QRAT$^+$: Generalizing QRAT by a More Powerful QBF Redundancy Property

Florian Lonsing

Research Division of Knowledge Based Systems
Institute of Logic and Computation, TU Wien, Vienna, Austria
http://www.kr.tuwien.ac.at/staff/lonsing/

Computability in Europe (CiE) 2018, July 30 - August 3, Kiel, Germany

This work is supported by the Austrian Science Fund (FWF) under grant S11409-N23.

Introduction (1)

Quantified Boolean Formulas (QBF):

- Existential ($\exists$) / universal ($\forall$) quantification of propositional variables.
- Checking QBF satisfiability: PSPACE-complete.
- QBF encodings: potentially more succinct than propositional logic.

Progress in QBF Reasoning:

- Theory: proof systems (foundations of solver implementations).
- Practice: solving, preprocessing.

Example

Syntax:

- QBF $\psi := \Pi.\phi$ in *prenex conjunctive normal form (PCNF)*.
- $\psi = \underbrace{\forall u \exists x.}_{\text{quantifier prefix}} \underbrace{(\bar{u} \vee x) \wedge (u \vee \bar{x})}_{\text{propositional CNF}}.$

Introduction (2)

Example

Syntax:

- QBF $\psi := \hat{Q}.\phi$ in *prenex conjunctive normal form (PCNF)*.
- $\psi = \underbrace{\forall u \exists x.}_{\text{quantifier prefix}} \underbrace{(\bar{u} \vee x) \wedge (u \vee \bar{x})}_{\text{propositional CNF}}.$

Semantics (recursive):

- Assign variables in prefix ordering, recurse on simplified formula $\psi[A]$ under current assignment A.
- Base cases: $\bot$ is unsatisfiable, $\top$ is satisfiable.
- $\forall u.\psi$ is satisfiable iff $\psi[u/\bot]$ and $\psi[u/\top]$ are satisfiable.
- $\exists x.\psi$ is satisfiable iff $\psi[x/\bot]$ or $\psi[x/\top]$ is satisfiable.

PCNF ψ above is satisfiable:

- $\psi[u/\bot] = \exists x.(\bar{x})$ is satisfiable by setting x to $\bot$.
- $\psi[u/\top] = \exists x.(x)$ is satisfiable by setting x to $\top$.

Introduction (2)

Example

Syntax:

- QBF $\psi := \hat{Q}.\phi$ in *prenex conjunctive normal form (PCNF)*.
- $\psi = \underbrace{\forall u \exists x.}_{\text{quantifier prefix}} \underbrace{(\bar{u} \vee x) \wedge (u \vee \bar{x})}_{\text{propositional CNF}}.$

Semantics (recursive):

- Assign variables in prefix ordering, recurse on simplified formula $\psi[A]$ under current assignment A.
- Base cases: $\bot$ is unsatisfiable, $\top$ is satisfiable.
- $\forall u.\psi$ is satisfiable iff $\psi[u/\bot]$ and $\psi[u/\top]$ are satisfiable.
- $\exists x.\psi$ is satisfiable iff $\psi[x/\bot]$ or $\psi[x/\top]$ is satisfiable.

PCNF ψ above is satisfiable:

- $\psi[u/\bot] = \exists x.(\bar{x})$ is satisfiable by setting x to $\bot$.
- $\psi[u/\top] = \exists x.(x)$ is satisfiable by setting x to $\top$.

Introduction (3): QBF Solving

Resolution-Based Solving: [GNT06, Let02, ZM02a, ZM02b]

- Backtracking algorithm, related to DPLL [CGS98, DLL62].
- Generalization of conflict-driven clause learning (CDCL) to QBF.
- (Variants of) Q-resolution: proof systems [BWJ14, KBKF95, VG12].

Expansion-Based Solving: [AB02, Bie04]

- Elimination of variables, eventually reducing formula to $\top/\bot$.
- Exponential space (worst case).
- Modern approach: CEGAR-based lazy expansion [JKMSC16, RT15].
- $\forall$Exp+Res: proof system [BCJ15, JM13].

$\Rightarrow$ Proof complexity results: orthogonality of Q-resolution and expansion.

Introduction (3): QBF Solving

Resolution-Based Solving: [GNT06, Let02, ZM02a, ZM02b]

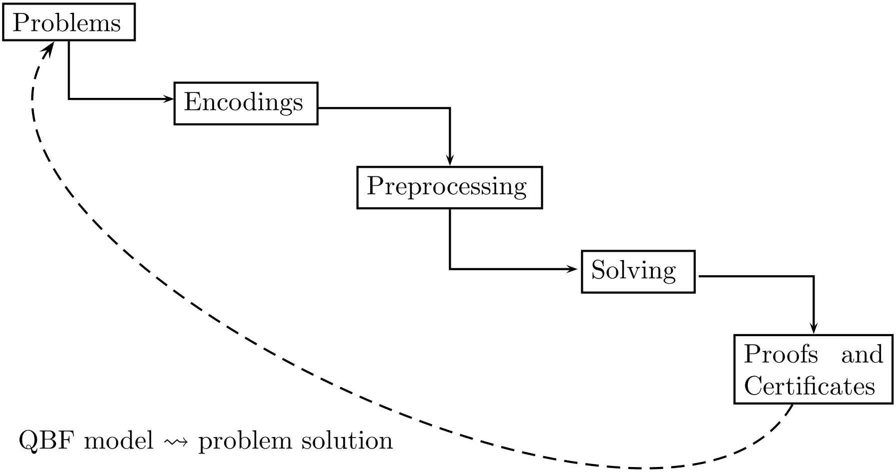
- Backtracking algorithm, related to DPLL [CGS98, DLL62].
- Generalization of conflict-driven clause learning (CDCL) to QBF.
- (Variants of) Q-resolution: proof systems [BWJ14, KBKF95, VG12].

Expansion-Based Solving: [AB02, Bie04]

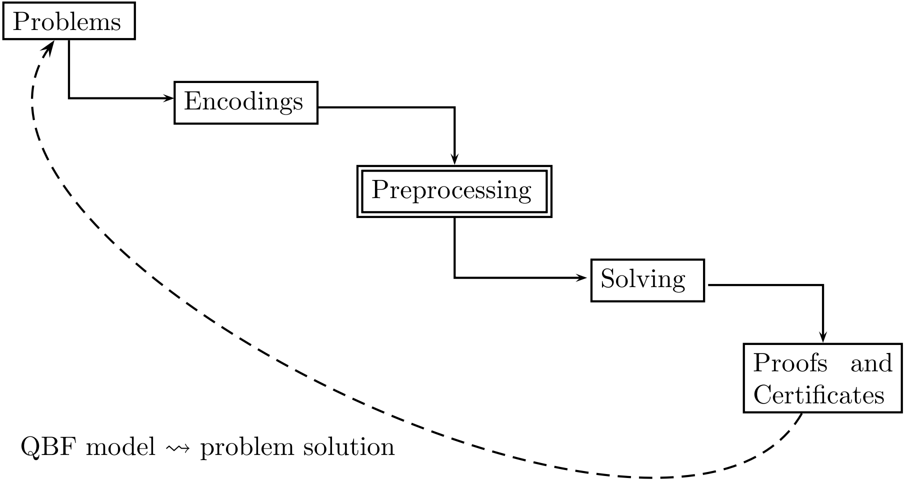
- Elimination of variables, eventually reducing formula to $\top/\bot$.
- Exponential space (worst case).
- Modern approach: CEGAR-based lazy expansion [JKMSC16, RT15].
- $\forall$Exp+Res: proof system [BCJ15, JM13].

$\Rightarrow$ Proof complexity results: orthogonality of Q-resolution and expansion.

Introduction (4): Typical QBF Workflow



Introduction (4): Typical QBF Workflow



Our focus: preprocessing

Introduction (5)

QBF Preprocessing:

- Elimination/addition of redundant clauses and literals in PCNF $\Pi.\phi$.
- Equivalence-preserving inference steps: $\Pi.(\phi \wedge C) \equiv \Pi.\phi$.
- Satisfiability-preserving interference steps: $\Pi.(\phi \wedge C) \equiv_{sat} \Pi.\phi$.
- Inferences/interferences relevant in proof systems, cf. [HK17].

QBF Preprocessing via QRAT **Proof System:** [HSB14, HSB17]

- Redundancy property: "quantified resolution asymmetric tautology."
- Incomplete, poly-time inferences checks by unit propagation on *quantifier-free* ϕ.

$\psi := \Pi.(\phi \wedge C) \longrightarrow$ [Inferences checks wrt. ϕ: Has C QRAT?] $\underbrace{\longrightarrow \quad \Pi.\phi}_{\text{Deleting } C\text{: interference}}$

Introduction (5)

QBF Preprocessing:

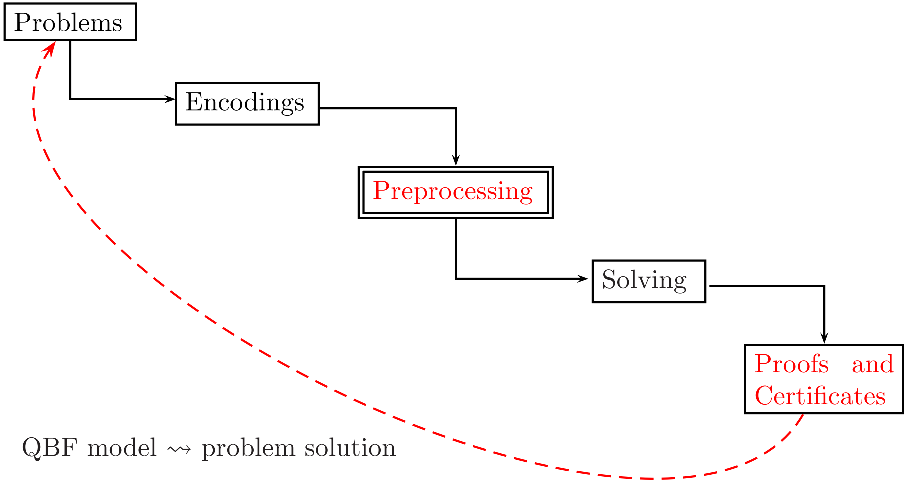
- Elimination/addition of redundant clauses and literals in PCNF $\Pi.\phi$.
- Equivalence-preserving inference steps: $\Pi.(\phi \wedge C) \equiv \Pi.\phi$.
- Satisfiability-preserving interference steps: $\Pi.(\phi \wedge C) \equiv_{sat} \Pi.\phi$.
- Inferences/interferences relevant in proof systems, cf. [HK17].

QBF Preprocessing via QRAT **Proof System:** [HSB14, HSB17]

- Redundancy property: "quantified resolution asymmetric tautology."
- Incomplete, poly-time inferences checks by unit propagation on *quantifier-free* ϕ.

$\psi := \Pi.(\phi \wedge C) \longrightarrow$ [Inferences checks wrt. ϕ: Has C QRAT?] $\underbrace{\longrightarrow \quad \Pi.\phi}_{\text{Deleting } C\text{: interference}}$

Introduction (6): Relevance of QRAT in Practice



Like DRAT [WHH14], QRAT simulates all QBF reasoning techniques.

QRAT$^+$ Proof System: Generalization of QRAT

$\psi := \Pi.(\phi \wedge C) \longrightarrow$ Inferences checks wrt. $\Pi.\phi$: Has C QRAT$^+$? $\underbrace{\longrightarrow \quad \Pi.\phi}_{\text{Deleting } C\text{: interference}}$

- Joint work with Uwe Egly: IJCAR 2018 [LE18b].
- Generalizes inference checking in QRAT based on quantifier prefix.
- Incomplete, poly-time inferences checks by *QBF unit propagation* to *leverage quantifier structure*.
- More powerful interferences by QRAT$^+$ redundancy property.
- Proof-theoretical impact of QRAT/QRAT$^+$ redundancy removal.
- Tool *QRATPre+*: QRAT$^+$ redundancy removal for QBF preprocessing.
- Experimental results: positive impact on QBF solver performance.

The Original QRAT Proof System (1)

Theorem ([HSB17]; QRAT-based interferences)

Given a PCNF $\psi := \Pi.\big(\phi \wedge (C' \cup \{l\})\big)$ *with a clause* $C = (C' \cup \{l\})$.
If clause C *has* QRAT *on literal* l *with respect to* $\Pi.\phi$ *and*

1. $q(l) = \exists$, *then* $\psi \equiv_{sat} \Pi.\phi$. *(add/delete clauses)*
2. $q(l) = \forall$, *then* $\psi \equiv_{sat} \Pi.(\phi \wedge C')$. *(add/delete literals)*

- QRAT property of clauses: "quantified resolution asymm. tautology."
- QRAT checking: inference checking in *resolution neighborhood* of C.

Definition (cf. [Kul99, JHB12, KSTB17])

Given a PCNF $\psi := \Pi.\big(\phi \wedge (C' \cup \{l\})\big)$ with a clause $C = (C' \cup \{l\})$.
Resolution neighborhood (RN) of C with respect to $l \in C$:
$\mathrm{RN}(C, l) := \{D \mid D \in \phi, \bar{l} \in D\}$.

The Original QRAT Proof System (1)

Theorem ([HSB17]; QRAT-based interferences)

Given a PCNF $\psi := \Pi.\big(\phi \wedge (C' \cup \{l\})\big)$ *with a clause* $C = (C' \cup \{l\})$*.*
If clause C *has* QRAT *on literal* l *with respect to* $\Pi.\phi$ *and*

1. $q(l) = \exists$*, then* $\psi \equiv_{sat} \Pi.\phi$*.* *(add/delete clauses)*
2. $q(l) = \forall$*, then* $\psi \equiv_{sat} \Pi.(\phi \wedge C')$*.* *(add/delete literals)*

- QRAT property of clauses: "quantified resolution asymm. tautology."
- QRAT checking: inference checking in *resolution neighborhood* of C.

Definition (cf. [Kul99, JHB12, KSTB17])

Given a PCNF $\psi := \Pi.\big(\phi \wedge (C' \cup \{l\})\big)$ with a clause $C = (C' \cup \{l\})$.
Resolution neighborhood (RN) of C with respect to $l \in C$:
$\text{RN}(C, l) := \{D \mid D \in \phi, \bar{l} \in D\}$.

The Original QRAT Proof System (2)

Definition ([HSB17]; informally)

Given a PCNF $\psi := \Pi.\Big(\phi \wedge (C' \cup \{l\})\Big)$, $C = (C' \cup \{l\})$, $D \in \text{RN}(C, l)$.
Outer resolvent (OR) of C and D on l: $\text{OR}(C, D, l) \subset (C \cup D)$.

Definition ([HSB17]; propositional inference checking)

Given a PCNF $\psi := \Pi.\Big(\phi \wedge (C' \cup \{l\})\Big)$ with a clause $C = (C' \cup \{l\})$.
Towards checking whether C has QRAT on l:

- For *all* $D \in \text{RN}(C, l)$, consider outer resolvent $\text{OR} := \text{OR}(C, D, l)$.
- Propositional implication check:
  $\phi \models \text{OR}$, i.e. $\phi \equiv \phi \wedge \text{OR}$?
- $\phi \models \text{OR}$ iff $\phi \rightarrow \text{OR}$ valid iff $\phi \wedge \overline{\text{OR}}$ unsatisfiable.

Problem: computationally hard (co-NP) propositional implication check.

The Original QRAT Proof System (2)

Definition ([HSB17]; informally)

Given a PCNF $\psi := \Pi.\Big(\phi \wedge (C' \cup \{l\})\Big)$, $C = (C' \cup \{l\})$, $D \in \mathrm{RN}(C, l)$.
Outer resolvent (OR) of C and D on l: $\mathrm{OR}(C, D, l) \subset (C \cup D)$.

Definition ([HSB17]; propositional inference checking)

Given a PCNF $\psi := \Pi.\Big(\phi \wedge (C' \cup \{l\})\Big)$ with a clause $C = (C' \cup \{l\})$.
Towards checking whether C has QRAT on l:

- For *all* $D \in \mathrm{RN}(C, l)$, consider outer resolvent $\mathrm{OR} := \mathrm{OR}(C, D, l)$.
- Propositional implication check:
  $\phi \models \mathrm{OR}$, i.e. $\phi \equiv \phi \wedge \mathrm{OR}$?
- $\phi \models \mathrm{OR}$ iff $\phi \rightarrow \mathrm{OR}$ valid iff $\phi \wedge \overline{\mathrm{OR}}$ unsatisfiable.

Problem: computationally hard (co-NP) propositional implication check.

The Original QRAT Proof System (3)

Definition ([HSB17]; incomplete propositional inference checks)

Given a PCNF $\psi := \Pi.\Big(\phi \wedge (C' \cup \{l\})\Big)$ with a clause $C = (C' \cup \{l\})$.
Checking whether C has QRAT on l in poly-time by unit propagation:

- For *all* $D \in \mathsf{RN}(C, l)$, consider outer resolvent $\mathsf{OR} := \mathsf{OR}(C, D, l)$.
- Propagate $\overline{\mathsf{OR}}$ on CNF ϕ to get empty clause ($\emptyset$):
  $\phi \wedge \overline{\mathsf{OR}} \vdash_1 \emptyset$?
- If $\phi \wedge \overline{\mathsf{OR}} \vdash_1 \emptyset$ then $\phi \wedge \overline{\mathsf{OR}}$ unsatisfiable:
  $\phi \models \mathsf{OR}$ and hence $\phi \equiv \phi \wedge \mathsf{OR}$.
- Otherwise, if $\phi \wedge \overline{\mathsf{OR}} \nvdash_1 \emptyset$ then C does not have QRAT on l.

The Original QRAT Proof System (3)

Example (inference checks by unit propagation)

PCNF $\psi := \Pi.\phi$, $\Pi := \forall u_1 \exists x_3 \forall u_2 \exists x_4$, $\phi := (u_1 \vee \bar{x}_3 \vee x_4) \wedge (\bar{u}_2 \vee \bar{x}_4)$.
Let $C := (u_1 \vee \bar{x}_3)$ and check if $\phi \models C$ by unit propagation.

- Propagating $\overline{C} = (\bar{u}_1) \wedge (x_3)$ on CNF ϕ: $(x_4) \wedge (\bar{u}_2 \vee \bar{x}_4) \rightsquigarrow (\bar{u}_2)$.
- $\phi \wedge \overline{C} \not\vdash_1 \emptyset$; all variables, including $\forall u_2$, are existential in CNF ϕ.
- Propositional unit propagation on ϕ is *not* aware of quantifiers in ψ.
- However, in this example: $\Pi.\phi \models \Pi.(\phi \wedge C)$.

The Original QRAT Proof System (3)

Example (inference checks by unit propagation)

PCNF $\psi := \Pi.\phi$, $\Pi := \forall u_1 \exists x_3 \forall u_2 \exists x_4$, $\phi := (u_1 \vee \bar{x}_3 \vee x_4) \wedge (\bar{u}_2 \vee \bar{x}_4)$.
Let $C := (u_1 \vee \bar{x}_3)$ and check if $\phi \models C$ by unit propagation.

- Propagating $\overline{C} = (\bar{u}_1) \wedge (x_3)$ on CNF ϕ: $(x_4) \wedge (\bar{u}_2 \vee \bar{x}_4) \rightsquigarrow (\bar{u}_2)$.
- $\phi \wedge \overline{C} \not\vdash_1 \emptyset$; all variables, including $\forall u_2$, are existential in CNF ϕ.
- Propositional unit propagation on ϕ is *not* aware of quantifiers in ψ.
- However, in this example: $\Pi.\phi \models \Pi.(\phi \wedge C)$.

$\Rightarrow$ Leverage PCNF quantifier structure for stronger propagation.

$\Rightarrow$ Checking QBF implication: $\Pi.\phi$ implies $\Pi.(\phi \wedge C)$.

QRAT$^+$: The Big Picture

$\psi := \Pi.(\phi \wedge C) \longrightarrow$ [Inferences checks wrt. ϕ: Has C QRAT?] $\underbrace{\longrightarrow \quad \Pi.\phi}_{\text{Deleting } C\text{: interference}}$

Theorem ([HSB17]; QRAT-based interferences)

Given a PCNF $\psi := \Pi.\big(\phi \wedge (C' \cup \{l\})\big)$ *with a clause* $C = (C' \cup \{l\})$.
If clause C *has* QRAT *on literal* l *with respect to* $\Pi.\phi$ *and*

1. $q(l) = \exists$, *then* $\psi \equiv_{sat} \Pi.\phi$. *(add/delete clauses)*
2. $q(l) = \forall$, *then* $\psi \equiv_{sat} \Pi.(\phi \wedge C')$. *(add/delete literals)*

QRAT$^+$: The Big Picture

$\psi := \Pi.(\phi \wedge C) \longrightarrow$ **Inferences** checks wrt. $\Pi.\phi$: Has C QRAT$^+$? $\underbrace{\longrightarrow \quad \Pi.\phi}_{\text{Deleting } C\text{: interference}}$

Theorem (QRAT$^+$-based interferences)

Given a PCNF $\psi := \Pi.\Big(\phi \wedge (C' \cup \{l\})\Big)$ with a clause $C = (C' \cup \{l\})$.

If clause C has QRAT$^+$ on literal l with respect to $\Pi.\phi$ and

1. $q(l) = \exists$, *then* $\psi \equiv_{sat} \Pi.\phi$. *(add/delete clauses)*
2. $q(l) = \forall$, *then* $\psi \equiv_{sat} \Pi.(\phi \wedge C')$. *(add/delete literals)*

- Interferences like QRAT: QRAT$^+$ clause/literal redundancy property.
- QRAT$^+$ interferences: more powerful, more general than QRAT.
- QRAT vs. QRAT$^+$: leveraging QBF quantifier structure in inference checking by a variant of unit propagation.

QRAT$^+$: More General Inference Checking

Definition ([HSB17]; propositional inference checking)

Given a PCNF $\psi := \Pi.\Big(\phi \wedge (C' \cup \{l\})\Big)$ with a clause $C = (C' \cup \{l\})$.
Towards checking whether C has QRAT on l:

- For *all* $D \in \mathrm{RN}(C, l)$, consider outer resolvent $\mathrm{OR} := \mathrm{OR}(C, D, l)$.
- Propositional implication check:
  $\phi \models \mathrm{OR}$, i.e. $\phi \equiv \phi \wedge \mathrm{OR}$?

QRAT$^+$: More General Inference Checking

Definition (QBF inference checking)

Given a PCNF $\psi := \Pi.\Big(\phi \wedge (C' \cup \{l\})\Big)$ with a clause $C = (C' \cup \{l\})$.

Towards checking whether C has QRAT$^+$ on l:

- For *all* $D \in \mathrm{RN}(C, l)$, consider outer resolvent $\mathrm{OR} := \mathrm{OR}(C, D, l)$.
- QBF implication check:
  $\Pi.\phi \models \Pi.(\phi \wedge \mathrm{OR})$, i.e. $\Pi.\phi \equiv_t \Pi.(\phi \wedge \mathrm{OR})$?

- Inference checking in QRAT$^+$: implication checking on QBF level.
- Generalization of propositional implication checking in QRAT.
- Known fact [SDB06]: for CNFs ϕ and ϕ', if $\phi \equiv \phi'$ then $\Pi.\phi \equiv_t \Pi.\phi'$.
- In general, if $\Pi.\phi \equiv_t \Pi.\phi'$ then $\phi \not\equiv \phi'$.
- **Problem (again):** computationally hard QBF implication check.

QRAT$^+$: QBF Unit Propagation (1)

Definition ([KBKF95])

Given a PCNF $\psi := \Pi.\phi$ and a non-tautological clause C, *universal reduction (UR)* of C produces the clause

$$UR(C) := C \setminus \{l \in C \mid q(l) = \forall, \forall l' \in C \text{ with } q(l') = \exists : l' < l\}$$

- Local deletion of "trailing" universal literals by prefix ordering '$<$'.
- *QBF unit propagation*: propositional unit propagation + UR.
- Notation $\Pi.\phi \vdash_{1\forall} \emptyset$: propagation on QBF $\Pi.\phi$ results in empty clause.

Example

Given PCNF $\psi := \Pi.\phi$ with $\Pi := \forall u_1 \exists x_3 \forall u_2 \exists x_4$.

- For clause $C := (x_3 \vee \bar{u}_2 \vee x_4)$, $UR(C) = C$ since $u_2 < x_4$, $q(x_4) = \exists$.
- For clause $C' := (u_1 \vee \bar{x}_3 \vee \bar{u}_2)$, $UR(C') = (u_1 \vee \bar{x}_3)$ since u_2 trailing.
- **Note:** UR leverages quantifier structure to eliminate universal literals.

QRAT$^+$: QBF Unit Propagation (2)

Definition (QBF inference checking)

Given a PCNF $\psi := \Pi.\big(\phi \wedge (C' \cup \{l\})\big)$ with a clause $C = (C' \cup \{l\})$.

Towards checking whether C has QRAT$^+$ on l:

- For *all* $D \in \text{RN}(C, l)$, consider outer resolvent $\text{OR} := \text{OR}(C, D, l)$.
- Computationally hard QBF implication check: $\Pi.\phi \models \Pi.(\phi \wedge \text{OR})$, i.e. $\Pi.\phi \equiv_t \Pi.(\phi \wedge \text{OR})$?
- **Goal (pitfall!):** $\Pi.(\phi \wedge \overline{\text{OR}}) \vdash_{1\forall} \emptyset$ such that $\Pi.\phi \models \Pi.(\phi \wedge \text{OR})$.

Unsoundness: if $\Pi.(\phi \wedge \overline{C}) \vdash_{1\forall} \emptyset$ then $\Pi.\phi \not\models \Pi.(\phi \wedge C)$

- Given QBF $\Pi.\phi$ with $\Pi := \forall u \exists x$, $\phi := (u \vee \bar{x})$, and $C := (x)$.
- $\Pi.\phi$ satisfiable, $\Pi.(\phi \wedge \overline{C}) \vdash_{1\forall} \emptyset$ but $\Pi.(\phi \wedge (x))$ unsatisfiable.

QRAT$^+$: QBF Unit Propagation on Abstractions (1)

Definition

Given a PCNF $\psi := \Pi.\phi$ with prefix $\Pi := Q_1 B_1 \dots Q_i B_i Q_{i+1} B_{i+1} \dots Q_n B_n$.

- *Abstraction* of ψ with respect to block i: $Abs(\psi, i) := Abs(\Pi, i).\phi$.
- $Abs(\Pi, i) := \underbrace{\exists(B_1 \cup \dots \cup B_i)}_{abstracted} \underbrace{Q_{i+1} B_{i+1} \dots Q_n B_n}_{original}$
- Leftmost quantifiers up to $Q_i B_i$ are all existential.
- Quantifier-free CNF ϕ unchanged.

Example

Given a PCNF $\psi := \Pi.\phi$ with prefix $\Pi := \forall B_1 \exists B_2 \forall B_3 \exists B_4$.

- $Abs(\psi, 0) = \psi$
- $Abs(\psi, 1) = Abs(\psi, 2) = \exists(B_1 \cup B_2)\forall B_3 \exists B_4.\phi$
- $Abs(\psi, 3) = Abs(\psi, 4) = \exists(B_1 \cup B_2 \cup B_3 \cup B_4).\phi$

QRAT$^+$: QBF Unit Propagation on Abstractions (2)

Lemma

Let $\psi := \Pi.\phi$ *and* $\psi' := \Pi.\phi'$ *be QBFs with the same prefix* $\Pi := Q_1 B_1 \ldots Q_i B_i \ldots Q_n B_n$.

- *For all* i, *if* $Abs(\psi, i) \equiv_t Abs(\psi', i)$ *then* $\psi \equiv_t \psi'$.

Lemma

Given a $\Pi.\phi$ *and a clause* C.
Let $i = \max(levels(\Pi, C))$ *be the maximum nesting level of variables in* C.

- *If* $Abs(\Pi.(\phi \wedge \overline{C}), i) \vdash_{1\forall} \emptyset$ *then* $\Pi.\phi \equiv_t \Pi.(\phi \wedge C)$.

- Check QBF implication of some clause C based on abstractions.
- Apply QBF unit propagation to *suitable* abstractions wrt. C.
- If empty clause derived on abstraction of $\Pi.\phi$, then C implied by $\Pi.\phi$.

QRAT$^+$: QBF Unit Propagation on Abstractions (3)

Example

PCNF $\psi := \Pi.\phi$, $\Pi := \exists x_1, x_2 \forall u_1 \exists x_3 \forall u_2 \exists x_4$ and CNF ϕ as follows.

$$
\begin{aligned}
C_1 &:= (x_2 \vee \bar{u}_1 \vee x_3) \\
C_2 &:= (\bar{x}_1 \vee \bar{u}_1 \vee \bar{x}_3) \\
C_3 &:= (\bar{x}_2 \vee u_1 \vee x_3) \\
C_4 &:= (u_1 \vee \bar{x}_3 \vee x_4) \\
C_5 &:= (\bar{u}_2 \vee \bar{x}_4) \\
C_6 &:= (\bar{x}_1 \vee u_2 \vee \bar{x}_4)
\end{aligned}
$$

Check if $C := (u_1 \vee \bar{x}_3)$ implied by ψ by QBF unit prop. on abstraction:

- If $Abs(\Pi.(\phi \wedge \overline{C}), i) \vdash_{1\forall} \emptyset$ then $\Pi.\phi \equiv_t \Pi.(\phi \wedge C)$.
- Maximum nesting level in C: $i = \max(levels(\Pi, C)) = 3$.

QRAT$^+$: QBF Unit Propagation on Abstractions (3)

Example

PCNF $\psi := \Pi.\phi$, $\Pi := \exists x_1, x_2 \exists u_1 \exists x_3 \forall u_2 \exists x_4$ and CNF ϕ as follows.

$$
\begin{aligned}
C_1 &:= (x_2 \vee \bar{u}_1 \vee x_3)\\
C_2 &:= (\bar{x}_1 \vee \bar{u}_1 \vee \bar{x}_3)\\
C_3 &:= (\bar{x}_2 \vee u_1 \vee x_3)\\
C_4 &:= (u_1 \vee \bar{x}_3 \vee x_4)\\
C_5 &:= (\bar{u}_2 \vee \bar{x}_4)\\
C_6 &:= (\bar{x}_1 \vee u_2 \vee \bar{x}_4)
\end{aligned}
$$

- Propagate $\overline{C} = (\bar{u}_1) \wedge (x_3)$

Check if $C := (u_1 \vee \bar{x}_3)$ implied by ψ by QBF unit prop. on abstraction:

- If $Abs(\Pi.(\phi \wedge \overline{C}), i) \vdash_{1\forall} \emptyset$ then $\Pi.\phi \equiv_t \Pi.(\phi \wedge C)$.
- Maximum nesting level in C: $i = \max(levels(\Pi, C)) = 3$.
- Abstract wrt. quantifier block $i = 3$.

QRAT$^+$: QBF Unit Propagation on Abstractions (3)

Example

PCNF $\psi := \Pi.\phi$, $\Pi := \exists x_1, x_2 \exists u_1 \exists x_3 \forall u_2 \exists x_4$ and CNF ϕ as follows.

$C_1 := (x_2 \vee \bar{u}_1 \vee x_3)$
$C_2 := (\bar{x}_1 \vee \bar{u}_1 \vee \bar{x}_3)$
$C_3 := (\bar{x}_2 \vee u_1 \vee x_3)$
$C_4 := (u_1 \vee \bar{x}_3 \vee x_4)$
$C_5 := (\bar{u}_2 \vee \bar{x}_4)$
$C_6 := (\bar{x}_1 \vee u_2 \vee \bar{x}_4)$

- Propagate $\overline{C} = (\bar{u}_1) \wedge (x_3)$
- Simplify, propagate (x_4) resulting from C_4.
- C_5 becomes empty since u_2 still universal.
- Hence $Abs(\Pi.(\phi \wedge \overline{C}), i) \vdash_{1\forall} \emptyset$.

Check if $C := (u_1 \vee \bar{x}_3)$ implied by ψ by QBF unit prop. on abstraction:

- If $Abs(\Pi.(\phi \wedge \overline{C}), i) \vdash_{1\forall} \emptyset$ then $\Pi.\phi \equiv_t \Pi.(\phi \wedge C)$.
- Maximum nesting level in C: $i = \max(levels(\Pi, C)) = 3$.
- Abstract wrt. quantifier block $i = 3$.

QRAT$^+$: Final View

Definition ([HSB17]; incomplete propositional inference checks)

Given a PCNF $\psi := \Pi.\Big(\phi \wedge (C' \cup \{l\})\Big)$ with a clause $C = (C' \cup \{l\})$.
Checking whether C has QRAT on l in poly-time by unit propagation:

- For *all* $D \in \text{RN}(C, l)$, consider outer resolvent $\text{OR} := \text{OR}(C, D, l)$.
- Propagate $\overline{\text{OR}}$ on CNF ϕ to get empty clause ($\emptyset$):
  $\phi \wedge \overline{\text{OR}} \vdash_1 \emptyset$?
- If $\phi \wedge \overline{\text{OR}} \vdash_1 \emptyset$ then $\phi \equiv \phi \wedge \text{OR}$.
- Otherwise, if $\phi \wedge \overline{\text{OR}} \nvdash_1 \emptyset$ then C does not have QRAT on l.

$\psi := \Pi.(\phi \wedge C) \longrightarrow$ [Inferences checks wrt. ϕ: Has C QRAT?] $\underbrace{\longrightarrow \quad \Pi.\phi}$

Deleting C: interference

QRAT$^+$: Final View

Definition (incomplete QBF inference checks)

Given a PCNF $\psi := \Pi.\Big(\phi \wedge (C' \cup \{l\})\Big)$ with a clause $C = (C' \cup \{l\})$.

Checking whether C has QRAT$^+$ on l in poly-time by QBF unit prop.:

- For *all* $D \in \mathrm{RN}(C, l)$, consider outer resolvent $\mathrm{OR} := \mathrm{OR}(C, D, l)$.
- Let $i = \max(\mathit{levels}(\Pi, \mathrm{OR}))$ be the maximum nesting level in OR.
- Consider the abstraction of $\Pi.\phi$ with respect to block i.
- If $\mathit{Abs}(\Pi.(\phi \wedge \overline{\mathrm{OR}}), i) \vdash_{1\forall} \emptyset$ then $\Pi.\phi \equiv_t \Pi.(\phi \wedge \mathrm{OR})$.
- If $\mathit{Abs}(\Pi.(\phi \wedge \overline{\mathrm{OR}}), i) \nvdash_{1\forall} \emptyset$ then C does not have QRAT$^+$ on literal l.

$\psi := \Pi.(\phi \wedge C) \longrightarrow$ [Inferences checks wrt. $\Pi.\phi$: Has C QRAT$^+$?] $\underbrace{\longrightarrow \quad \Pi.\phi}_{\text{Deleting } C\text{: interference}}$

- Abstractions: crucial for *sound and poly-time* QRAT$^+$ checking.

QRAT$^+$: Outer Resolvents and Abstractions

$D_1 = D'_1 \cup \{\bar{l}\} \ \ldots \qquad D_j = D'_j \cup \{\bar{l}\} \ \ldots \qquad D_n = D'_n \cup \{\bar{l}\}$

$\mathsf{OR}_1 := \mathsf{OR}(C, D_1, l) \subset (C \cup D_1)$
$i = max(levels(\Pi, \mathsf{OR}_1)) = 3$
$\Pi = \exists B_1 \forall B_2 \exists B_3 \forall B_4 \exists B_5$
$Abs(\Pi, i) := \exists B_1 \exists B_2 \exists B_3 \forall B_4 \exists B_5$
Check: $Abs(\Pi.(\phi \wedge \overline{\mathsf{OR}_1}), i) \vdash_{1\forall} \emptyset$?

$C = C' \cup \{l\}$

- Abstractions may differ with respect to current outer resolvent.
- Different numbers of universal variables in resp. abstracted formula.
- Quantifier structure is leveraged to a different extent.

QRAT$^+$: Outer Resolvents and Abstractions

$D_1 = D'_1 \cup \{\bar{l}\}$ ... $D_j = D'_j \cup \{\bar{l}\}$ ... $D_n = D'_n \cup \{\bar{l}\}$

$$\mathsf{OR}_j := \mathsf{OR}(C, D_j, l) \subset (C \cup D_j)$$
$$i = max(levels(\Pi, \mathsf{OR}_j)) = 1$$
$$\Pi = \exists B_1 \forall B_2 \exists B_3 \forall B_4 \exists B_5$$
$$Abs(\Pi, i) := \exists B_1 \forall B_2 \exists B_3 \forall B_4 \exists B_5$$
$$\text{Check: } Abs(\Pi.(\phi \wedge \overline{\mathsf{OR}_j}), i) \vdash_{1\forall} \emptyset?$$

$$C = C' \cup \{l\}$$

- Abstractions may differ with respect to current outer resolvent.
- Different numbers of universal variables in resp. abstracted formula.
- Quantifier structure is leveraged to a different extent.

QRAT$^+$: Outer Resolvents and Abstractions

$D_1 = D'_1 \cup \{\bar{l}\}$ ... $D_j = D'_j \cup \{\bar{l}\}$ ... $D_n = D'_n \cup \{\bar{l}\}$

$$\mathsf{OR}_n := \mathsf{OR}(C, D_n, l) \subset (C \cup D_n)$$
$$i = max(levels(\Pi, \mathsf{OR}_n)) = 5$$
$$\Pi = \exists B_1 \forall B_2 \exists B_3 \forall B_4 \exists B_5$$
$$Abs(\Pi, i) := \exists B_1 \exists B_2 \exists B_3 \exists B_4 \exists B_5$$
$$\text{Check: } Abs(\Pi.(\phi \wedge \overline{\mathsf{OR}_n}), i) \vdash_{1\forall} \emptyset?$$

$C = C' \cup \{l\}$

- Abstractions may differ with respect to current outer resolvent.
- Different numbers of universal variables in resp. abstracted formula.
- Quantifier structure is leveraged to a different extent.

The Power of QRAT and QRAT$^+$

Proposition (cf. example in paper [LE18b])

There exists a class of PCNFs where every clause C has QRAT$^+$*, but not* QRAT*, on an existential literal, i.e. C can be eliminated by* QRAT$^+$ *only.*

- More powerful QRAT$^+$ interferences: adding/deleting clauses.
- Abstractions in QBF unit propagation are crucial for this observation.
- Similar result for elimination of universal literals by QRAT$^+$.

The Power of QRAT and QRAT$^+$

Proposition (cf. example in paper [LE18b])

Eliminating universal literals by QRAT *or* QRAT$^+$ *in PCNFs may result in exponentially shorter proofs in the LQU$^+$-resolution calculus.*

- LQU$^+$-resolution [BWJ14]: strongest resolution-based calculus.
- Impact of QRAT/QRAT$^+$-interferences on power of proof systems.
- Enabling proof steps by redundancy elimination, cf. [HK17].

Shortening Proofs with QRAT and QRAT$^+$

Definition (QUParity(n) [BCJ15])

QUParity(n) ($n > 1$) is a QBF with prefix $\Pi^n_Q := \exists x_1, \ldots, x_n \forall z_1, z_2 \exists t_2, \ldots, t_n$ and CNF $\phi^n_Q := C_0 \wedge C_1 \wedge \bigwedge_{i:=2}^{n} \mathcal{C}(i)$ with $C_0 := (z_1 \vee z_2 \vee t_n)$, $C_1 := (\bar{z}_1 \vee \bar{z}_2 \vee \bar{t}_n)$, and $\mathcal{C}(i) := \bigwedge_{j:=0}^{7} C_{i,j}$ defined as follows.

$$
\begin{array}{llll}
C_{2,0} := & (\bar{x}_1 \vee \bar{x}_2 \vee z_1 \vee z_2 \vee \bar{t}_2) & C_{i,0} := & (\bar{t}_{i-1} \vee \bar{x}_i \vee z_1 \vee z_2 \vee \bar{t}_i) \\
C_{2,1} := & (x_1 \vee x_2 \vee z_1 \vee z_2 \vee \bar{t}_2) & C_{i,1} := & (t_{i-1} \vee x_i \vee z_1 \vee z_2 \vee \bar{t}_i) \\
C_{2,2} := & (\bar{x}_1 \vee x_2 \vee z_1 \vee z_2 \vee t_2) & C_{i,2} := & (\bar{t}_{i-1} \vee x_i \vee z_1 \vee z_2 \vee t_i) \\
C_{2,3} := & (x_1 \vee \bar{x}_2 \vee z_1 \vee z_2 \vee t_2) & C_{i,3} := & (t_{i-1} \vee \bar{x}_i \vee z_1 \vee z_2 \vee t_i) \\
C_{2,4} := & (\bar{x}_1 \vee \bar{x}_2 \vee \bar{z}_1 \vee \bar{z}_2 \vee \bar{t}_2) & C_{i,4} := & (\bar{t}_{i-1} \vee \bar{x}_i \vee \bar{z}_1 \vee \bar{z}_2 \vee \bar{t}_i) \\
C_{2,5} := & (x_1 \vee x_2 \vee \bar{z}_1 \vee \bar{z}_2 \vee \bar{t}_2) & C_{i,5} := & (t_{i-1} \vee x_i \vee \bar{z}_1 \vee \bar{z}_2 \vee \bar{t}_i) \\
C_{2,6} := & (\bar{x}_1 \vee x_2 \vee \bar{z}_1 \vee \bar{z}_2 \vee t_2) & C_{i,6} := & (\bar{t}_{i-1} \vee x_i \vee \bar{z}_1 \vee \bar{z}_2 \vee t_i) \\
C_{2,7} := & (x_1 \vee \bar{x}_2 \vee \bar{z}_1 \vee \bar{z}_2 \vee t_2) & C_{i,7} := & (t_{i-1} \vee \bar{x}_i \vee \bar{z}_1 \vee \bar{z}_2 \vee t_i)
\end{array}
$$

LQU$^+$-resolution refutations of QUParity(n) have size exponential in n.

Shortening Proofs with QRAT and QRAT$^+$

$C_0 :=$	$(z_1 \vee z_2 \vee t_n)$	$C_1 :=$	$(\bar{z}_1 \vee \bar{z}_2 \vee \bar{t}_n)$
$C_{2,0} :=$	$(\bar{x}_1 \vee \bar{x}_2 \vee z_1 \vee z_2 \vee \bar{t}_2)$	$C_{i,0} :=$	$(\bar{t}_{i-1} \vee \bar{x}_i \vee z_1 \vee z_2 \vee \bar{t}_i)$
$C_{2,1} :=$	$(x_1 \vee x_2 \vee z_1 \vee z_2 \vee \bar{t}_2)$	$C_{i,1} :=$	$(t_{i-1} \vee x_i \vee z_1 \vee z_2 \vee \bar{t}_i)$
$C_{2,2} :=$	$(\bar{x}_1 \vee x_2 \vee z_1 \vee z_2 \vee t_2)$	$C_{i,2} :=$	$(\bar{t}_{i-1} \vee x_i \vee z_1 \vee z_2 \vee t_i)$
$C_{2,3} :=$	$(x_1 \vee \bar{x}_2 \vee z_1 \vee z_2 \vee t_2)$	$C_{i,3} :=$	$(t_{i-1} \vee \bar{x}_i \vee z_1 \vee z_2 \vee t_i)$
$C_{2,4} :=$	$(\bar{x}_1 \vee \bar{x}_2 \vee \bar{z}_1 \vee \bar{z}_2 \vee \bar{t}_2)$	$C_{i,4} :=$	$(\bar{t}_{i-1} \vee \bar{x}_i \vee \bar{z}_1 \vee \bar{z}_2 \vee \bar{t}_i)$
$C_{2,5} :=$	$(x_1 \vee x_2 \vee \bar{z}_1 \vee \bar{z}_2 \vee \bar{t}_2)$	$C_{i,5} :=$	$(t_{i-1} \vee x_i \vee \bar{z}_1 \vee \bar{z}_2 \vee \bar{t}_i)$
$C_{2,6} :=$	$(\bar{x}_1 \vee x_2 \vee \bar{z}_1 \vee \bar{z}_2 \vee t_2)$	$C_{i,6} :=$	$(\bar{t}_{i-1} \vee x_i \vee \bar{z}_1 \vee \bar{z}_2 \vee t_i)$
$C_{2,7} :=$	$(x_1 \vee \bar{x}_2 \vee \bar{z}_1 \vee \bar{z}_2 \vee t_2)$	$C_{i,7} :=$	$(t_{i-1} \vee \bar{x}_i \vee \bar{z}_1 \vee \bar{z}_2 \vee t_i)$

- Any occurrence of the literal $\bar{z}_2$ has QRAT and can be deleted.
- After the deletion of $\bar{z}_2$, the variable z_2 is pure and can be deleted.
- Simplified formula has a short refutation in LQU$^+$-resolution.
- Related result [KHS17] for weaker QU-resolution calculus [VG12].

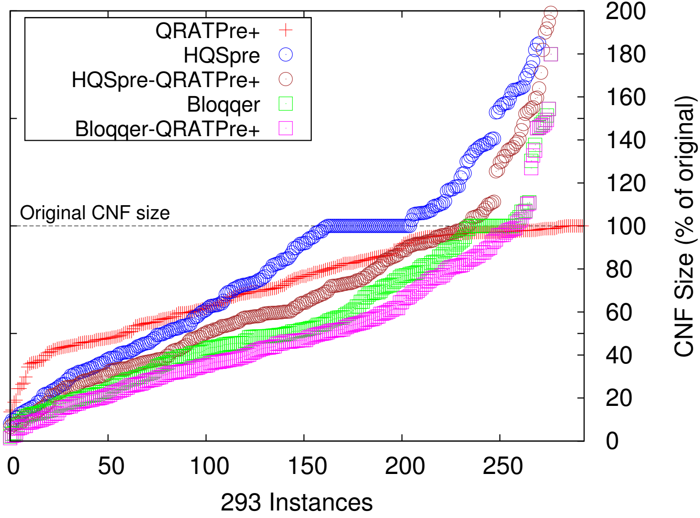
Experiments (1)

Preprocessor "QRATPre+":

- First implementation of *redundancy removal* by QRAT/QRAT$^+$.
- Focus: 523 PCNFs from QBFEVAL'17 competition.
- Impact of QRAT$^+$ less pronounced compared to QRAT: only 32% of instances have > 3 quantifier blocks, cf. [LE18a].
- Clause elimination more effective than universal literal elimination (no effects on 2% resp. 75% of benchmarks).
- Positive effects on QBF solving: more instances solved.

Source code: `https://lonsing.github.io/qratpreplus/`

Experiments (2): Clause Elimination

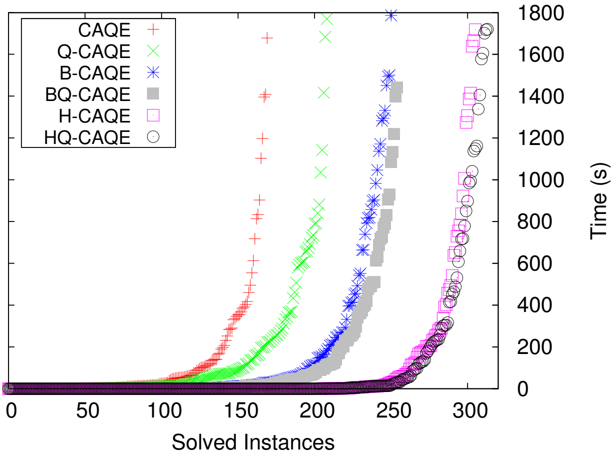


- Variable expansion in state-of-the-art preprocessors Bloqqer and HQSpre may increase CNF sizes, QRATPre+ still effective.

Experiments (3): Solver Performance on 523 PCNFs

Preprocessing:

- "Q": QRATPre+
- "B": Bloqqer
- "H": HQSpre
- "BQ": first Bloqqer, then QRATPre+
- "HQ": first HQSpre, then QRATPre+

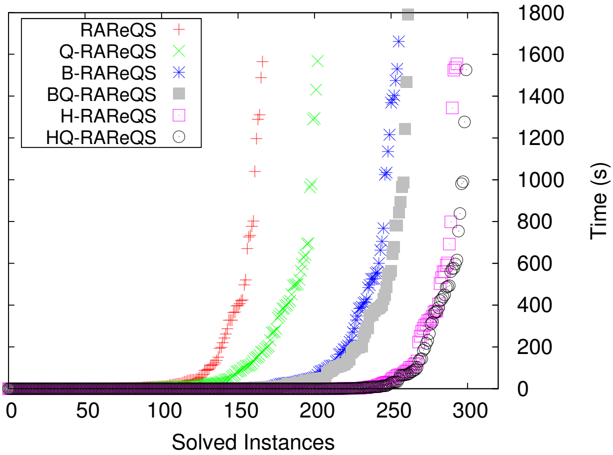


Solver	*Original*	*Q*	*B*	*BQ*	*H*	*HQ*
CAQE	170	209	251	255	306	314

Experiments (3): Solver Performance on 523 PCNFs

Preprocessing:

- "Q": QRATPre+
- "B": Bloqqer
- "H": HQSpre
- "BQ": first Bloqqer, then QRATPre+
- "HQ": first HQSpre, then QRATPre+

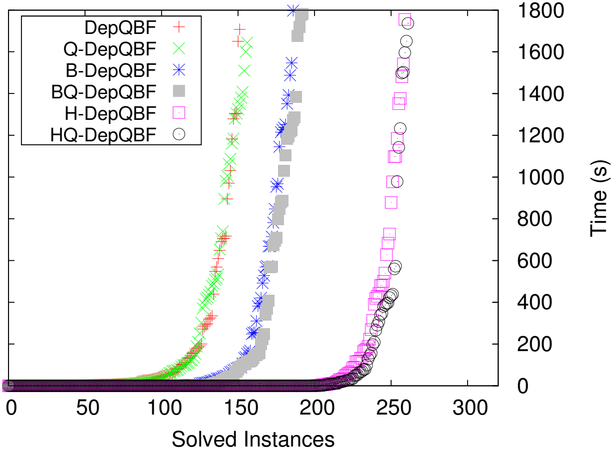


Solver	*Original*	*Q*	*B*	*BQ*	*H*	*HQ*
CAQE	170	209	251	255	306	314
RAReQS	167	203	256	262	294	300

Experiments (3): Solver Performance on 523 PCNFs

Preprocessing:

- "Q": QRATPre+
- "B": Bloqqer
- "H": HQSpre
- "BQ": first Bloqqer, then QRATPre+
- "HQ": first HQSpre, then QRATPre+

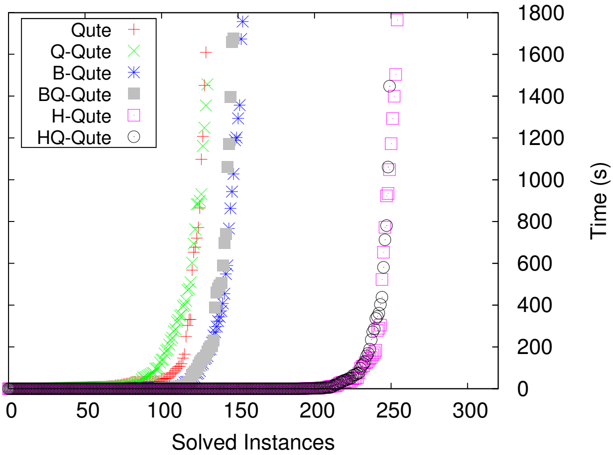


Solver	*Original*	*Q*	*B*	*BQ*	*H*	*HQ*
CAQE	170	209	251	255	306	314
RAReQS	167	203	256	262	294	300
DepQBF	152	157	187	193	260	262

Experiments (3): Solver Performance on 523 PCNFs

Preprocessing:

- "Q": QRATPre+
- "B": Bloqqer
- "H": HQSpre
- "BQ": first Bloqqer, then QRATPre+
- "HQ": first HQSpre, then QRATPre+

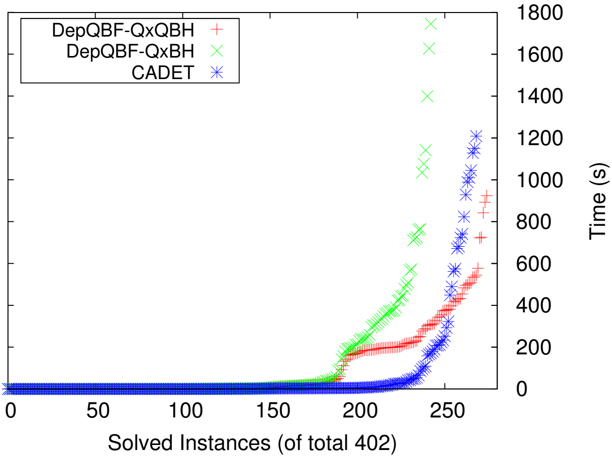


Solver	*Original*	*Q*	*B*	*BQ*	*H*	*HQ*
CAQE	170	209	251	255	306	314
RAReQS	167	203	256	262	294	300
DepQBF	152	157	187	193	260	262
Qute	130	131	154	148	255	250

Experiments (4): 2QBF Track of QBFEVAL 2018

- DepQBF with massive preprocessing, including QRATPre+.
- DepQBF with massive preprocessing, NOT including QRATPre+.
- CADET [RS16, RTRS18], state-of-the-art 2QBF solver, no preprocessing.

FLoC 2018 Olympic Games

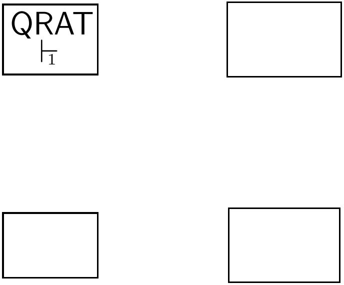


- QRATPre+ removes up to 30% of ∀-literals in instance family "stmt*".
- With/Without QRATPre+: 275 vs. 243 solved instances.
- CADET often faster, solves 269 instances, produces Skolem functions.

Summary

Poly-time inference checks for QRAT:

- Cf. [HSB17].
- $\phi \wedge \overline{\mathrm{OR}(C, D, l)} \vdash_1 \emptyset$?



- Read $\boxed{\text{X}} \rightarrow \boxed{\text{Y}}$ as "Y is more general than X".
- Interferences based on Y are more powerful than based on X.

$\psi := \Pi.(\phi \wedge C) \longrightarrow$ [Inferences checks wrt. ϕ: Has C QRAT?] $\longrightarrow \Pi.\phi$

Deleting C: interference

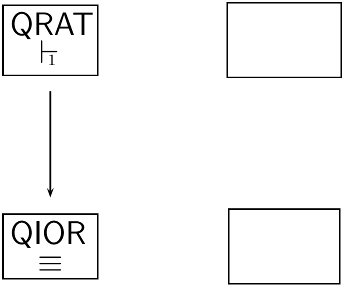
Summary

Poly-time inference checks for QRAT:

- Cf. [HSB17].
- $\phi \wedge \overline{\mathrm{OR}(C, D, l)} \vdash_1 \emptyset$?

Complete inference checks for QIOR:

- Cf. [HSB17].
- quantified implied outer resolvent.
- $\phi \equiv \phi \wedge \mathrm{OR}(\Pi, C, D, l)$?



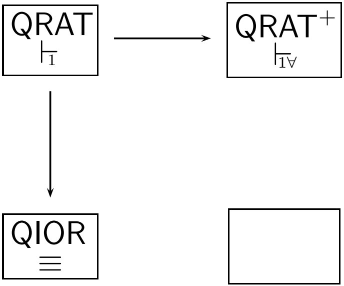
- Read $\boxed{\mathrm{X}} \rightarrow \boxed{\mathrm{Y}}$ as "Y is more general than X".
- Interferences based on Y are more powerful than based on X.

$\psi := \Pi.(\phi \wedge C) \longrightarrow$ [Inferences checks wrt. ϕ: Has C QRAT?] $\underbrace{\longrightarrow \quad \Pi.\phi}_{\text{Deleting } C\text{: interference}}$

Summary

Poly-time inference checks for QRAT$^+$:

- $Abs(\Pi.(\phi \wedge \overline{\mathrm{OR}(C, D, l)}), i) \vdash_{1\forall} \emptyset$?



- Read $\boxed{\mathrm{X}} \rightarrow \boxed{\mathrm{Y}}$ as "Y is more general than X".
- Interferences based on Y are more powerful than based on X.

$\psi := \Pi.(\phi \wedge C) \longrightarrow$ **Inferences checks wrt. $\Pi.\phi$: Has C QRAT$^+$?** $\underbrace{\longrightarrow \quad \Pi.\phi}$

Deleting C: interference

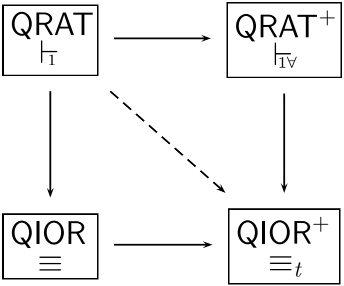
Summary

Poly-time inference checks for QRAT$^+$:

- $Abs(\Pi.(\phi \wedge \overline{\mathrm{OR}(C, D, l)}), i) \vdash_{1\forall} \emptyset$?

Complete inference checks for QIOR$^+$:

- quantified implied outer resolvent.
- $\Pi.\phi \equiv_t \Pi.(\phi \wedge \mathrm{OR}(\Pi, C, D, l))$?



- Read $\boxed{X} \rightarrow \boxed{Y}$ as "Y is more general than X".
- Interferences based on Y are more powerful than based on X.

$\psi := \Pi.(\phi \wedge C) \longrightarrow$ Inferences checks wrt. $\Pi.\phi$: Has C QRAT$^+$? $\underbrace{\longrightarrow \quad \Pi.\phi}_{\text{Deleting } C\text{: interference}}$

Summary

Future Work:

- Workflow for QRAT^+ proof checking and Skolem function extraction.
- Handling non-confluence of $\mathrm{QRAT}/\mathrm{QRAT}^+$ interferences in practice.
- Selective addition of redundancies (clauses, literals) to enable further interferences.

$\mathrm{QRATPre}_+$ tool: `https://lonsing.github.io/qratpreplus/`

Summary

Future Work:

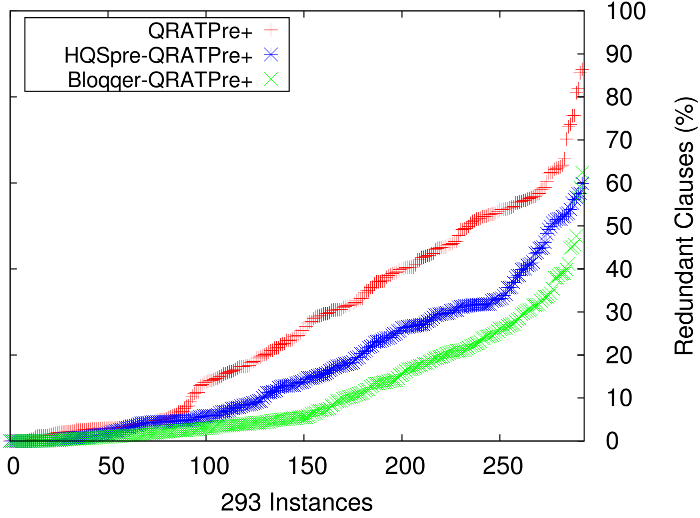
- Workflow for QRAT^+ proof checking and Skolem function extraction.
- Handling non-confluence of $\mathrm{QRAT}/\mathrm{QRAT}^+$ interferences in practice.
- Selective addition of redundancies (clauses, literals) to enable further interferences.

QRATPre+ tool: `https://lonsing.github.io/qratpreplus/`

Thank you!

Appendix

[Appendix] Experiments: Clause Elimination



- QRATPre+ detects redundant clauses even after the application of state-of-the-art preprocessors Bloqqer and HQSpre.

References

References I

[AB02] Abdelwaheb Ayari and David A. Basin.
QUBOS: Deciding Quantified Boolean Logic Using Propositional Satisfiability Solvers.
In *FMCAD*, volume 2517 of *LNCS*, pages 187–201. Springer, 2002.

[BCJ15] Olaf Beyersdorff, Leroy Chew, and Mikolás Janota.
Proof Complexity of Resolution-based QBF Calculi.
In *STACS*, volume 30 of *LIPIcs*, pages 76–89. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2015.

[Bie04] Armin Biere.
Resolve and Expand.
In *SAT*, volume 3542 of *LNCS*, pages 59–70. Springer, 2004.

References II

[BWJ14] Valeriy Balabanov, Magdalena Widl, and Jie-Hong R. Jiang.
QBF Resolution Systems and Their Proof Complexities.
In *SAT*, volume 8561 of *LNCS*, pages 154–169. Springer, 2014.

[CGS98] Marco Cadoli, Andrea Giovanardi, and Marco Schaerf.
An Algorithm to Evaluate Quantified Boolean Formulae.
In *AAAI*, pages 262–267. AAAI Press / The MIT Press, 1998.

[DLL62] Martin Davis, George Logemann, and Donald W. Loveland.
A Machine Program for Theorem-Proving.
Commun. ACM, 5(7):394–397, 1962.

References III

[GNT06] Enrico Giunchiglia, Massimo Narizzano, and Armando Tacchella.
Clause/Term Resolution and Learning in the Evaluation of Quantified Boolean Formulas.
JAIR, 26:371–416, 2006.

[HK17] Marijn Heule and Benjamin Kiesl.
The Potential of Interference-Based Proof Systems.
In *ARCADE*, volume 51 of *EPiC Series in Computing*, pages 51–54. EasyChair, 2017.

[HSB14] Marijn Heule, Martina Seidl, and Armin Biere.
A Unified Proof System for QBF Preprocessing.
In *IJCAR*, volume 8562 of *LNCS*, pages 91–106. Springer, 2014.

[HSB17] Marijn Heule, Martina Seidl, and Armin Biere.
Solution Validation and Extraction for QBF Preprocessing.
J. Autom. Reasoning, 58(1):97–125, 2017.

[JHB12] Matti Järvisalo, Marijn Heule, and Armin Biere.
Inprocessing Rules.
In *IJCAR*, volume 7364 of *LNCS*, pages 355–370. Springer, 2012.

[JKMSC16] Mikolás Janota, William Klieber, João Marques-Silva, and Edmund Clarke.
Solving QBF with Counterexample Guided Refinement.
Artif. Intell., 234:1–25, 2016.

[JM13] Mikolás Janota and João Marques-Silva.
On Propositional QBF Expansions and Q-Resolution.
In *SAT*, volume 7962 of *LNCS*, pages 67–82. Springer, 2013.

References V

[KBKF95] Hans Kleine Büning, Marek Karpinski, and Andreas Flögel.
Resolution for Quantified Boolean Formulas.
Inf. Comput., 117(1):12–18, 1995.

[KHS17] Benjamin Kiesl, Marijn Heule, and Martina Seidl.
A Little Blocked Literal Goes a Long Way.
In *SAT*, volume 10491 of *LNCS*, pages 281–297. Springer, 2017.

[KSTB17] Benjamin Kiesl, Martina Seidl, Hans Tompits, and Armin Biere.
Blockedness in Propositional Logic: Are You Satisfied With Your Neighborhood?
In *IJCAI*, pages 4884–4888. ijcai.org, 2017.

References VI

[Kul99] Oliver Kullmann.
On a Generalization of Extended Resolution.
Discrete Applied Mathematics, 96-97:149–176, 1999.

[LE18a] Florian Lonsing and Uwe Egly.
Evaluating QBF Solvers: Quantifier Alternations Matter.
In *Proc. of 24th International Conference on Principles and Practice of Constraint Programming (CP), August 27-31, 2018, Lille, France*. Springer LNCS, 2018.
To appear.

[LE18b] Florian Lonsing and Uwe Egly.
QRAT$^+$: Generalizing QRAT by a More Powerful QBF Redundancy Property.
In *IJCAR*, volume 10900 of *Lecture Notes in Computer Science*, pages 161–177. Springer, 2018.

[Let02] Reinhold Letz.
Lemma and Model Caching in Decision Procedures for Quantified Boolean Formulas.
In *TABLEAUX*, volume 2381 of *LNCS*, pages 160–175. Springer, 2002.

[RS16] Markus N. Rabe and Sanjit A. Seshia.
Incremental Determinization.
In *SAT*, volume 9710 of *LNCS*, pages 375–392. Springer, 2016.

[RT15] Markus N. Rabe and Leander Tentrup.
CAQE: A Certifying QBF Solver.
In *FMCAD*, pages 136–143. IEEE, 2015.

[RTRS18] Markus N. Rabe, Leander Tentrup, Cameron Rasmussen, and Sanjit A. Seshia.
Understanding and Extending Incremental Determinization for 2QBF.
In *CAV*, volume 10982 of *LNCS*, pages 256–274. Springer, 2018.

[SDB06] Horst Samulowitz, Jessica Davies, and Fahiem Bacchus.
Preprocessing QBF.
In *CP*, volume 4204 of *LNCS*, pages 514–529. Springer, 2006.

[VG12] Allen Van Gelder.
Contributions to the Theory of Practical Quantified Boolean Formula Solving.
In *CP*, volume 7514 of *LNCS*, pages 647–663. Springer, 2012.

References IX

[WHH14] Nathan Wetzler, Marijn Heule, and Warren A. Jr. Hunt.
DRAT-trim: Efficient Checking and Trimming Using Expressive Clausal Proofs.
In *SAT*, volume 8561 of *LNCS*, pages 422–429. Springer, 2014.

[ZM02a] Lintao Zhang and Sharad Malik.
Conflict Driven Learning in a Quantified Boolean Satisfiability Solver.
In *ICCAD*, pages 442–449. ACM / IEEE Computer Society, 2002.

[ZM02b] Lintao Zhang and Sharad Malik.
Towards a Symmetric Treatment of Satisfaction and Conflicts in Quantified Boolean Formula Evaluation.
In *CP*, volume 2470 of *LNCS*, pages 200–215. Springer, 2002.