# Reasoning with Quantified Boolean Formulas

**Marijn J.H. Heule**

Carnegie Mellon University

http://www.cs.cmu.edu/~mheule/15816-f24/

Automated Reasoning and Satisfiability

October 9, 2024

# What are QBF?

- **Quantified Boolean formulas (QBF)** are

  **formulas of propositional logic + quantifiers**

- *Examples*:
  - $(x \vee \overline{y}) \wedge (\overline{x} \vee y)$ (propositional logic)

# What are QBF?

- **Quantified Boolean formulas (QBF)** are

  **formulas of propositional logic + quantifiers**

- *Examples*:
  - $(x \lor \overline{y}) \land (\overline{x} \lor y)$ (propositional logic)
  - $\exists x \forall y (x \lor \overline{y}) \land (\overline{x} \lor y)$
    Is there a value for $x$ such that for all values of $y$ the formula is true?

# What are QBF?

- **Quantified Boolean formulas (QBF)** are

  **formulas of propositional logic + quantifiers**

- *Examples*:
  - $(x \vee \overline{y}) \wedge (\overline{x} \vee y)$ (propositional logic)
  - $\exists x \forall y (x \vee \overline{y}) \wedge (\overline{x} \vee y)$
    Is there a value for $x$ such that for all values of $y$ the formula is true?
  - $\forall y \exists x (x \vee \overline{y}) \wedge (\overline{x} \vee y)$
    For all values of $y$, is there a value for $x$ such that the formula is true?

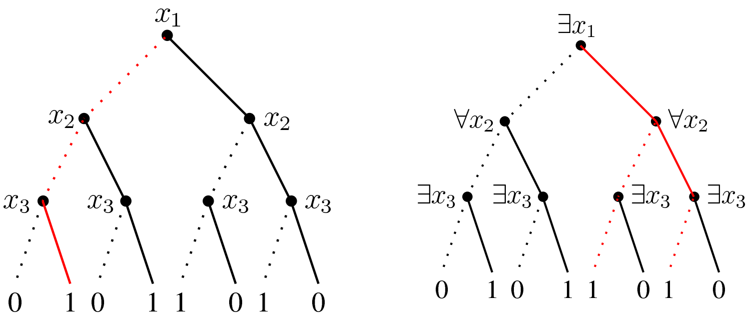
# SAT vs. QSAT aka NP-complete vs. PSPACE-complete

SAT

$\phi(x_1, x_2, x_3)$

QBF

$\exists x_1 \forall x_2 \exists x_3 \phi(x_1, x_2, x_3)$




Is there a satisfying assignment?

Is there a satisfying assignment **tree**?

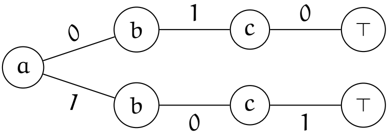
# Small Example QSAT Problems

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

# Small Example QSAT Problems

Consider the formula $\forall a\, \exists b, c.(a \lor b) \land (\overline{a} \lor c) \land (\overline{b} \lor \overline{c})$

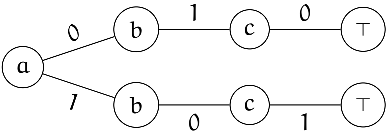
A model is:



# Small Example QSAT Problems

Consider the formula $\forall a\, \exists b, c. (a \lor b) \land (\overline{a} \lor c) \land (\overline{b} \lor \overline{c})$

A model is:

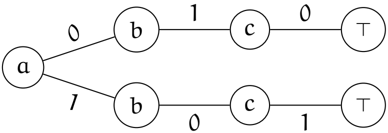



Consider the formula $\exists b\, \forall a\, \exists c. (a \lor b) \land (\overline{a} \lor c) \land (\overline{b} \lor \overline{c})$

# Small Example QSAT Problems

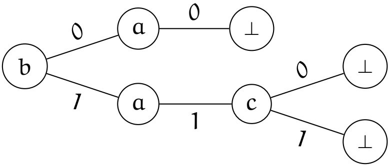
Consider the formula $\forall a \, \exists b, c . (a \vee b) \wedge (\overline{a} \vee c) \wedge (\overline{b} \vee \overline{c})$

A model is:




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

A counter-model is:




The quantifier prefix frequently determines the truth of a QBF.

# The Two Player Game Interpretation of QSAT

Interpretation of QSAT as *two player game* for a QBF $\exists x_1 \forall a_1 \exists x_2 \forall a_2 \cdots \exists x_n \forall a_n \psi$:

- Player A (existential player) tries to satisfy the formula by assigning existential variables
- Player B (universal player) tries to falsify the formula by assigning universal variables

- Player A and Player B make alternately an assignment of the variables in the outermost quantifier block
- Player A wins: formula is satisfiable, i.e., there is a strategy for assigning the existential variables such that the formula is always satisfied
- Player B wins: formula is unsatisfiable

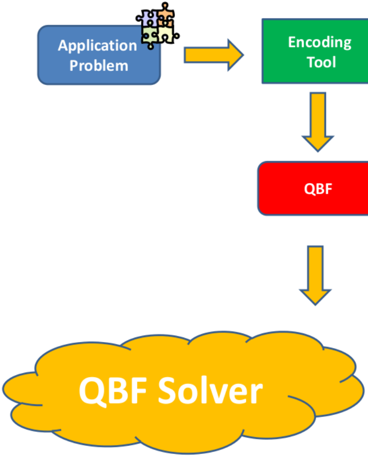
# Promises of QBF

- QSAT is the prototypical problem for *PSPACE*.

- QBFs are suitable as *host language* for the encoding of many application problems like
  - verification
  - artificial intelligence
  - knowledge representation
  - game solving

- In general, QBF allow more succinct encodings then SAT

# Application of a QBF Solver




QBF Solver returns

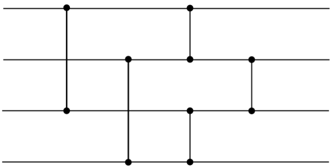
1. yes/no
2. witnesses

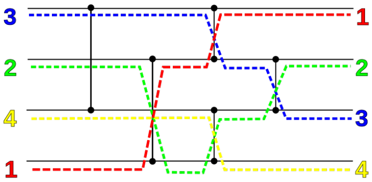
# Example of ∃∀∃: Synthesis

Given an input-output specification, does there exists a circuit that satisfies the input-output specification.

QBF solving can be used to find the smallest sorting network:

- (∃) Does there exists a sorting network of $k$ wires,
- (∀) such that for all input variables of the network
- (∃) the output $O_i \leq O_{i+1}$

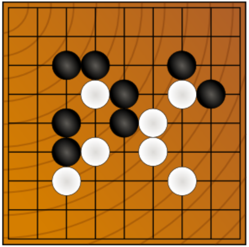




# Example of $\forall\exists\ldots\forall\exists$: Games

Many games, such as Go and Reversi, can be naturally expressed as a QBF problem.

Boolean variables $a_{i,k}$, $b_{j,k}$ express that the existential player places a piece on row $i$ and column $j$ at his $k$th turn. Variables $c_{i,k}$, $d_{j,k}$ are used for the universal player.



Go

Reversi

The QBF problem is of the form

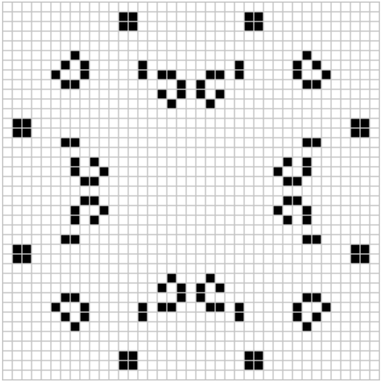
$$\forall c_{i,1}, d_{j,1} \exists a_{i,1}, b_{j,1} \ldots \forall c_{i,n}, d_{j,n} \exists a_{i,n}, b_{j,n} . \psi$$

Outcome "satisfiable": the second player (existential) can always prevent that the first player (universal) wins.

# Illustrating Example ∀∃: Conway's Game of Life

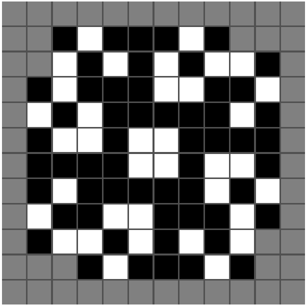
Conway's Game of Life is an infinite 2D grid of cells that are either alive or dead using the following update rules:

- Any alive cell with fewer than two alive neighbors dies;
- Any alive cell with two or three live neighbors lives;
- Any alive cell with more than three alive neighbors dies;
- Any dead cell with exactly three alive neighbors becomes alive.



Game of Life is very popular: over 1,100 wiki articles

# Garden of Eden in Conway's Game of Life



A Garden of Eden (GoE) is a state that can only exist as initial state.

Let $T(x, y)$ denote the CNF formula that encodes the transition relation from a state to its successor using variables $x$ that describe the current state and variables $y$ the successor state.

A QBF that encodes the GoE problem is simply

$$\forall y \exists x . T(x, y)$$

The smallest Garden of Eden known so far (shown above) was found using a QBF solver. [Hartman et al. 2013]

# The Language of QBF

The language of **quantified Boolean formulas** $\mathcal{L}_{\mathcal{P}}$ over a set of propositional variables $\mathcal{P}$ is the smallest set such that

- if $v \in \mathcal{P} \cup \{\top, \bot\}$ then $v \in \mathcal{L}_{\mathcal{P}}$ (variables, constants)
- if $\phi \in \mathcal{L}_{\mathcal{P}}$ then $\overline{\phi} \in \mathcal{L}_{\mathcal{P}}$ (negation)
- if $\phi$ and $\psi \in \mathcal{L}_{\mathcal{P}}$ then $\phi \wedge \psi \in \mathcal{L}_{\mathcal{P}}$ (conjunction)
- if $\phi$ and $\psi \in \mathcal{L}_{\mathcal{P}}$ then $\phi \vee \psi \in \mathcal{L}_{\mathcal{P}}$ (disjunction)
- if $\phi \in \mathcal{L}_{\mathcal{P}}$ then $\exists v \phi \in \mathcal{L}_{\mathcal{P}}$ (existential quantifier)
- if $\phi \in \mathcal{L}_{\mathcal{P}}$ then $\forall v \phi \in \mathcal{L}_{\mathcal{P}}$ (universal quantifier)

# Some Notes on Variables and Truth Constants

- $\top$ stands for *top*
  - always true
  - empty conjunction
- $\bot$ stands for *bottom*
  - always false
  - empty disjunction
- *literal*: variable or negation of a variable
  - examples: $l_1 = v$, $l_2 = \overline{w}$
  - var($l$) $= v$ if $l = v$ or $l = \overline{v}$
  - complement of literal $l$: $\overline{l}$
- $var(\phi)$: set of variables occurring in QBF $\phi$

# Some QBF Terminology

Let $Qv\psi$ with $Q \in \{\forall, \exists\}$ be a subformula in a QBF $\phi$, then

- $\psi$ is the *scope* of $v$
- $Q$ is the *quantifier binding* of $v$
- *quant(v) = Q*
- *free variable* $w$ in $\phi$: $w$ has no quantifier binding in $\phi$
- *bound variable* $w$ in QBF $\phi$: $w$ has quantifier binding in $\phi$
- *closed QBF*: no free variables

## Example

$$\overbrace{\forall a}^{\textit{bound var } a} \; \underbrace{(a \wedge \overbrace{x}^{\textit{free variable}} \vee \overbrace{\overbrace{\forall y \exists z}^{\textit{bound vars } y,\, z} \; \overbrace{((y \vee \bar{z}) \wedge (\bar{y} \vee z))}^{\textit{scope of } y,\, z}}^{\text{closed QBF}})}_{\textit{scope of } a}$$

# Prenex Conjunctive Normal Form (PCNF)

A QBF $\phi$ is in **prenex conjunctive normal form** iff

- $\phi$ is in *prenex normal form* $\phi = Q_1 v_1 \dots Q_n v_n \psi$
- matrix $\psi$ is in *conjunctive normal form*, i.e.,

$$\psi = C_1 \wedge \cdots \wedge C_m$$

where $C_i$ are clauses, i.e., disjunctions of literals.

Example

$$\underbrace{\forall x \exists y}_{\textit{prefix}} \underbrace{((x \vee \overline{y}) \wedge (\overline{x} \vee y))}_{\textit{matrix in CNF}}$$

# Some Words on Notation

If convenient, we write

- a conjunction of clauses as a set, i.e.,

$$C_1 \land \ldots \land C_m = \{C_1, \ldots, C_m\}$$

- a clause as a set of literals, i.e.,

$$l_1 \lor \ldots \lor l_k = \{l_1, \ldots, l_k\}$$

- $var(\phi)$ for the variables occurring in $\phi$
- $var(l)$ for the variable of a literal, i.e.,

$$var(l) = x \text{ iff } l = x \text{ or } l = \bar{x}$$

Example

$$\underbrace{\forall x \exists y}_{\textit{prefix}} \underbrace{((x \lor \bar{y}) \land (\bar{x} \lor y))}_{\textit{matrix in CNF}} \approx \underbrace{\forall x \exists y}_{\textit{prefix}} \underbrace{\{\{x, \bar{y}\}, \{\bar{x}, y\}\}}_{\textit{matrix in CNF}}$$

# Semantics of QBFs

A **valuation function** $\mathcal{I}$: $\mathcal{L}_{\mathcal{P}} \rightarrow \{\mathcal{T}, \mathcal{F}\}$ for closed QBFs is defined as follows:

- $\mathcal{I}(\top) = \mathcal{T}$; $\mathcal{I}(\bot) = \mathcal{F}$
- $\mathcal{I}(\overline{\psi}) = \mathcal{T}$ iff $\mathcal{I}(\psi) = \mathcal{F}$
- $\mathcal{I}(\phi \vee \psi) = \mathcal{T}$ iff $\mathcal{I}(\phi) = \mathcal{T}$ or $\mathcal{I}(\psi) = \mathcal{T}$
- $\mathcal{I}(\phi \wedge \psi) = \mathcal{T}$ iff $\mathcal{I}(\phi) = \mathcal{T}$ and $\mathcal{I}(\psi) = \mathcal{T}$
- $\mathcal{I}(\forall v.\psi) = \mathcal{T}$ iff $\mathcal{I}(\psi[\bot/v]) = \mathcal{T}$ and $\mathcal{I}(\psi[\top/v]) = \mathcal{T}$
- $\mathcal{I}(\exists v.\psi) = \mathcal{T}$ iff $\mathcal{I}(\psi[\bot/v]) = \mathcal{T}$ or $\mathcal{I}(\psi[\top/v]) = \mathcal{T}$

```
Boolean split (QBF φ)

switch(φ)
  case ⊤: return true;
  case ⊥: return false;
  case ψ̄: return (not split(ψ));
  case ψ′∧ψ″: return split(ψ′) && split(ψ″);
  case ψ′∨ψ″: return split(ψ′) || split(ψ″);
  case QXψ:
    select x ∈ X; X′ = X\{x};
    if (Q == ∀)
      return (split(QX′ψ[⊤/x]) &&
              split(QX′ψ[⊥/x]));
    else
      return (split(QX′ψ[⊤/x]) ||
              split(QX′ψ[⊥/x]));
```

# Some Simplifications

The following rewritings are *equivalence preserving*:

1. $\overline{\top} \Rightarrow \bot$; $\overline{\bot} \Rightarrow \top$;
2. $\top \wedge \phi \Rightarrow \phi$; $\bot \wedge \phi \Rightarrow \bot$; $\top \vee \phi \Rightarrow \top$; $\bot \vee \phi \Rightarrow \phi$;
3. $(Qx\,\phi) \Rightarrow \phi$, $Q \in \{\forall, \exists\}$, $x$ does not occur in $\phi$;

Example

$$\forall ab\exists x\forall c\exists yz\forall d\{\{a, b, \bar{c}\}, \{a, \bar{b}, \overline{\top}\}, \{c, y, d, \bot\}, \{x, y, \overline{\bot}\}, \{x, c, d, \top\}\}$$

$$\approx$$

$$\forall abc\exists y\forall d\{\{a, b, \bar{c}\}, \{a, \bar{b}\}, \{c, y, d\}\}$$

```
Boolean splitCNF (Prefix P, matrix ψ)

if (ψ == ⊤): return true;
if (⊥ ∈ ψ): return false;

P = QXP′, x ∈ X,  X′ = X\{x};

if (Q == ∀)
    return (splitCNF(QX′P′, ψ′) &&
            splitCNF(QX′P′, ψ″));
else
    return (splitCNF(QX′P′, ψ′) ||
            splitCNF(QX′P′, ψ″));
where
ψ′ : take clauses of ψ, delete clauses with x, delete literals x̄
ψ″ : take clauses of ψ, delete clauses with x̄, delete literals x
```

# Unit Clauses

A clause $C$ is called **unit** in a formula $\phi$ iff

- $C$ contains exactly one existential literal
- the universal literals of $C$ are to the right of the existential literal in the prefix

The existential literal in the unit clause is called *unit literal*.

Example

$$\forall ab \exists x \forall c \exists y \forall d \{\{a, b, \bar{x}, \bar{c}\}, \{a, \bar{b}\}, \{c, y, d\}, \{x, y\}, \{x, c, d\}, \{y\}\}$$

# Unit Clauses

A clause $C$ is called **unit** in a formula $\phi$ iff

- $C$ contains exactly one existential literal
- the universal literals of $C$ are to the right of the existential literal in the prefix

The existential literal in the unit clause is called *unit literal*.

Example

$$\forall ab \exists x \forall c \exists y \forall d \{\{a, b, \bar{x}, \bar{c}\}, \{a, \bar{b}\}, \{c, y, d\}, \{x, y\}, \{x, c, d\}, \{y\}\}$$

*Unit literals:* $x$, $y$

# Unit Literal Elimination

Let $\phi$ be a QBF with unit literal $l$ and let $\phi'$ be a QBF obtained from $\phi$ by

- removing all clauses containing $l$
- removing all occurrences of $\bar{l}$

Then

$$\phi \approx \phi'$$

Example

$\forall ab\exists x\forall c\exists y\forall d\{\{a, b, \bar{x}, \bar{c}\}, \{a, \bar{b}\}, \{c, y, d\}, \{x, y\}, \{x, c, d\}, \{y\}\}$

*After unit literal elimiation:* $\forall abc\{\{a, b, \bar{c}\}, \{a, \bar{b}\}\}$

# Pure Literals

A literal $l$ is called **pure** in a formula $\phi$ iff

- $l$ occurs in $\phi$
- the complement of $l$, i.e., $\bar{l}$ does not occur in $\phi$

Example

$$\forall ab \exists x \forall c \exists yz \forall d \{\{a, b, \overline{c}\}, \{a, \overline{b}\}, \{c, y, d\}, \{x, y\}, \{x, c, d\}\}$$

# Pure Literals

A literal $l$ is called **pure** in a formula $\phi$ iff

- $l$ occurs in $\phi$
- the complement of $l$, i.e., $\bar{l}$ does not occur in $\phi$

Example

$$\forall ab \exists x \forall c \exists yz \forall d \{\{a, b, \overline{c}\}, \{a, \overline{b}\}, \{c, y, d\}, \{x, y\}, \{x, c, d\}\}$$

*Pure:* $a, d, x, y$

# Pure Literal Elimination

Let $\phi$ be a QBF with pure literal $l$ and let $\phi'$ be a QBF obtained from $\phi$ by

- removing all clauses with $l$ if $\text{quant}(l) = \exists$
- removing all occurrences of $l$ if $\text{quant}(l) = \forall$

Then

$$\phi \approx \phi'$$

Example

$$\forall ab \exists x \forall c \exists yz \forall d \{\{a, b, \bar{c}\}, \{a, \bar{b}\}, \{c, y, d\}, \{x, y\}, \{x, c, d\}\}$$

*After Pure Literal Elimination:* $\forall b \{\{b\}, \{\bar{b}\}\}$

# Universal Reduction (UR)

- Let $\Pi.\psi$ be a QBF in PCNF and $C \in \psi$.
- Let $l \in C$ with
  - $\mathrm{quant}(l) = \forall$
  - forall $k \in C$ with $\mathrm{quant}(k) = \exists$ $k <_{\Pi} l$, i.e., all existential variables $k$ of $C$ are to the left of $l$ in $\Pi$.
- Then $l$ may be removed from $C$.
- $C \backslash \{l\}$ is called the *universal reduct* of $C$.

Example

$$\forall ab \exists x \forall c \exists yz \forall d \{\{a, b, x, \bar{c}\}, \{a, \bar{b}, x\}, \{c, y, d\}, \{x, y\}, \{x, c, d\}\}$$

# Universal Reduction (UR)

- Let $\Pi.\psi$ be a QBF in PCNF and $C \in \psi$.
- Let $l \in C$ with
  - $\text{quant}(l) = \forall$
  - forall $k \in C$ with $\text{quant}(k) = \exists$ $k <_\Pi l$, i.e., all existential variables $k$ of $C$ are to the left of $l$ in $\Pi$.
- Then $l$ may be removed from $C$.
- $C \backslash \{l\}$ is called the *universal reduct* of $C$.

## Example

$$\forall ab \exists x \forall c \exists yz \forall d \{\{a, b, x, \bar{c}\}, \{a, \bar{b}, x\}, \{c, y, d\}, \{x, y\}, \{x, c, d\}\}$$

*After Universal Reduction:*

$$\forall ab \exists x \forall c \exists yz \forall d \{\{a, b, x\}, \{a, \bar{b}, x\}, \{c, y\}, \{x, y\}, \{x\}\}$$

```
Boolean splitCNF2 (Prefix P, matrix ψ)

(P,ψ) = simplify(P,ψ);

if (ψ == ⊥): return true;
if (⊥ ∈ ψ): return false;

P = QXP', x ∈ X,  X' = X\{x};

if (Q == ∀)
    return (splitCNF2(QX'P',ψ') &&
            splitCNF2(QX'P',ψ''));
else
    return (splitCNF2(QX'P',ψ') ||
            splitCNF2(QX'P',ψ''));
where
ψ' : take clauses of ψ, delete clauses with x, delete x̄
ψ'' : take clauses of ψ, delete clauses with x̄, delete x
```

# Resolution for QBF

**Q-Resolution:** propositional resolution + universal reduction.

Definition

Let $C_1, C_2$ be clauses with existential literal $l \in C_1$ and $\bar{l} \in C_2$.

1. Tentative Q-resolvent:
   $C_1 \bowtie C_2 := (UR(C_1) \cup UR(C_2)) \setminus \{l, \bar{l}\}$.
2. If $\{x, \overline{x}\} \subseteq C_1 \bowtie C_2$ then no Q-resolvent exists.
3. Otherwise, Q-resolvent $C := (C_1 \bowtie C_2)$.

- Q-resolution is a sound and complete calculus.
- Universals as pivot are also possible.

# Q-Resolution Small Example

**Exclusive OR (XOR):** QBF $\psi = \exists x \forall y (x \vee y) \wedge (\overline{x} \vee \overline{y})$

# Q-Resolution Small Example

**Exclusive OR (XOR):** QBF $\psi = \exists x \forall y (x \vee y) \wedge (\overline{x} \vee \overline{y})$

**Truth Table**

| $x$ | $y$ | $\psi$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

unsat

# Q-Resolution Small Example

**Exclusive OR (XOR):** QBF $\psi = \exists x \forall y (x \lor y) \land (\bar{x} \lor \bar{y})$

**Q-Resolution Proof**

$$
\begin{tikzcd}
x \lor y \arrow[d] & & \bar{x} \lor \bar{y} \arrow[d] \\
x \arrow[dr] & & \bar{x} \arrow[dl] \\
& \bot &
\end{tikzcd}
$$

# Q-Resolution Small Example

**Exclusive OR (XOR):** QBF $\psi = \exists x \forall y (x \lor y) \land (\bar{x} \lor \bar{y})$

**Q-Resolution Proof**

$$\begin{tikzcd} \text{Universal-Reduction} \arrow[r] & x \lor \boxed{y} \arrow[d] & & \bar{x} \lor \boxed{\bar{y}} \arrow[d] \\ & x \arrow[dr] & & \bar{x} \arrow[dl] \\ & & \bot & \end{tikzcd}$$

# Q-Resolution Small Example

**Exclusive OR (XOR):** QBF $\psi = \exists x \forall y (x \lor y) \land (\bar{x} \lor \bar{y})$

**Q-Resolution Proof**

$$\begin{tikzcd}
\text{Universal-Reduction} \arrow[r] & x \lor \boxed{y} \arrow[d] & & \bar{x} \lor \boxed{\bar{y}} \arrow[d] \\
\text{Resolution} \arrow[r] & \boxed{x} \arrow[dr] & & \boxed{\bar{x}} \arrow[dl] \\
& & \bot &
\end{tikzcd}$$

# Q-Resolution Small Example

**Exclusive OR (XOR):** QBF $\psi = \exists x \forall y (x \lor y) \land (\bar{x} \lor \bar{y})$

**Truth Table**

| $x$ | $y$ | $\psi$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

unsat

**Q-Resolution Proof**

$$
\begin{tikzcd}
x \lor y \arrow[d] & & \bar{x} \lor \bar{y} \arrow[d] \\
x \arrow[dr] & & \bar{x} \arrow[dl] \\
& \bot &
\end{tikzcd}
$$

$$\longrightarrow \quad y = x \;\Rightarrow\; \psi = 0$$

# Q-Resolution Small Example

**Exclusive OR (XOR):** QBF $\psi = \exists x \forall y (x \lor y) \land (\overline{x} \lor \overline{y})$

**Truth Table**

| $x$ | $y$ | $\psi$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

unsat

**Q-Resolution Proof**

$$\begin{tikzcd}
x \lor y \arrow[d] & & \overline{x} \lor \overline{y} \arrow[d] \\
x \arrow[dr] & & \overline{x} \arrow[dl] \\
& \perp &
\end{tikzcd}$$

$$\longrightarrow \quad y = x \;\Rightarrow\; \psi = 0$$

$$\longrightarrow \quad f_y(x) = x \quad \text{(counter model)}$$

# Q-Resolution Large Example

**Input Formula**

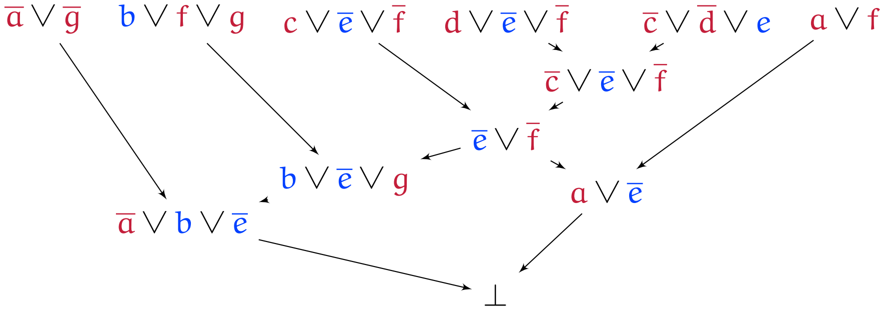
$$\exists a \forall b \exists c d \forall e \exists f g . (\overline{a} \lor \overline{g}) \land (b \lor f \lor g) \land (c \lor \overline{e} \lor \overline{f}) \land (d \lor \overline{e} \lor \overline{f}) \land (\overline{c} \lor \overline{d} \lor e) \land (a \lor f)$$

# Q-Resolution Large Example

### Input Formula

$$\exists a \forall b \exists c d \forall e \exists f g.(\overline{a} \lor \overline{g}) \land (b \lor f \lor g) \land (c \lor \overline{e} \lor \overline{f}) \land (d \lor \overline{e} \lor \overline{f}) \land (\overline{c} \lor \overline{d} \lor e) \land (a \lor f)$$

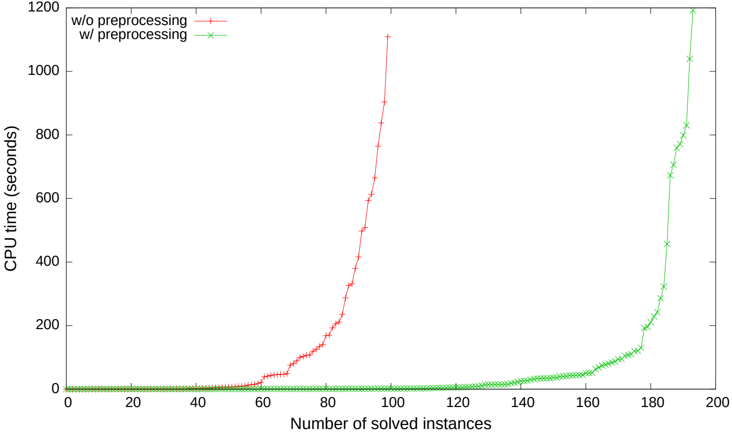
### Q-Resolution Proof DAG



# QBF Preprocessing

Preprocessing is crucial to solve most QBF instances efficiently.

Results of DepQBF w/ and w/o bloqqer on QBF Eval 2012 [1]



# Quantified Blocked Clause

**Definition (Quantified Blocking literal)**

An existential literal $l$ in a clause $C$ of a QBF $\Pi.\varphi$ blocks $C$ with respect to $\Pi.\varphi$ if for every clause $D \in F_{\bar{l}}$, there exists a literal $k \neq l$ with $k \leq_{\Pi} l$ such that $k \in C$ and $\bar{k} \in D$.

**Definition (Quantified Blocked clause)**

A clause is blocked if it contains a literal that blocks it.

# Quantified Blocked Clause

**Definition (Quantified Blocking literal)**

An existential literal $l$ in a clause $C$ of a QBF $\Pi.\varphi$ blocks $C$ with respect to $\Pi.\varphi$ if for every clause $D \in F_{\bar{l}}$, there exists a literal $k \neq l$ with $k \leq_{\Pi} l$ such that $k \in C$ and $\bar{k} \in D$.

**Definition (Quantified Blocked clause)**

A clause is blocked if it contains a literal that blocks it.

**Example**

$$\exists a \forall bcd \exists ef \forall g.(\bar{a} \vee \bar{g}) \wedge (b \vee f \vee g) \wedge (c \vee \bar{e} \vee \bar{f}) \wedge (d \vee \bar{e} \vee \bar{f}) \wedge (\bar{c} \vee \bar{d} \vee e) \wedge (a \vee f)$$

# Quantified Blocked Clause

**Definition (Quantified Blocking literal)**

An existential literal $l$ in a clause $C$ of a QBF $\Pi.\varphi$ blocks $C$ with respect to $\Pi.\varphi$ if for every clause $D \in F_{\bar{l}}$, there exists a literal $k \neq l$ with $k \leq_{\Pi} l$ such that $k \in C$ and $\bar{k} \in D$.

**Definition (Quantified Blocked clause)**

A clause is blocked if it contains a literal that blocks it.

**Example**

$$\exists a \forall bcd \exists ef \forall g.(\bar{a} \vee \bar{g}) \wedge (b \vee f \vee g) \wedge \cancel{(c \vee \bar{e} \vee \bar{f})} \wedge (d \vee \bar{e} \vee \bar{f}) \wedge (\bar{c} \vee \bar{d} \vee e) \wedge (a \vee f)$$

# Quantified Blocked Clause

### Definition (Quantified Blocking literal)

An existential literal $l$ in a clause $C$ of a QBF $\Pi.\varphi$ blocks $C$ with respect to $\Pi.\varphi$ if for every clause $D \in F_{\bar{l}}$, there exists a literal $k \neq l$ with $k \leq_{\Pi} l$ such that $k \in C$ and $\bar{k} \in D$.

### Definition (Quantified Blocked clause)

A clause is blocked if it contains a literal that blocks it.

### Example

$$\exists a \forall bcd \exists ef \forall g.(\bar{a} \vee \bar{g}) \wedge (b \vee f \vee g) \wedge \cancel{(c \vee \bar{e} \vee \bar{f})} \wedge \cancel{(d \vee \bar{e} \vee \bar{f})} \wedge (\bar{c} \vee \bar{d} \vee e) \wedge (a \vee f)$$

# Quantified Blocked Clause

**Definition (Quantified Blocking literal)**

An existential literal $l$ in a clause $C$ of a QBF $\Pi.\varphi$ blocks $C$ with respect to $\Pi.\varphi$ if for every clause $D \in F_{\bar{l}}$, there exists a literal $k \neq l$ with $k \leq_{\Pi} l$ such that $k \in C$ and $\bar{k} \in D$.

**Definition (Quantified Blocked clause)**

A clause is blocked if it contains a literal that blocks it.

**Example**

$$\exists a \forall bcd \exists ef \forall g. (\bar{a} \lor \bar{g}) \land (b \lor f \lor g) \land \cancel{(c \lor \bar{e} \lor \bar{f})} \land \cancel{(d \lor \bar{e} \lor \bar{f})} \land \cancel{(\bar{c} \lor \bar{d} \lor e)} \land (a \lor f)$$

# Quantified Blocked Clause

**Definition (Quantified Blocking literal)**

An existential literal $l$ in a clause $C$ of a QBF $\Pi.\varphi$ blocks $C$ with respect to $\Pi.\varphi$ if for every clause $D \in F_{\bar{l}}$, there exists a literal $k \neq l$ with $k \leq_{\Pi} l$ such that $k \in C$ and $\bar{k} \in D$.

**Definition (Quantified Blocked clause)**

A clause is blocked if it contains a literal that blocks it.

**Example**

$$\exists a \forall bcd \exists ef \forall g.(\bar{a} \vee \bar{g}) \wedge \cancel{(b \vee f \vee g)} \wedge \cancel{(c \vee \bar{e} \vee \bar{f})} \wedge \cancel{(d \vee \bar{e} \vee \bar{f})} \wedge \cancel{(\bar{c} \vee \bar{d} \vee e)} \wedge (a \vee f)$$

# Quantified Blocked Clause

**Definition (Quantified Blocking literal)**

An existential literal $l$ in a clause $C$ of a QBF $\Pi.\varphi$ blocks $C$ with respect to $\Pi.\varphi$ if for every clause $D \in F_{\bar{l}}$, there exists a literal $k \neq l$ with $k \leq_{\Pi} l$ such that $k \in C$ and $\bar{k} \in D$.

**Definition (Quantified Blocked clause)**

A clause is blocked if it contains a literal that blocks it.

**Example**

$$\exists a \forall bcd \exists ef \forall g.(\bar{a} \vee \bar{g}) \wedge \cancel{(b \vee f \vee g)} \wedge \cancel{(c \vee \bar{e} \vee \bar{f})} \wedge \cancel{(d \vee \bar{e} \vee \bar{f})} \wedge \cancel{(\bar{c} \vee \bar{d} \vee e)} \wedge \cancel{(a \vee f)}$$

# Quantified Blocked Clause

**Definition (Quantified Blocking literal)**

An existential literal $l$ in a clause $C$ of a QBF $\Pi.\varphi$ blocks $C$ with respect to $\Pi.\varphi$ if for every clause $D \in F_{\bar{l}}$, there exists a literal $k \neq l$ with $k \leq_{\Pi} l$ such that $k \in C$ and $\bar{k} \in D$.

**Definition (Quantified Blocked clause)**

A clause is blocked if it contains a literal that blocks it.

**Example**

$$\exists a \forall bcd \exists ef \forall g . \cancel{(\bar{a} \vee \bar{g})} \wedge \cancel{(b \vee f \vee g)} \wedge \cancel{(c \vee \bar{e} \vee \bar{f})} \wedge \cancel{(d \vee \bar{e} \vee \bar{f})} \wedge \cancel{(\bar{c} \vee \bar{d} \vee e)} \wedge \cancel{(a \vee f)}$$