB by $m(m+1)+1$
  - Thus, PHP_m^{m+1} is **unsatisfiable**

Results on PHP instances: pw vs. sc



SAT		**SAT+**		**IHS MaxSAT**		**CG MaxSAT**			**MRes**	**MIP**	**OPB**		**BDD**
minisat	*glucose*	*lgl*	crypto	*maxhs*	*lmhs*	*mscg*	*wbo*	wpm3	*eva*	*lp*	*cc*	sat4j*	*zres*

Effect of $\mathcal{P}$ clauses



Some results from our AAAI'18 paper – see MLB's talk

Remark

Formalize DrMaxSAT proof system, using MaxSAT resolution

Result 1

DrMaxSAT p-simulates RES/CL

$\therefore$ DrMaxSAT stronger proof system than RES/CL

Result 2

MaxSAT refutations of the dual-rail encoded Parity Principle require exponential size 2^{n^ϵ} for some $\epsilon > 0$

$\therefore$ DrMaxSAT does not p-simulate CP

But, several open questions ...

Conclusions & research directions

- Initial motivation: optimize MaxHS-like algorithms
  - E.g. by exploiting Horn MaxSAT & LTUR

Conclusions & research directions

- Initial motivation: optimize MaxHS-like algorithms
  - E.g. by exploiting Horn MaxSAT & LTUR
- Simple reduction from SAT to Horn MaxSAT
  - Many other simple reductions to Horn MaxSAT

Conclusions & research directions

- Initial motivation: optimize MaxHS-like algorithms
  - E.g. by exploiting Horn MaxSAT & LTUR
- Simple reduction from SAT to Horn MaxSAT
  - Many other simple reductions to Horn MaxSAT
- (Horn) MaxSAT solvers can solve (in polynomial time) hard instances for resolution
  - If equipped with the right reduction

Conclusions & research directions

- Initial motivation: optimize MaxHS-like algorithms
  - E.g. by exploiting Horn MaxSAT & LTUR
- Simple reduction from SAT to Horn MaxSAT
  - Many other simple reductions to Horn MaxSAT
- (Horn) MaxSAT solvers can solve (in polynomial time) hard instances for resolution
  - If equipped with the right reduction
- Where to go with Horn MaxSAT?
  - Also, additional results about the new proof system(s)?

Conclusions & research directions

- Initial motivation: optimize MaxHS-like algorithms
  - E.g. by exploiting Horn MaxSAT & LTUR
- Simple reduction from SAT to Horn MaxSAT
  - Many other simple reductions to Horn MaxSAT
- (Horn) MaxSAT solvers can solve (in polynomial time) hard instances for resolution
  - If equipped with the right reduction
- Where to go with Horn MaxSAT?
  - Also, additional results about the new proof system(s)?
- Still many open questions?
  - E.g. MaxHS unreasonably efficient. **Why?**

Questions?

Some references

- A. Ignatiev, A. Morgado, J. Marques-Silva:
  On Tackling the Limits of Resolution in SAT Solving.
  SAT 2017: 164-183

- J. Marques-Silva, A. Ignatiev, A. Morgado:
  Horn Maximum Satisfiability:
  Reductions, Algorithms and Applications.
  EPIA 2017: 681-694

- M.L. Bonet, S. Buss, A. Ignatiev, J. Marques-Silva, A. Morgado:
  MaxSAT Resolution With the Dual Rail Encoding.
  AAAI 2018: 6565-6572