

# Blocked Literals are Universal

Marijn J.H. Heule



*Joint work with*

Martina Seidl and Armin Biere



JOHANNES KEPLER  
UNIVERSITY LINZ | JKU

NASA Formal Methods, April 29, 2015

Introduction to QBF

Blocked Literal Elimination

Experimental Results

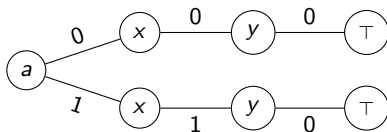
Conclusions

## Introduction to QBF

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

Consider  $\forall a \exists x \forall b \exists y. (a \vee \neg x \vee y) \wedge (\neg a \vee x \vee y) \wedge (b \vee \neg y)$

A **model** is:

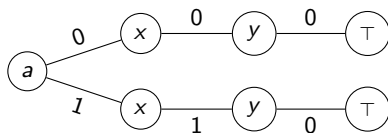


## Introduction to QBF

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

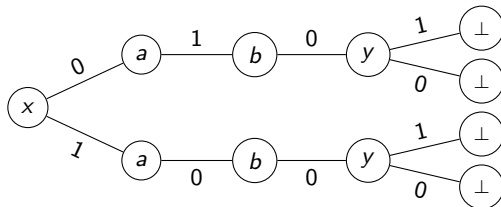
Consider  $\forall a \exists x \forall b \exists y. (a \vee \neg x \vee y) \wedge (\neg a \vee x \vee y) \wedge (b \vee \neg y)$

A **model** is:



Consider  $\exists x \forall a, b \exists y. (\neg x \vee a \vee y) \wedge (x \vee \neg a \vee y) \wedge (b \vee \neg y)$

A **counter-model** is:



# 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
  - ▶ synthesis
  - ▶ artificial intelligence
  - ▶ knowledge representation
  - ▶ game solving
- ▶ In general, QBF allow more succinct encodings than SAT

## Introduction to QBF Preprocessing

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

Consider  $\forall a \exists x \forall b \exists y. (a \vee \neg x \vee y) \wedge (\neg a \vee x \vee y) \wedge (b \vee \neg y)$

Existing QBF preprocessing techniques can **eliminated all clauses** in the above formula making it trivially satisfiable.

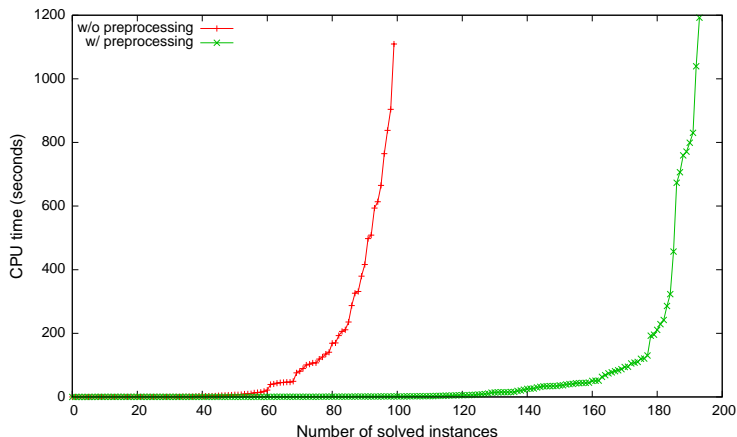
Consider  $\exists x \forall a, b \exists y. (\neg x \vee a \vee y) \wedge (x \vee \neg a \vee y) \wedge (b \vee \neg y)$

Our new QBF preprocessing technique can **eliminated all universal literals**, thereby reducing the problem to SAT.

# QBF Preprocessing

Preprocessing is **crucial** to solve most QBF instances efficiently.

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



# 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)

Here we show present a new preprocessing technique called **Blocked Literal Elimination** that follows from [1].



# Blocked Literal Elimination

## Quantified Blocked Clauses [BiereLonsingSeidl 2011]

### Definition (Quantified Blocking Literal)

An **existential literal**  $l$  in a clause  $C$  of a QBF  $\pi.\varphi$  blocks  $C$  w.r.t.  $\pi.\varphi$  if **for every clause**  $D \in \varphi$  with  $\neg l \in D$  holds that there exists  $k$  s.t.  $k \in C$ ,  $\neg k \in D$ ,  $l \neq k$  and  $k \leq_{\pi} l$ .

With respect to a fixed QBF  $\pi.\varphi$  and its clauses we have:

### Definition (Quantified Blocked Clause)

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

## Quantified Blocked Clauses [BiereLonsingSeidl 2011]

### Definition (Quantified Blocking Literal)

An **existential literal**  $l$  in a clause  $C$  of a QBF  $\pi.\varphi$  blocks  $C$  w.r.t.  $\pi.\varphi$  if **for every clause**  $D \in \varphi$  with  $\neg l \in D$  holds that there exists  $k$  s.t.  $k \in C$ ,  $\neg k \in D$ ,  $l \neq k$  and  $k \leq_{\pi} l$ .

With respect to a fixed QBF  $\pi.\varphi$  and its clauses we have:

### Definition (Quantified Blocked Clause)

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

### Example

$$\forall a \exists x \forall b \exists y. (a \vee \neg x \vee y) \wedge (\neg a \vee x \vee y) \wedge (b \vee \neg y)$$

## Quantified Blocked Clauses [BiereLonsingSeidl 2011]

### Definition (Quantified Blocking Literal)

An **existential literal**  $l$  in a clause  $C$  of a QBF  $\pi.\varphi$  blocks  $C$  w.r.t.  $\pi.\varphi$  if **for every clause**  $D \in \varphi$  with  $\neg l \in D$  holds that there exists  $k$  s.t.  $k \in C$ ,  $\neg k \in D$ ,  $l \neq k$  and  $k \leq_{\pi} l$ .

With respect to a fixed QBF  $\pi.\varphi$  and its clauses we have:

### Definition (Quantified Blocked Clause)

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

### Example

$$\forall a \exists x \forall b \exists y. (a \vee \neg x \vee y) \wedge (\neg a \vee x \vee y) \wedge (b \vee \neg y)$$

### Proposition 1

Removal of quantified blocked clauses preserves **unsatisfiability**.

## Blocked Literals [This paper]

### Definition (Blocked Literal)

A **universal literal**  $l$  in a clause  $C$  of a QBF  $\pi.\varphi$  blocks  $C$  w.r.t.  $\pi.\varphi$  if **for every clause**  $D \in \varphi$  with  $\neg l \in D$  holds that there exists  $k$  s.t.  $k \in C$ ,  $\neg k \in D$ ,  $l \neq k$  and  $k \leq_{\pi} l$ .

## Blocked Literals [This paper]

### Definition (Blocked Literal)

A **universal literal**  $l$  in a clause  $C$  of a QBF  $\pi.\varphi$  blocks  $C$  w.r.t.  $\pi.\varphi$  if **for every clause**  $D \in \varphi$  with  $\neg l \in D$  holds that there exists  $k$  s.t.  $k \in C$ ,  $\neg k \in D$ ,  $l \neq k$  and  $k \leq_{\pi} l$ .

### Example

$$\exists x \forall a, b \exists y. (\neg x \vee a \vee y) \wedge (x \vee \neg a \vee y) \wedge (b \vee \neg y)$$

## Blocked Literals [This paper]

### Definition (Blocked Literal)

A **universal literal**  $l$  in a clause  $C$  of a QBF  $\pi.\varphi$  blocks  $C$  w.r.t.  $\pi.\varphi$  if **for every clause**  $D \in \varphi$  with  $\neg l \in D$  holds that there exists  $k$  s.t.  $k \in C$ ,  $\neg k \in D$ ,  $l \neq k$  and  $k \leq_{\pi} l$ .

### Example

$$\exists x \forall a, b \exists y. (\neg x \vee a \vee y) \wedge (x \vee \neg a \vee y) \wedge (b \vee \neg y)$$

### Proposition 2

Removal of a quantified blocked literal preserves **satisfiability**.

## Blocked Literals [This paper]

### Definition (Blocked Literal)

A **universal literal**  $l$  in a clause  $C$  of a QBF  $\pi.\varphi$  blocks  $C$  w.r.t.  $\pi.\varphi$  if **for every clause**  $D \in \varphi$  with  $\neg l \in D$  holds that there exists  $k$  s.t.  $k \in C$ ,  $\neg k \in D$ ,  $l \neq k$  and  $k \leq_{\pi} l$ .

### Example

$$\exists x \forall a, b \exists y. (\neg x \vee a \vee y) \wedge (x \vee \neg a \vee y) \wedge (b \vee \neg y)$$

### Proposition 2

Removal of a quantified blocked literal preserves **satisfiability**.

### Proposition 3

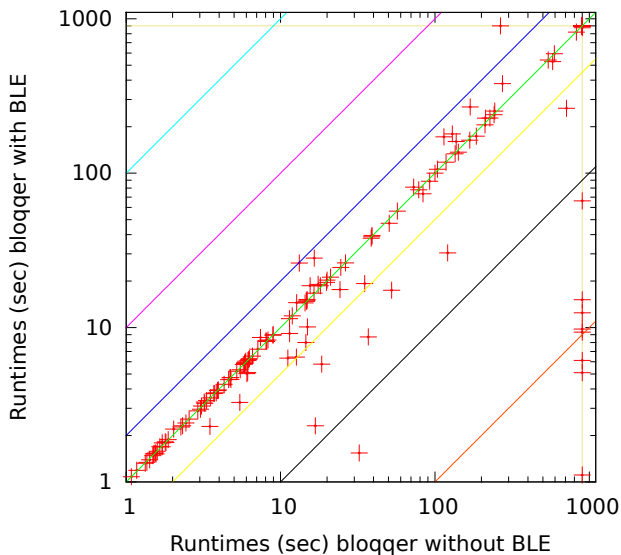
Elimination of blocked literals is **not confluent** in contrast to quantified blocked clause elimination:

$$\forall a, b \exists x. (a \vee b \vee x) \wedge (\neg a \vee \neg b \vee \neg x)$$



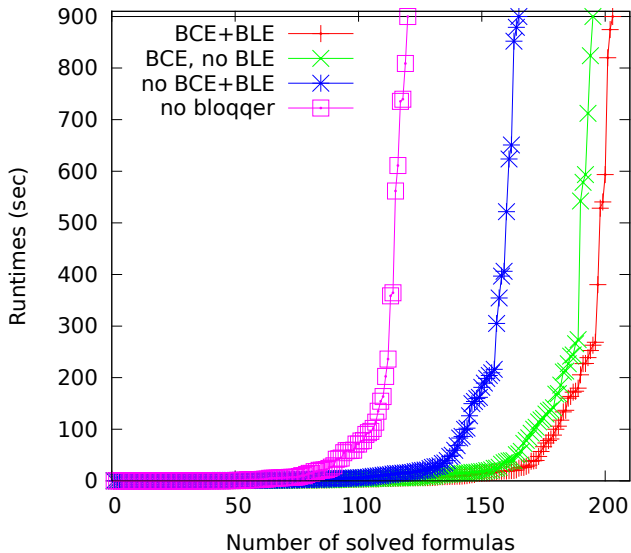
# Experimental Results

## Experimental Results: Runtime Scatter Plot



Above the diagonal: **faster** due to blocked literal elimination (BLE)

# Experimental Results: Runtime Cactus Plot



## Experimental Results: Only solvable with BLE

formula	formula statistics			preprocessing		solving	
	#vars	#cl	#Q	#bl	time	time	val
adder-6-sat	1727	1259	4	1278	0.74	0.36	T
C88020_0_0_inp	1046	2644	21	3	0.2	874.32	F
cache-coh-2-fixp-5	9604	28198	2	3599	9.32	–	F*
ethernet-fixpoint-3	12514	33884	2	3879	9.76	–	F*
k_branch_n-14	7068	33865	33	389	5.09	–	T*
k_branch_n-20	13821	78949	44	1397	12.45	–	T*
k_branch_p-15	8035	39595	34	239	6.12	–	F*
k_branch_p-21	15161	88627	46	1532	15.12	–	F*
s820_d7_s	24757	26960	3	5365	54.7	11.44	T

\* solved directly by bloqper

#vars : number of variables

#cl : number of clauses

#Q : number of quantifier alternations

#bl : number of eliminated **blocked literals**

# Conclusions

# Conclusions

We presented a new preprocessing technique:

- ▶ Blocked Literal Elimination (BLE) is useful in practice
- ▶ BLE is the dual of Blocked Clause Elimination (BCE)
- ▶ BLE is not confluent, in contrast to BCE

Directions for future work:

- ▶ Can the addition of blocked literals be helpful?
  - ▶ For example in combination with universal expansion
- ▶ Which other QRAT simplifications are useful in practice?
  - ▶ For example Asymmetric Blocked Literal Elimination

## Conclusions

We presented a new preprocessing technique:

- ▶ Blocked Literal Elimination (BLE) is useful in practice
- ▶ BLE is the dual of Blocked Clause Elimination (BCE)
- ▶ BLE is not confluent, in contrast to BCE

Directions for future work:

- ▶ Can the addition of blocked literals be helpful?
  - ▶ For example in combination with universal expansion
- ▶ Which other QRAT simplifications are useful in practice?
  - ▶ For example Asymmetric Blocked Literal Elimination

# Thanks!