Solution Validation and Extraction for QBF Preprocessing

Marijn J.H. Heule

THE UNIVERSITY OF TEXAS AT AUSTIN

Joint work with
Martina Seidl and Armin Biere

JOHANNES KEPLER UNIVERSITY LINZ | JKU

Dagstuhl September 20, 2016

Introduction and Motivation

Clausal Proofs for QBF Preprocessing

From Clausal Proofs to Skolem Functions

From Clausal Proofs to Skolem Functions

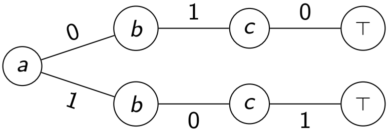
Future Directions and Conclusions

Introduction to QBF

A quantified Boolean formula (QBF) is a propositional formula where variables are existentially ($\exists$) or universally ($\forall$) quantified.

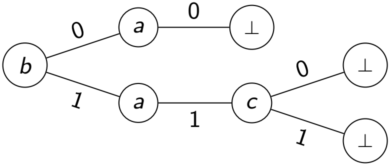
Consider the formula $\forall a \, \exists b, c. (a \vee b) \wedge (\bar{a} \vee c) \wedge (\bar{b} \vee \bar{c})$

A model is:



Consider the formula $\exists b \, \forall a \, \exists c. (a \vee b) \wedge (\bar{a} \vee c) \wedge (\bar{b} \vee \bar{c})$

A counter-model is:



Motivation for our QBF Proof System

Lots of "discrepancies" and unique results in QBF solvers:

- i.e., results that disagree with the majority of solvers.

To gain confidence in QBF results they need to be validated:

- existing methods cannot validate some QBF preprocessing.

QBF preprocessing is crucial for fast performance:

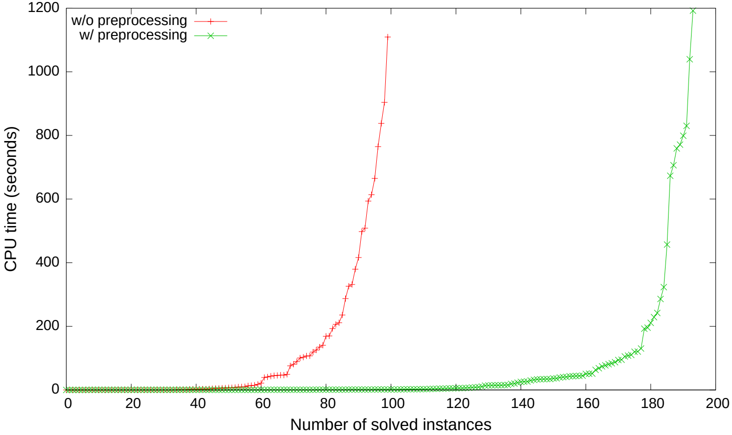
- most state-of-the-art solvers use the preprocessor bloqqer;
- current methods can produce exponentially large proofs or require exponential checking time in worst case;
- some techniques cannot be checked with these methods.

Clausal Proofs for QBF Preprocessing

QBF Preprocessing

Preprocessing is crucial to solve most QBF instances efficiently.

Results of DepQBF w/ and w/o bloqqer on QBF Eval 2012



QBF Preprocessing

Preprocessing is crucial to solve most QBF instances efficiently.

There exists lots of techniques. The most important ones are:

- tautology elimination, subsumption, universal reduction, existential pure literal elimination, strengthening, blocked clause elimination, unit literal elimination, universal pure literal elimination, covered literal addition, variable elimination, and universal expansion.

Existing methods and proof formats have shortcomings:

- some techniques require exponentially-sized proofs; and
- for some other techniques, it is not even known whether one can construct such a proof.

Challenges for Quantified Boolean Formulas (QBF)

Preprocessing is crucial to solve most QBF instances efficiently.

Proofs are useful for applications and to validate solver output.

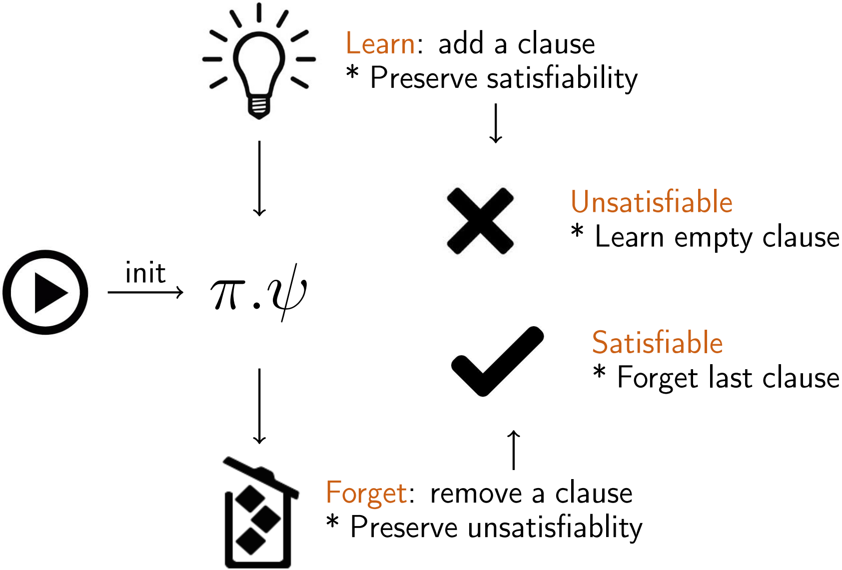
Main challenges regarding QBF and preprocessing [Janota'13]:

1. produce proofs that can be validated in polynomial time;
2. develop methods to validate all QBF preprocessing; and
3. narrow the performance gap between solving with and without proof generation.

In our IJCAR'14 paper [1], we meet all three challenges!

[1] Marijn J. H. Heule, Matina Seidl and Armin Biere: A Unified Proof System for QBF Preprocessing. IJCAR 2014, LNCS 8562, pp 91-106 (2014)

Clausal Proof System



QRAT: Quantified Resolution Asymmetric Tautologies

Clause C has AT (Asymmetric Tautology) w.r.t. $\psi \setminus \{C\}$ iff unit propagation derives a conflict in $(\psi \setminus \{C\}) \wedge \neg C$.

- E.g. $(a \vee b)$ has AT w.r.t. $(a \vee c) \wedge (\bar{c} \vee \bar{d}) \wedge (b \vee d)$
- Tautologies have AT

Clause C *has* QRAT (Quantified Resolution Asymmetric Tautology) w.r.t. $\psi \setminus \{C\}$ under π iff

- there exists a literal $l \in C$ such that
  for each clause $D \in \psi$ with $\bar{l} \in D$
  clause $\{k \mid k \in D, k <_\pi \bar{l}\} \cup C$ has AT w.r.t. $\psi \setminus C$.
- E.g. (a) has QRAT w.r.t.
  $\forall b, c \exists a.(a \vee b) \wedge (\bar{a} \vee c) \wedge (\bar{b} \vee c)$
- Clauses with AT w.r.t. ψ have QRAT w.r.t. ψ

Main Theoretical Result

We defined one Forget, one Learn, and two Strengthen rules:

- The rules are based on a redundancy property called QRAT
- The property QRAT can be computed in polynomial time

We showed that all QBF preprocessing techniques can be translated into a sequence of these Learn and Forget rules

- Our proof system can be used to validate all techniques
- The validation costs is similar to solving costs

Example

$\forall x_1..x_n \exists y_1..y_n.(x_1 \vee \bar{y}_1) \wedge (\bar{x}_1 \vee y_1)..(x_n \vee \bar{y}_n) \wedge (\bar{x}_n \vee y_n)$

- Our Forget rule can eliminate all clauses (linear time)
- A model for the formula is exponential in n

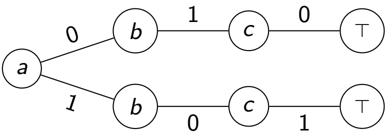
From Clausal Proofs to Skolem Functions

Introduction to Skolem functions for QBF

A **Skolem function** $f_x(U_x)$ for a QBF formula $\pi.\psi$ defines the truth value of an **existential variable** x based on the set U_x of universal variables that occur earlier in the prefix than x

Consider the formula $\forall a\, \exists b, c.(a \vee b) \wedge (\bar{a} \vee c) \wedge (\bar{b} \vee \bar{c})$

A **model** is:



The **set of Skolem functions** F (defining all existentials) is

$$F = \{f_b(a) = \bar{a},\ f_c(a) = a\}$$

The set of Skolem functions can be much smaller than a model

Redundancy Concepts in the QRAT Proof System

Informal definitions of the redundancy concepts in the QRAT proof system. They can be computed in polynomial time.

Definition (Asymmetric Tautologies (AT))

An asymmetric tautology is a clause that becomes a tautology after adding "hidden literals". ATs are logically implied by a formula.

Definition (Quantified Resolution AT (QRAT))

A quantified resolution AT is a clause that contains a literal for which all "outer resolvents" are ATs.

Definition (Extended Universal Reduction (EUR))

A universal literal is redundant if assigning it to false cannot influence the value of universal literals.

Rules of the QRAT Proof System

	Rule	Preconditions	Postconditions
(N1)	$\dfrac{\pi.\psi}{\pi.\psi \backslash \{C\}}$	C is an asymmetric tautology	
(N2)	$\dfrac{\pi.\psi}{\pi'.\psi \cup \{C\}}$	C is an asymmetric tautology	$\pi' = \pi\exists X$ with $X = \{x \mid x \in \text{vars}(C), x \notin \text{vars}(\pi)\}$
(E1)	$\dfrac{\pi.\psi}{\pi.\psi \backslash \{C\}}$	$C \in \psi$, $Q(\pi, l) = \exists$ C has QRAT on l w.r.t. ψ	
(E2)	$\dfrac{\pi.\psi}{\pi'.\psi \cup \{C\}}$	$C \notin \psi$, $Q(\pi, l) = \exists$ C has QRAT on l w.r.t. ψ	$\pi' = \pi\exists X$ with $X = \{x \mid x \in \text{vars}(C), x \notin \text{vars}(\pi)\}$
(U1)	$\dfrac{\pi.\psi \cup \{C\}}{\pi.\psi \cup \{C \backslash \{l\}\}}$	$l \in C$, $Q(\pi, l) = \forall$, $\bar{l} \notin C$, C has QRAT on l w.r.t. ψ	
(U2)	$\dfrac{\pi.\psi \cup \{C\}}{\pi.\psi \cup \{C \backslash \{l\}\}}$	$l \in C$, $Q(\pi, l) = \forall$, $\bar{l} \notin C$, C has EUR on l w.r.t. ψ	

Rules of the QRAT Proof System

	Rule	Preconditions	Postconditions
(N1)	$\dfrac{\pi.\psi}{\pi.\psi\setminus\{C\}}$	C i tau **Preserves Logical Equivalence**	
(N2)	$\dfrac{\pi.\psi}{\pi'.\psi\cup\{C\}}$	C i tau **Preserves Logical Equivalence**	$(\pi)\}$
(E1)	$\dfrac{\pi.\psi}{\pi.\psi\setminus\{C\}}$	$C\in\psi$, Q(C has QR **Weakens the Formula**	
(E2)	$\dfrac{\pi.\psi}{\pi'.\psi\cup\{C\}}$	$C\notin\psi$, C has (**Strengthens the Formula**	vars$(\pi)\}$
(U1)	$\dfrac{\pi.\psi\cup\{C\}}{\pi.\psi\cup\{C\setminus\{l\}\}}$	$l\in C$, Q C has (**Strengthens the Formula**	
(U2)	$\dfrac{\pi.\psi\cup\{C\}}{\pi.\psi\cup\{C\setminus\{l\}\}}$	$l\in C$, Q C has E **Strengthens the Formula**	

Pseudo-Code of Skolem Function Computation

ComputeSkolem (prefix π, QRAT proof P)

let ψ be an empty formula
foreach existential variable e **do** $f_e(U) := *$ // initialize F
while (P is not empty) **do**
  $\langle \mathit{rule}\ R, \mathit{clause}\ C, \mathit{literal}\ l \rangle := P.\mathrm{pop}()$
  if ($R = \mathrm{E1}$) **then**
    let e be $\mathrm{var}(l)$
    $f_e(U) := \mathrm{IfThenElse}(F(\mathcal{OF}(\pi, \psi, l)), \mathrm{polarity}(l), f_e(U))$
  if ($R = \mathrm{E1}$ **or** $R = \mathrm{N1}$) **then** // Forget rules
    $\psi := \psi \cup \{C\}$
  if ($R = \mathrm{E2}$ **or** $R = \mathrm{N2}$) **then** // Learn rules
    $\psi := \psi \setminus \{C\}$

Checks to Validate Skolem Functions

Two tests are required to validate Skolem functions:

1. Can we falsify a clause in formula ψ while satisfying the Skolem functions $F(U)$?

$$\text{solve}(\bar{\psi} \land F(U)) \; = \text{UNSAT?}$$

2. Check that all Skolem functions depend only on universal variables that occur earlier in the prefix.

   Problem: our method could create a Skolem function

$$f_x(U_x) := f_y(U_y) \text{ with } \pi(x) < \pi(y)$$

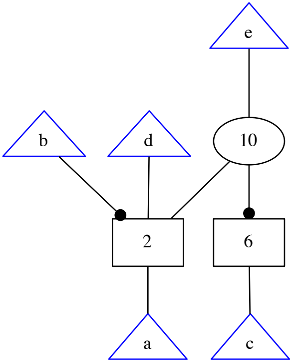
   Solution: convert Skolem functions to And-Inverter-Graphs (AIGs) and check for reachability.

Check Reachability in AIGs

Consider the formula $\pi.\psi$:

$$
\begin{aligned}
\forall a \exists b \forall c \exists d, e.\quad & \\
(a \vee b) &\quad \wedge \\
(\bar{a} \vee \bar{b} \vee d) &\quad \wedge \\
(a \vee c \vee \bar{d}) &\quad \wedge \\
(a \vee \bar{b} \vee \bar{e}) &\quad \wedge \\
(\bar{a} \vee c \vee e) &\quad \wedge \\
(\bar{c} \vee \bar{e}) &
\end{aligned}
$$

Skolem functions for $\pi.\psi$:

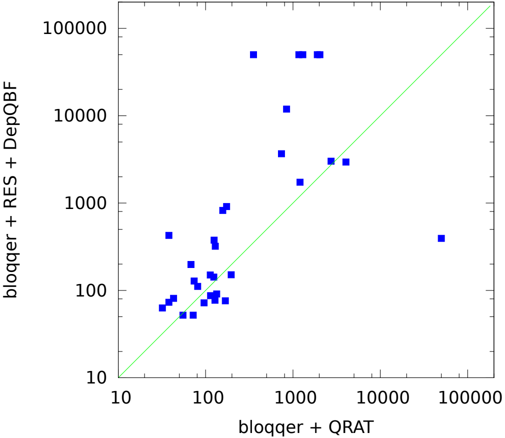


Our algorithm could have produced $f_b(a) := f_d(a, c)$, but that is not problematic because $f_d(a, c)$ does not depend on c.

How to simplify the circuit and preserve the dependencies?

Results Summary

Our approach was able to compute more Skolem functions for formulas that are solvable by preprocessing techniques only as no techniques had to be turned off.



Above the diagonal: Skolem functions from QRAT proofs are smaller

Future Directions and Conclusions

Future Directions

Novel techniques arise from the proof systems

- SAT: Elimination and addition of RAT clauses
- SAT: Partial variable elimination
- QBF: Elimination of universal RAT literals
- Many other options

Efficient expression of all techniques

- Main focus: all QBF solving techniques (i.e., not only preprocessing)
- Gaussian Elimination
- Symmetry breaking
- Cardinality / pseudo-Boolean reasoning

Conclusions

Our Abstract Proof System for SAT Inprocessing

- Captures generally used inprocessing and CDCL techniques
- Check individual techniques for correctness via the inprocessing rules
- Yields a generic and simple model reconstruction algorithm
- A basis for developing novel inprocessing techniques

Conclusions

Our Abstract Proof System for SAT Inprocessing

- Captures generally used inprocessing and CDCL techniques
- Check individual techniques for correctness via the inprocessing rules
- Yields a generic and simple model reconstruction algorithm
- A basis for developing novel inprocessing techniques

Our Proof System for QBF Preprocessing

- Polynomially-verifiable certificates for true and false QBFs;
- Overhead of emitting QRAT proofs is very low; and
- All preprocessing techniques used in state-of-the-art QBF tools are covered by QRAT, including universal expansion.
- A basis for developing novel QBF preprocessing techniques

Conclusions

Our Abstract Proof System for SAT Inprocessing

- Captures generally used inprocessing and CDCL techniques
- Check individual techniques for correctness via the inprocessing rules
- Yields a generic and simple model reconstruction algorithm
- A basis for developing novel inprocessing techniques

Our Proof System for QBF Preprocessing

- Polynomially-verifiable certificates for true and false QBFs;
- Overhead of emitting QRAT proofs is very low; and
- All preprocessing techniques used in state-of-the-art QBF tools are covered by QRAT, including universal expansion.
- A basis for developing novel QBF preprocessing techniques

Thanks!

QBF: Universal Expansion Example

Universal expansion eliminates an innermost universal variable x by duplicating the formula inner to x.

$$\frac{\pi \forall x \exists Y . \psi, C_1 \lor \bar{x}, \ldots, C_i \lor \bar{x}, D_1 \lor x, \ldots, D_j \lor x, E_1, \ldots, E_k}{\pi \exists Y Y' . \psi, C_1, \ldots, C_i, D_1', \ldots, D_j', E_1, \ldots, E_k, E_1', \ldots, E_k'}$$

QBF: Universal Expansion Example

Universal expansion eliminates an innermost universal variable x by duplicating the formula inner to x.

$$\frac{\pi\forall x\exists Y.\psi, C_1 \vee \bar{x}, \ldots, C_i \vee \bar{x}, D_1 \vee x, \ldots, D_j \vee x, E_1, \ldots, E_k}{\pi\exists YY'.\psi, C_1, \ldots, C_i, D_1', \ldots, D_j', E_1, \ldots, E_k, E_1', \ldots, E_k'}$$

The true formula $\forall a\,\exists b, c.(\bar{a} \vee c) \wedge (a \vee b) \wedge (\bar{b} \vee \bar{c})$ can be expanded to:

$$\exists b, c, b', c'.(c) \wedge (b') \wedge (\bar{b} \vee \bar{c}) \wedge (\bar{b}' \vee \bar{c}')$$

QBF: Universal Expansion Example

Universal expansion eliminates an innermost universal variable x by duplicating the formula inner to x.

$$\frac{\pi\forall x\exists Y.\psi, C_1 \vee \bar{x}, \ldots, C_i \vee \bar{x}, D_1 \vee x, \ldots, D_j \vee x, E_1, \ldots, E_k}{\pi\exists YY'.\psi, C_1, \ldots, C_i, D'_1, \ldots, D'_j, E_1, \ldots, E_k, E'_1, \ldots, E'_k}$$

The true formula $\forall a\exists b, c.(\bar{a} \vee c) \wedge (a \vee b) \wedge (\bar{b} \vee \bar{c})$ can be expanded to:

$$\exists b, c, b', c'.(c) \wedge (b') \wedge (\bar{b} \vee \bar{c}) \wedge (\bar{b}' \vee \bar{c}')$$

The false formula $\exists b\forall a\exists c.(\bar{a} \vee c) \wedge (a \vee b) \wedge (\bar{b} \vee \bar{c})$ can be expanded to:

$$\exists b, c, c'.(c) \wedge (b) \wedge (\bar{b} \vee \bar{c}) \wedge (\bar{b} \vee \bar{c}')$$

QBF: Universal Expansion Example with QRAT

$$\frac{\pi \forall x \exists Y.\psi, C_1 \vee \bar{x}, \dots, C_i \vee \bar{x}, D_1 \vee x, \dots, D_j \vee x, E_1, \dots, E_k}{\pi \exists Y Y'.\psi, C_1, \dots, C_i, D'_1, \dots, D'_j, E_1, \dots, E_k, E'_1, \dots, E'_k}$$

$$\frac{\forall a \exists b, c.(\bar{a} \vee c) \wedge (a \vee b) \wedge (\bar{b} \vee \bar{c})}{\exists b, c, b', c'.(c) \wedge (b') \wedge (\bar{b} \vee \bar{c}) \wedge (\bar{b}' \vee \bar{c}')}$$

QBF: Universal Expansion Example with QRAT

$$\frac{\pi\forall x\exists Y.\psi,\ C_1 \vee \bar{x}, \ldots, C_i \vee \bar{x},\ D_1 \vee x, \ldots, D_j \vee x,\ E_1, \ldots, E_k}{\pi\exists YY'.\psi,\ C_1, \ldots, C_i,\ D_1', \ldots, D_j',\ E_1, \ldots, E_k, E_1', \ldots, E_k'}$$

$$\frac{\forall a\,\exists b, c.(\bar{a} \vee c) \wedge (a \vee b) \wedge (\bar{b} \vee \bar{c})}{\exists b, c, b', c'.(c) \wedge (b') \wedge (\bar{b} \vee \bar{c}) \wedge (\bar{b}' \vee \bar{c}')}$$

Phase 1: Learn

1. $(a \vee b \vee \bar{b}')$
2. $(a \vee \bar{b} \vee b')$
3. $(a \vee c \vee \bar{c}')$
4. $(a \vee \bar{c} \vee c')$
5. $(\bar{a} \vee \bar{b} \vee \bar{c})$
6. $(a \vee b')$
7. $(a \vee \bar{b}' \vee \bar{c}')$

Phase 2: Forget

1. $(a \vee b)$
2. $(\bar{b} \vee \bar{c})$
3. $(a \vee b \vee \bar{b}')$
4. $(a \vee \bar{b} \vee b')$
5. $(a \vee c \vee \bar{c}')$
6. $(a \vee \bar{c} \vee c')$

Phase 3: Strengthen

1. $(\bar{a} \vee c)$
2. $(a \vee b')$
3. $(\bar{a} \vee \bar{b} \vee \bar{c})$
4. $(a \vee \bar{b}' \vee \bar{c}')$