Towards a theory of good SAT representations

Oliver Kullmann

Computer Science Department
Swansea University
http://cs.swan.ac.uk/~csoliver/papers.html

Exploring Computational Complexity
November 9, 2014
Clause-sets

- Let $\mathcal{V}A$ be the set of variables.
- Let $\mathcal{LIT}$ be the set of literals, which are either variables or complemented variables, i.e., $\mathcal{LIT} = \mathcal{V}A \cup \overline{\mathcal{V}A}$.
- A clause is a finite and complement-free subset of $\mathcal{LIT}$, the set of all clauses is $\mathcal{CL}$.
- Let $\mathcal{CLS}$ be the set of clause-sets, finite subsets of $\mathcal{CL}$.

\[
\bot := \emptyset \in \mathcal{CL} \\
\top := \emptyset \in \mathcal{CLS}.
\]
“Hardness”

\[
\begin{align*}
\text{hd} & : \mathcal{CLS} \rightarrow \mathbb{N}_0 \\
\text{awid} & : \mathcal{CLS} \rightarrow \mathbb{N}_0.
\end{align*}
\]

“Hardness” for historical reasons; \( \text{hd} = \text{thd} \).

Open Problem

*Develop a general framework for “hardness measures”.*

*Our approach:*

1. **Precise**, not asymptotic: clause-sets have an intrinsic “hardness”, under a certain perspective; for example Horn clause-sets have hardness 1.

2. **SAT** by worst-case from UNSAT.

Our “hardness” is “hardness for very simple, oblivious SAT algorithms”.

What do you mean?

What does it mean that we have

\[ \text{hd}(F) = k \]

resp.

\[ \text{awid}(F) = k \]

where, just to emphasise, \( F \in \mathcal{CLS} \) is a (single, concrete) clause-set (no hidden parameters), and \( k \in \mathbb{N}_0 \) is a natural number (again, no hidden parameters)?

From a complexity-theory perspective this looks suspicious?

The meaning is, roughly, that with a generic algorithm with time \( n^{O(k)} \) and space \( n^{O(1)} \) resp. \( n^{O(k)} \) all “implicit information” of \( F \) can be uncovered.

\( k \) is a structural parameter of \( F \), measuring at which maximal “level” we can extract prime implicates from \( F \).
Outline

1. Introduction
2. Hardness measures
3. Hierarchies
4. Separations
5. Monotone circuits
6. Conclusion
From USAT to SAT

- Let $\textit{USAT} := \textit{CLS} \setminus \textit{SAT}$.
- Let $\textit{PASS}$ be the set of partial assignments.
- For $\varphi \in \textit{PASS}$ and $F \in \textit{CLS}$ let $\varphi \ast F \in \textit{CLS}$ be the result of applying $\varphi$ to $F$.

In Beyersdorff and Kullmann [4] the following approach was formally introduced:

Consider $h_0 : \textit{USAT} \rightarrow \mathbb{N}_0$.

We extend to $h : \textit{CLS} \rightarrow \mathbb{N}_0$ by

$$h(F) := \max \{ h_0(\varphi \ast F) : \varphi \in \textit{PASS} \land \varphi \ast F \in \textit{USAT} \}.$$ 

If we assume that applying partial assignments does no increase $h_0$ (and this we always do), then this holds also for $h$. 
Game characterisations of hardness’s I

We characterise $\text{hd}(F)$ and $\text{awid}(F)$, indeed for arbitrary $F \in \mathcal{CLS}$, by a game according to [4], extending

- Pudlák and Impagliazzo [11]
- and Atserias and Dalmau [1].
The general structure of our games is:

- There is a global partial assignment $\theta$; initially $\theta$ is empty.
- There are two players, DELAYER and FINISHER, manipulating $\theta$.
- FINISHER seeks $\theta \ast F = \top$ or $\bot \in \theta \ast F$ — once established, the game ends (and it will end).
- DELAYER starts, and must never have $\theta \ast F = \top$ or $\bot \in \theta \ast F$.
- A move of DELAYER is to extend $\theta$ to $\theta' \supseteq \theta$ with $\theta' \ast F \neq \top$ and $\bot \notin \theta \ast F$.
- FINISHER can always extend $\theta$ to any $\theta' \supseteq \theta$ with $\theta' \ast F = \top$, and then DELAYER gets zero points.
- DELAYER maximises points, FINISHER minimises.

The variations concern the moves of FINISHER and the accounting of points when $\bot$ has been created.
The hardness game

The Finisher extends $\theta$ to $\theta'$ with exactly one more assignment, i.e., $n(\theta') = n(\theta) + 1$.

The points Delayer obtains is the number of rounds.

1. The optimal value of this game is $hd(F)$.
2. There are quite a few equivalent characterisations: see [7, 8, 9, 4].
The asymmetric-width game

The **Finisher** first restricts $\theta$ to $\theta' \subseteq \theta$, and then extends $\theta'$ to $\theta''$, not contradicting $\theta$, with $n(\theta'') = n(\theta') + 1$.

The points **Delay**er obtains is the maximal $n(\theta'')$ after moves of **Finisher**.

1. The optimal value of this game is $\text{awid}(F)$.

2. Again, there are quite a few equivalent characterisations.

**Finisher** might always remove precisely all assignments made by **Delay**er, and in this way we can prove:

$$\forall F \in \mathcal{LS} : \text{awid}(F) \leq \text{hd}(F).$$
Relations to resolution complexity

For $F \in USAT$ holds:

$$2^{\text{hd}(F)} \leq \text{Comp}^*_R(F) \leq (n(F) + 1)^{\text{hd}(F)}$$

$$\exp\left(\frac{1}{8} \frac{\text{awid}(F)^2}{n(F)}\right) < \text{Comp}_R(F) < 6 \cdot n(F)^{\text{awid}(F) + 2}$$

where

- $\text{Comp}^*_R(F)$ is the minimal number of leaves in a tree resolution refutation of $F$;
- $\text{Comp}_R(F)$ is the minimal number of nodes in a dag resolution refutation of $F$. 
For $k \in \mathbb{N}_0$:

$$\mathcal{UC}_k := \{ F \in \mathcal{CLS} : \text{hd}(F) \leq k \}$$

$$\mathcal{WC}_k := \{ F \in \mathcal{CLS} : \text{awid}(F) \leq k \}.$$  

We have $F \in \mathcal{UC}_k$ resp. $F \in \mathcal{WC}_k$ iff for all prime implicants $C$ of $F$ there is a tree/dag resolution derivation of $C$ from $F$ such that

from all nodes there exists a path to some leaf of length at most $k$ resp.

after removal of the literals of $C$ from the refutation, for every resolution step at least one of the parent clauses has length at most $k$. 

Hierarchies

Basic relations

\[ \mathcal{UC}_0 \subset \mathcal{UC}_1 \subset \mathcal{UC}_2 \subset \ldots \]
\[ \mathcal{WC}_0 \subset \mathcal{WC}_1 \subset \mathcal{WC}_2 \subset \ldots \]
\[ \mathcal{UC}_0 = \mathcal{WC}_0 \]
\[ \mathcal{UC}_1 = \mathcal{WC}_1 \]
\[ \mathcal{UC}_k \subset \mathcal{WC}_k \text{ for } k \geq 2 \]
\[ \mathcal{UC}_{k+1} \not\subset \mathcal{WC}_k \text{ for } k \geq 0 \]
\[ \mathcal{WC}_3 \not\subset \mathcal{UC}_k \text{ for } k \geq 0. \]

Open Problem

For the last relation, can we use \( \mathcal{WC}_2 \)?
Decision complexity

$\mathcal{UC}_0 = \mathcal{WC}_0$ is decidable in polynomial time.

All $\mathcal{UC}_k$, $\mathcal{WC}_k$ for $k \geq 1$ are coNP-complete.
\( WC_0 = UC_0 \) is the class of clause-sets which contain all their prime implicates.

The class \( UC := UC_1 = WC_1 \) showed up in two different contexts:

1. **UC** was introduced in del Val [5] for the purpose of Knowledge Compilation (KC).

2. In [7] we showed \( UC = SLUR \), for the umbrella class \( SLUR \) for polytime SAT decision as introduced in Schlipf, Annexstein, Franco, and Swaminathan [12].

More generally we have \( UC_k = SLUR_k \) for \( k \geq 0 \).
Strong separation

In Gwynne and Kullmann [6] we show:

**Theorem**

For all $k \geq 0$ there are (sequences of) short clause-sets in $UC_{k+1}$, where all (sequences of) equivalent clause-sets in $WC_k$ are of exponential size.

**Conjecture**

This strong separation holds between classes $\mathcal{C}, \mathcal{D} \in \{UC_p, WC_q\}$ iff it is not trivially false, i.e., iff $\mathcal{C} \not\subseteq \mathcal{D}$. 
Allowing auxiliary variables

Consider $F, G \in \mathcal{CLS}$ with $\text{var}(F) \subseteq \text{var}(G)$.

**Definition**

*G represents F* if the satisfying assignments of $G$ projected to $\text{var}(F)$ are precisely the satisfying assignments of $F$.

**Conjecture**

*For all $k \geq 0$ there are (sequences of) short clause-sets in $\mathcal{UC}_{k+1}$, where all (sequences of) representing clause-sets in $\mathcal{WC}_k$ are of exponential size.*

*More generally, such a separation holds between classes $\mathcal{C}, \mathcal{D} \in \{\mathcal{UC}_p, \mathcal{WC}_q\}$ iff it is not trivially false.*
The “relative condition”

If $G$ represents $F$, then the **absolute condition** for $G$ is a requirement

- $G \in \mathcal{UC}_k$ or
- $G \in \mathcal{WC}_k$

for some suitable $k$.

So the requirements on prime implicates also concern prime implicates containing auxiliary variables (i.e., variables in $G$ but not in $F$).

Now the **relative condition** considers only prime implicates with variables from $F$.

We then speak of **relative hardness**.

This is, when using auxiliary variables, a weaker requirement.
Collapse under the relative condition

In [10] we show:

**Theorem**

Allowing representations with auxiliary variables, under the relative condition all classes $\mathcal{UC}_k$, $\mathcal{WC}_k$ collapse in polynomial time to $\mathcal{UC}_0$ or $\mathcal{UC}_1$.

**Conjecture**

There are (sequences of) clause-sets which have short representations of relative hardness 1, but for each $k$ have only (sequences of) superpolynomial / exponential size representations in $\mathcal{WC}_k$. 
Monotone circuits

Monotonisation of boolean functions

Consider a boolean function $f$.

We want partial assignments to $f$, handled by a boolean function $\hat{f}$.

- Every variable is doubled.
- So we can encode “not assigned”.

Now

$$\hat{f} = 0 \text{ iff the corresponding partial assignment makes } f \text{ unsatisfiable.}$$

Example: the monotonisation of the bijective PHP$_m^m$ function is the matching function (essentially).
Theorem

Consider a boolean function $f$ and a representation $F$ with relative hardness $1$. From $F$ we can compute in time $O(\ell(F) \cdot n(F)^2)$ a monotone circuit computing $\hat{f}$.

Corollary

Boolean functions $f_n$ have a CNF-representation $F_n$ with relative hardness $1$ and $\ell(F_n) = n^{O(1)}$ if and only if $\hat{f}_n$ can be computed by monotone circuits of size polynomial in $n$.

(The predecessor of these results is Bessiere, Katsirelos, Narodytska, and Walsh [3] (with $\hat{f}$ “hidden”, and a more complicated proof).)
No polysize good representations for XOR’s

Exploiting Babai, Gál, and Wigderson [2] (monotone span programs):

**Theorem**

_The size of representations of systems of XOR-constraints with bounded relative asymmetric width is super-polynomial in the number of constraints._
Theorem

From a clause-set $F$ and $V \subseteq \text{var}(F)$, such that the relative asymmetric width of $F$ w.r.t. $V$ is a constant $k$, we can compute in polynomial time a $G \in \mathcal{CLS}$ with $V \subseteq \text{var}(G)$ such that

- $G$ represents the same boolean function w.r.t. $V$ as $F$.
- $G$ has relative hardness $1$.
- More strongly, for every partial assignment $\varphi$ with $\text{var}(\varphi) \subseteq V$, running unit-clause propagation on $\varphi \ast G$ will find all forced assignments on variables from $V$.
- Moreover, for every $\varphi$ with $\text{var}(\varphi) = V$, such that $\varphi \ast G$ is satisfiable, running unit-clause propagation on $\varphi \ast G$ yields $\top$.

The terminology “strongly forcing” has been developed in collaboration with Donald Knuth (for his forthcoming fascicle on satisfiability).
Summary and outlook

I Hopefully a theory of “good SAT representations” will emerge.

II The translation of XOR-systems is a good first test-case: Despite the bad news “no poly-size good representation”, there seem to be a lot of opportunities for good representations (under various circumstances).

III The conjectures seem to require new techniques; inside monotone circuits there should be corresponding subclasses.
End

(references on the remaining slides).

For my papers see

http://cs.swan.ac.uk/~csoliver/papers.html.
Bibliography I

[1] Albert Atserias and Víctor Dalmau. A combinatorial characterization of resolution width. *Journal of Computer and System Sciences*, 74(3):323–334, May 2008. doi:10.1016/j.jcss.2007.06.025.

[2] László Babai, Anna Gál, and Avi Wigderson. Superpolynomial lower bounds for monotone span programs. *Combinatorica*, 19(3):301–319, March 1999. doi:10.1007/s004930050058.

[3] Christian Bessiere, George Katsirelos, Nina Narodytska, and Toby Walsh. Circuit complexity and decompositions of global constraints. In *Twenty-First International Joint Conference on Artificial Intelligence (IJCAI-09)*, pages 412–418, 2009.
Bibliography II

[4] Olaf Beyersdorff and Oliver Kullmann. Unified characterisations of resolution hardness measures. In Uwe Egly and Carsten Sinz, editors, Theory and Applications of Satisfiability Testing - SAT 2014, volume 8561 of Lecture Notes in Computer Science, pages 170–187. Springer, 2014. ISBN 978-3-319-09283-6. doi:10.1007/978-3-319-09284-3_13.

[5] Alvaro del Val. Tractable databases: How to make propositional unit resolution complete through compilation. In Proceedings of the 4th International Conference on Principles of Knowledge Representation and Reasoning (KR’94), pages 551–561, 1994.

[6] Matthew Gwynne and Oliver Kullmann. Trading inference effort versus size in CNF knowledge compilation. Technical Report arXiv:1310.5746v2 [cs.CC], arXiv, November 2013. URL http://arxiv.org/abs/1310.5746.
[7] Matthew Gwynne and Oliver Kullmann. Generalising unit-refutation completeness and SLUR via nested input resolution. *Journal of Automated Reasoning*, 52(1):31–65, January 2014. doi:10.1007/s10817-013-9275-8.

[8] Matthew Gwynne and Oliver Kullmann. On SAT representations of XOR constraints. In Adrian-Horia Dediu, Carlos Martín-Vide, José-Luis Sierra, and Bianca Truthe, editors, *LATA 2014: Language and Automata Theory and Applications, 8th International Conference*, volume 8370 of *Lecture Notes in Computer Science (LNCS)*, pages 409–420. Springer, 2014. doi:10.1007/978-3-319-04921-2_33.

[9] Matthew Gwynne and Oliver Kullmann. A framework for good SAT translations, with applications to CNF representations of XOR constraints. Technical Report arXiv:1406.7398v2 [cs.CC], arXiv, August 2014. URL http://arxiv.org/abs/1406.7398.
[10] Oliver Kullmann. Collapse of strong relative representation conditions to generalised arc consistency. In preparation, December 2014.

[11] Pavel Pudlák and Russell Impagliazzo. A lower bound for DLL algorithms for $k$-SAT (preliminary version). In SODA, pages 128–136. ACM/SIAM, 2000.

[12] John S. Schlipf, Fred S. Annexstein, John V. Franco, and R.P. Swaminathan. On finding solutions for extended Horn formulas. Information Processing Letters, 54(3):133–137, May 1995. doi:10.1016/0020-0190(95)00019-9.