Enhanced Unsatisfiable Cores for QBF: Weakening Universal to Existential Quantifiers

Viktor Schuppan

Viktor.Schuppan@gmx.de
http://schuppan.de/viktor/

ICTAI 2018, Volos, Greece, 5–7 November 2018

Intro — Some Problems with Natural Formulations as QBF

Artificial Intelligence

- Two-player games
- Variants of planning
- Many problems in knowledge representation

Formal Methods

- Verification: black box design, termination check
- Synthesis

Prototypical PSPACE-complete problem.

Introduction — Unsatisfiable Cores

Idea

- Part of an unsatisfiable formula that is by itself unsatisfiable.
- Typically obtained by syntactic weakening.

Some Applications

- Causes and explanations of unsatisfiability.
  (Extends to (un)wanted implications.)
- Via duality: diagnoses and repairs.
- ... and many more ...

Fundamental concept in applied logic.

Introduction — Overview

Quantified Boolean Formulas in Prenex Conjunctive Normal Form

$$\underbrace{Q_1 p_1 \ldots Q_n p_n}_{\text{prefix}} \; . \; \underbrace{(l_{1,1} \vee \ldots \vee l_{1,n_1}) \wedge \ldots \wedge (l_{m,1} \vee \ldots \vee l_{m,n_m})}_{\text{matrix: (propositional) CNF}}$$

$Q_i \in \{\exists, \forall\}$, p_i Boolean variables, $l_{i,i'}$ literals over $p_1, \ldots, p_n$.

Existing notion of unsatisfiable cores: remove clauses from matrix

$\forall p \, . \, (p) \wedge (\neg p) \quad \rightsquigarrow \quad \forall p \, . \, (p) \wedge (\neg p), \quad \forall p \, . \, (p), \quad \forall p \, . \, (\neg p).$

This paper: additionally weaken $\forall$ to $\exists$

$\forall p \, . \, (p) \wedge (\neg p) \quad \rightsquigarrow \quad \ldots, \quad \exists p.(p) \wedge (\neg p).$

$\Rightarrow$ More causes/explanations of unsatisfiability. (Transfers to repairs.)

UCs for QBF in PCNF — Definitions

Let $\Pi.C$ be a QBF in PCNF.

Definition (C-,Q-, and QC-Core)

C-Core	Remove 0 or more clauses from the matrix C [YM05].
Q-Core	Weaken 0 or more $\forall$ to $\exists$ in the prefix Π.
QC-Core	Combined c-core and q-core.

Definition (Unsatisfiable Core)

Unsatisfiable C-/Q-/QC-Core A c-/q-/qc-core that is unsatisfiable.

Definition (Minimal Unsatisfiability)

C-Minimally Unsatisfiable Unsatisfiable and no clause can be removed from the matrix C without making the result satisfiable.

Q-Minimally Unsatisfiable Unsatisfiable and no $\forall$ can be weakened to $\exists$ in the prefix Π without making the result satisfiable.

UCs for QBF in PCNF — Example

Consider $\Pi.C = \forall p.(p) \wedge (\neg p)$.

C-Cores:	$\Pi.C$,	$\forall p.(p)$,	$\forall p.(\neg p)$,	$\forall p.\top$
Q-Cores:	$\Pi.C$,			
	$\exists p.(p) \wedge (\neg p)$			
QC-Cores:	$\Pi.C$,	$\forall p.(p)$,	$\forall p.(\neg p)$,	$\forall p.\top$,
	$\exists p.(p) \wedge (\neg p)$,	$\exists p.(p)$,	$\exists p.(\neg p)$,	$\exists p.\top$

Unsatisfiable cores are red, satisfiable ones are green.

A2AECC — Q- and QC-Cores as C-Cores

Let $\Pi.C$ be a QBF in PCNF.

Definition (A2AECC)

Let $\Pi' := \Pi$, $C' := C$;
For every $\forall p_i$ in Π:
    Let p_i' be fresh;
    Replace $\forall p_i$ with $\forall p_i' \exists p_i$ in Π';
    Replace C' with $(p_i \rightarrow p_i') \wedge (p_i \rightarrow p_i') \wedge C'$;
Return $\Pi'.C'$;

Theorem (Correctness of A2AECC)

Let $\tilde{P}$ be a subset of the universally quantified variables in Π and let $\tilde{C}$ be the corresponding clauses added by A2AECC. Then

$\Pi.C$ with variables in $\tilde{P}$ weakened from $\forall$ to $\exists$ is satisfiable

iff

$A2AECC(\Pi.C)$ with clauses in $\tilde{C}$ removed is satisfiable.

Use methods and tools for c-cores to obtain q- and qc-cores.

A2AECC — Example

Consider $\Pi.C = \forall p.(p) \wedge (\neg p)$.

$A2AECC(\Pi.C) = \forall p' \exists p.(p \rightarrow p') \wedge (p' \rightarrow p) \wedge (p) \wedge (\neg p)$.

Treat $(p \rightarrow p') \wedge (p' \rightarrow p)$ as clause group [Nad10; LS08].

QC-Core of $\Pi.C$	C-Core of $A2AECC(\Pi.C)$
$\forall p.(p) \wedge (\neg p)$	$\forall p' \exists p.(p \rightarrow p') \wedge (p' \rightarrow p) \wedge (p) \wedge (\neg p)$
$\forall p.(p)$	$\forall p' \exists p.(p \rightarrow p') \wedge (p' \rightarrow p) \wedge (p)$
$\forall p. \quad (\neg p)$	$\forall p' \exists p.(p \rightarrow p') \wedge (p' \rightarrow p) \wedge \quad (\neg p)$
$\forall p.\top$	$\forall p' \exists p.(p \rightarrow p') \wedge (p' \rightarrow p)$
$\exists p.(p) \wedge (\neg p)$	$\forall p' \exists p. \quad (p) \wedge (\neg p)$
$\exists p.(p)$	$\forall p' \exists p. \quad (p)$
$\exists p. \quad (\neg p)$	$\forall p' \exists p. \quad (\neg p)$
$\exists p.\top$	$\forall p' \exists p.\top$

Unsatisfiable cores are red, satisfiable ones are green.

Experimental Evaluation — Implementation and Examples

Implementation

- Extends DepQBF 6.03 [LE17], which provides some basic infrastructure to extract c-cores, with A2AECC.
- Can be used as preprocessor or unsatisfiable c-/q-/qc-core extractor.
- Optionally performs deletion-based minimization [Mar12] with clause set refinement [BLM12].

Examples

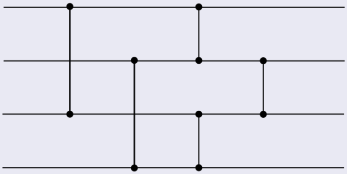
- 5342 instances from QBFLIB [GNPT]
- Interested in potential to weaken $\forall$ to $\exists$ $\Rightarrow$ no preprocessor

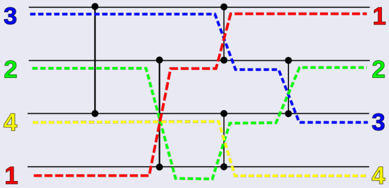
http://schuppan.de/viktor/ictai18/

Experimental Evaluation — Case Studies

Conformant Planning: Sorting Networks [Rin07]

- Does there exist a sorting network of depth 3 for input sequences of length 6?
  - $\exists$ plan $\forall$ (input sequence) ...
  - Unsatisfiable core: $\forall$ over the first number weakened to $\exists$.
  - No such sorting network independent of value of the first number.
  - $\Rightarrow$ no such sorting network of depth 3 for input sequences of length 5.



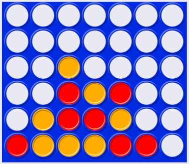


“The entire operation of a simple sorting network” by Oskar Sigvardsson is licensed under CC BY 3.0.

Experimental Evaluation — Case Studies

Two-Player Games: Generalized Connect-4 [GR03]

- Can player 1 enforce a draw on a 2-by-2 board with winning lines of length 2?
  - ∃ (move 1 of player 1) ∀ (move 1 of player 2) ...
  - Unsatisfiable core with no ∀ left.
  - Not possible, even if player 1 had full control over the moves of player 2.
- As before but on larger boards and with longer winning lines?
  - ∃ (move 1 of player 1) ∀ (move 1 of player 2) ...
  - Unsatisfiable core with a single ∀ left.
  - Game is modeled [GR03] such that player 2 can play an illegal first move, thus forcing a win of player 1.
  - Is this model of the game as intended?



Experimental Evaluation — Overhead of UC Extraction

mode	solved instances
no unsatisfiable core	1911
unsatisfiable c-core	1830
c-minimally unsatisfiable c-core	1682
unsatisfiable q-core	1649
q-minimally unsatisfiable q-core	1139
unsatisfiable qc-core	1551
q-,c-minimally unsatisfiable qc-core	927

Related Work

- [RSMB14]: most closely related
  - introduces soft variables: may be placed at different positions in prefix, subject to preference function;
  - maximises preference function while maintaining satisfiability;
  - uses generalized version of A2AECC to reduce to weighted partial MaxSAT (we discovered our transformation independently);
  - differences:
    - makes no connection to unsatisfiable cores,
    - still satisfiable vs. still unsatisfiable,
    - always maximum vs. optionally minimal,
    - does not optimize the matrix.
- [YM05; KZ06; IJM13; LE15]: compute c-cores.
- [BLB10]: manipulates quantifiers when minimizing failure-inducing input.
- [LB11; LES16]: refer to weakening $\forall$ to $\exists$ as "quantifier abstraction" and "existential abstraction".

The End

Summary

- Propose to weaken $\forall$ to $\exists$ in QBF unsatisfiable cores.
- Obtain additional causes of unsatisfiability.
- Implementation: enhanced UCs obtained in many instances.
- Case studies: enhanced unsatisfiable cores provide useful information.

Potential Future Work

- Understand impact of A2AECC transformation on different solvers.
- Avoid use of A2AECC transformation.
- Other logics with quantification.

Acknowledgment

I thank the authors of [RSMB14], especially Sven Reimer, for discussion of their work and for providing me with `quantom` [RPSB12].

References

[BLB10] R. Brummayer, F. Lonsing, and A. Biere. "Automated Testing and Debugging of SAT and QBF Solvers". In: *SAT*. Vol. 6175. LNCS. 2010.

[BLM12] A. Belov, I. Lynce, and J. Marques-Silva. "Towards efficient MUS extraction". In: *AI Commun.* 25.2 (2012).

[GNPT] E. Giunchiglia, M. Narizzano, L. Pulina, and A. Tacchella. *Quantified Boolean Formulas satisfiability library (QBFLIB)*. http://www.qbflib.org/.

[GR03] I. P. Gent and A. G. D. Rowley. "Encoding Connect-4 Using Quantified Boolean Formulae". In: *ModRef*. 2003.

[IJM13] A. Ignatiev, M. Janota, and J. Marques-Silva. "Quantified Maximum Satisfiability: A Core-Guided Approach". In: *SAT*. Vol. 7962. LNCS. 2013.

[KZ06] H. Kleine Büning and X. Zhao. "Minimal False Quantified Boolean Formulas". In: *SAT*. Vol. 4121. LNCS. 2006.

[LB11] F. Lonsing and A. Biere. "Failed Literal Detection for QBF". In: *SAT*. Vol. 6695. LNCS. 2011.

[LE15] F. Lonsing and U. Egly. "Incrementally Computing Minimal Unsatisfiable Cores of QBFs via a Clause Group Solver API". In: *SAT*. Vol. 9340. LNCS. 2015.

[LE17] F. Lonsing and U. Egly. "DepQBF 6.0: A Search-Based QBF Solver Beyond Traditional QCDCL". In: *CADE*. Vol. 10395. LNCS. 2017.

[LES16] F. Lonsing, U. Egly, and M. Seidl. "Q-Resolution with Generalized Axioms". In: *SAT*. Vol. 9710. LNCS. 2016.

[LS08] M. H. Liffiton and K. A. Sakallah. "Algorithms for Computing Minimal Unsatisfiable Subsets of Constraints". In: *J. Autom. Reasoning* 40.1 (2008).

[Mar12] J. Marques-Silva. "Computing Minimally Unsatisfiable Subformulas: State of the Art and Future Directions". In: *Multiple-Valued Logic and Soft Computing* 19.1-3 (2012).

[Nad10] A. Nadel. "Boosting minimal unsatisfiable core extraction". In: *FMCAD*. IEEE, 2010.

[Rin07] J. Rintanen. "Asymptotically Optimal Encodings of Conformant Planning in QBF". In: *AAAI*. AAAI Press, 2007.

[RPSB12] S. Reimer, F. Pigorsch, C. Scholl, and B. Becker. "Enhanced Integration of QBF Solving Techniques". In: *MBMV*. Verlag Dr. Kovac, 2012.

[RSMB14] S. Reimer, M. Sauer, P. Marin, and B. Becker. "QBF with Soft Variables". In: *ECEASST* 70 (2014).

[YM05] Y. Yu and S. Malik. "Validating the result of a Quantified Boolean Formula (QBF) solver: theory and practice". In: *ASP-DAC*. ACM Press, 2005.