JOHANNES KEPLER
UNIVERSITY LINZ

A Duality-Aware Calculus for Quantified Boolean Formulas

Katalin Fazekas Martina Seidl Armin Biere

Institute for Formal Models and Verification
Johannes Kepler University
Linz, Austria

24. September, 2016

18th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing

JOHANNES KEPLER UNIVERSITY LINZ

logics LOGICAL METHODS IN COMPUTER SCIENCE

ARiSE Austrian Rigorous Systems Engineering

FWF

INTRODUCTION

Introduction (1)

Quantified Boolean Formulas (QBF):

- Extension of propositional logic with explicit quantifiers ($\forall$, $\exists$) over the variables
- Canonical PSPACE-complete problem: more succinct encoding than SAT
- Several application domains: synthesis, AI, verification, ...

Introduction (1)

Quantified Boolean Formulas (QBF):

- Extension of propositional logic with explicit quantifiers ($\forall$, $\exists$) over the variables
- Canonical PSPACE-complete problem: more succinct encoding than SAT
- Several application domains: synthesis, AI, verification, ...

$$\overbrace{\exists x \exists y \forall u \exists z.(u \Rightarrow z) \wedge (y \vee u \vee \neg z) \wedge (x \vee \neg u \vee \neg z) \wedge (x \Leftrightarrow \neg y)}^{\text{closed QBF in prenex form}}$$

Introduction (1)

Quantified Boolean Formulas (QBF):

- Extension of propositional logic with explicit quantifiers ($\forall$, $\exists$) over the variables
- Canonical PSPACE-complete problem: more succinct encoding than SAT
- Several application domains: synthesis, AI, verification, ...

$$\overbrace{\underbrace{\exists x \exists y \forall u \exists z}_{\text{prefix}}.(u \Rightarrow z) \wedge (y \vee u \vee \neg z) \wedge (x \vee \neg u \vee \neg z) \wedge (x \Leftrightarrow \neg y)}^{\text{closed QBF in prenex form}}$$

JKU

Introduction (1)

Quantified Boolean Formulas (QBF):

- Extension of propositional logic with explicit quantifiers ($\forall$, $\exists$) over the variables
- Canonical PSPACE-complete problem: more succinct encoding than SAT
- Several application domains: synthesis, AI, verification, ...

$$\overbrace{\underbrace{\exists x \exists y \forall u \exists z.}_{\text{prefix}} \underbrace{(u \Rightarrow z) \wedge (y \vee u \vee \neg z) \wedge (x \vee \neg u \vee \neg z) \wedge (x \Leftrightarrow \neg y)}_{\text{matrix}}}^{\text{closed QBF in prenex form}}$$

JKU

Introduction (2)

- QBFs in Prenex CNF (PCNF):

$$\exists x \exists y \forall u \exists z . (\neg u \vee z) \wedge (y \vee u \vee \neg z) \wedge (x \vee \neg u \vee \neg z)$$

Introduction (2)

- QBFs in Prenex CNF (PCNF):

$$\exists x \exists y \forall u \exists z.(\overset{\text{literals}}{\overset{\downarrow}{\neg u}} \vee \overset{\downarrow}{z}) \wedge (y \vee u \vee \neg z) \wedge (x \vee \neg u \vee \neg z)$$

JKU

Introduction (2)

- QBFs in Prenex CNF (PCNF):

$$\exists x \exists y \forall u \exists z.(\overset{\text{literals}}{\overset{\downarrow}{\neg u}} \vee \overset{\downarrow}{z}) \wedge \overbrace{(y \vee u \vee \neg z)}^{\text{clause}} \wedge (x \vee \neg u \vee \neg z)$$

JKU

Introduction (2)

- QBFs in Prenex CNF (PCNF):

literals

$$\exists x \exists y \forall u \exists z . \underbrace{(\overset{\downarrow}{\neg u} \vee \overset{\downarrow}{z}) \wedge \overbrace{(y \vee u \vee \neg z)}^{\text{clause}} \wedge (x \vee \neg u \vee \neg z)}_{\text{CNF}}$$

Introduction (2)

- QBFs in Prenex CNF (PCNF):

literals

$$\exists x \exists y \forall u \exists z.\underbrace{(\overset{\downarrow}{\neg u} \vee \overset{\downarrow}{z}) \wedge \overbrace{(y \vee u \vee \neg z)}^{\text{clause}} \wedge (x \vee \neg u \vee \neg z)}_{\text{CNF}}$$

- QBFs in Prenex DNF (PDNF):

$$\forall x \forall y \exists u \forall z.(u \wedge \neg z) \vee (\neg y \wedge \neg u \wedge z) \vee (\neg x \wedge u \wedge z)$$

Introduction (2)

- QBFs in Prenex CNF (PCNF):

$$\exists x \exists y \forall u \exists z.\underbrace{(\overset{\text{literals}}{\neg u \vee z}) \wedge \overbrace{(y \vee u \vee \neg z)}^{\text{clause}} \wedge (x \vee \neg u \vee \neg z)}_{\text{CNF}}$$

- QBFs in Prenex DNF (PDNF):

$$\forall x \forall y \exists u \forall z.\overbrace{(u \wedge \neg z)}^{\text{cube}} \vee (\neg y \wedge \neg u \wedge z) \vee (\neg x \wedge u \wedge z)$$

Introduction (2)

- QBFs in Prenex CNF (PCNF):

$$\exists x \exists y \forall u \exists z.\underbrace{(\overset{\text{literals}}{\overset{\downarrow}{\neg u}} \vee \overset{\downarrow}{z}) \wedge \overbrace{(y \vee u \vee \neg z)}^{\text{clause}} \wedge (x \vee \neg u \vee \neg z)}_{\text{CNF}}$$

- QBFs in Prenex DNF (PDNF):

$$\forall x \forall y \exists u \forall z.\underbrace{\overbrace{(u \wedge \neg z)}^{\text{cube}} \vee (\neg y \wedge \neg u \wedge z) \vee (\neg x \wedge u \wedge z)}_{\text{DNF}}$$

Introduction (3)

PCNF (PDNF) formula under assignment

- remove clauses (cubes) with satisfied (falsified) literals
- remove falsified (satisfied) literals from clauses (cubes)

Introduction (3)

PCNF (PDNF) formula under assignment

- remove clauses (cubes) with satisfied (falsified) literals
- remove falsified (satisfied) literals from clauses (cubes)

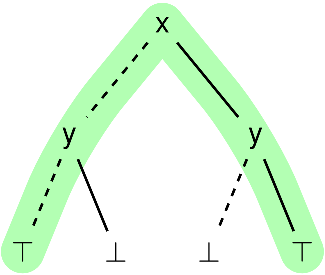
Semantics of QBF

- QBF $\forall x \mathcal{Q}.\varphi$ is satisfiable iff $\mathcal{Q}.\varphi[x]$ **and** $\mathcal{Q}.\varphi[\neg x]$ are satisfiable
- QBF $\exists x \mathcal{Q}.\varphi$ is satisfiable iff $\mathcal{Q}.\varphi[x]$ **or** $\mathcal{Q}.\varphi[\neg x]$ is satisfiable

Introduction (4)

Tree model of true formula:

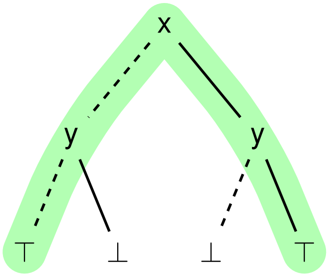
$$\forall x \exists y.(x \vee \bar{y}) \wedge (\bar{x} \vee y)$$



Introduction (4)

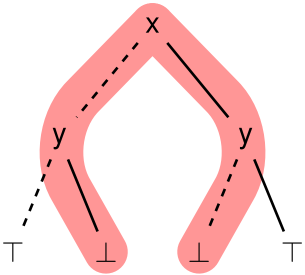
Tree model of true formula:

$\forall x \exists y.(x \vee \bar{y}) \wedge (\bar{x} \vee y)$



Tree refutation of false formula:

$\exists x \forall y.(x \vee \bar{y}) \wedge (\bar{x} \vee y)$



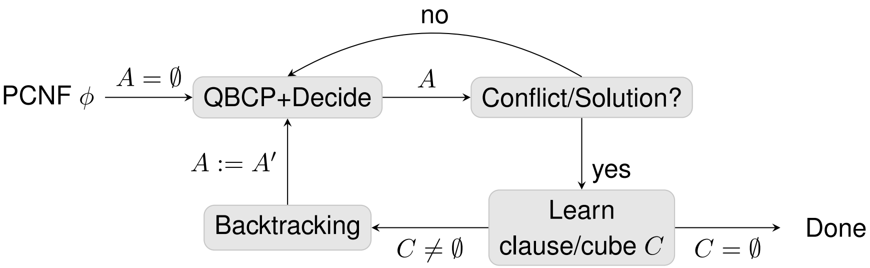
SEARCH-BASED QBF SOLVING

Search-Based QBF Solving: QCDCL (1)

```
Result qcdcl (PCNF φ)
  Result R = UNDEF;
  Assignment A = ∅;
  while (true)
    /* Simplify under A. */
    (R,A) = qbcp(φ,A);
    if (R == UNDET)
      /* Decision making. */
      A = assign_dec_var(φ,A);
    else
      /* Backtracking. */
      /* R == UNSAT/SAT */
      B = analyze(R,A);
      if (B == INVALID)
        return R;
      else
        A = backtrack(B);
```

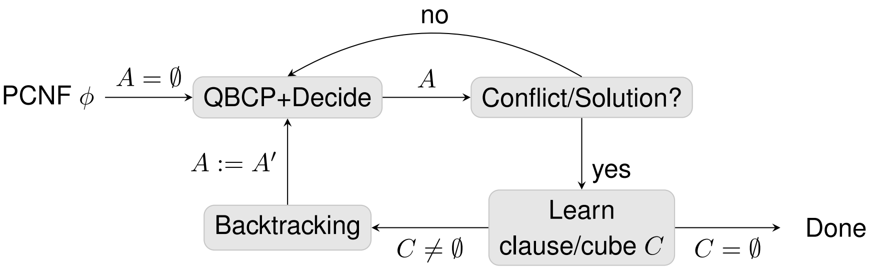
- QBF-specific version of SAT CDCL algorithm
- expects the problem to be formulated in PCNF
- traverses the assignment tree of the input formula
- conflict analysis similar to SAT solvers
- satisfaction recognition requires additional efforts

Search-Based QBF Solving: QCDCL (2)



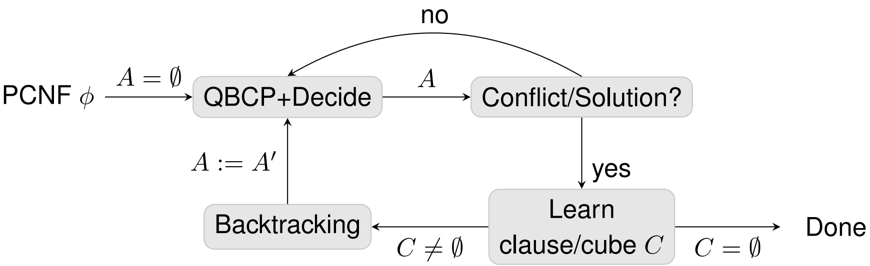
1. Construct assignment (A) by QBF specific propagation and decisions.

Search-Based QBF Solving: QCDCL (2)



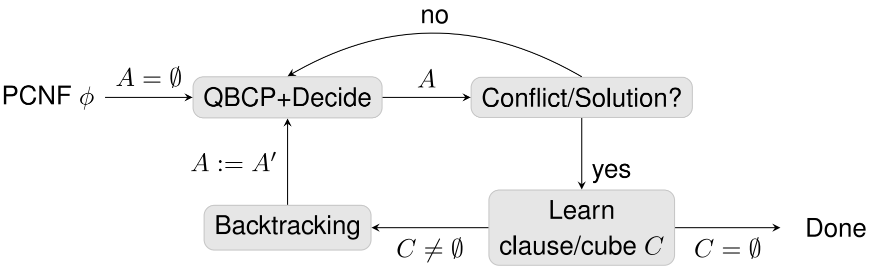
1. Construct assignment (A) by QBF specific propagation and decisions.
2. Check the followings:
   - ☐ Is there a falsified clause under A and universal reduction? (Conflict)
   - ☐ Are all the clauses of the formula satisfied under A? (Solution)

Search-Based QBF Solving: QCDCL (2)



3. Derive a clause (cube) C from A and ϕ and learn it.

Search-Based QBF Solving: QCDCL (2)



3. Derive a clause (cube) C from A and ϕ and learn it.
4. Use the learned clause (cube) to backtrack.
   - ☐ $C = \emptyset$: no place to backtrack, the formula is UNSAT (SAT).
   - ☐ $C \neq \emptyset$: C is 'driving' the backtracking.

Observations

- Either too technical (pseudo-code) or simply informal (high-level workflow) description of search-based QBF solvers

Observations

- Either too technical (pseudo-code) or simply informal (high-level workflow) description of search-based QBF solvers
- How to analyse the behaviour of search-based QBF solvers?

Observations

- Either too technical (pseudo-code) or simply informal (high-level workflow) description of search-based QBF solvers
- How to analyse the behaviour of search-based QBF solvers?
- How to compare them with other QBF solving approaches?

Observations

- Either too technical (pseudo-code) or simply informal (high-level workflow) description of search-based QBF solvers
- How to analyse the behaviour of search-based QBF solvers?
- How to compare them with other QBF solving approaches?
- How to verify them?

ABSTRACT QBF SOLVING

Duality-Aware Abstract QBF Solver

Abstract Solver:

- Describes the essential properties of QBF solvers similar to the well-known DPLL transition system
- Provides a framework for analysing, comparing and composing solvers without technical details
- Duality-Aware: Conflicts and solutions are handled in the same way

State Transition System as QBF Solver

Possible states of a QBF solver: $A \parallel \mathcal{D} \parallel \mathcal{C}$

- A quantifier prefix: $\mathcal{Q}$
- The current assignment (sequence of literals): A
- A set of cubes: $\mathcal{D}$
- A set of clauses: $\mathcal{C}$

State Transition System as QBF Solver

Possible states of a QBF solver: $A \parallel \mathcal{D} \parallel \mathcal{C}$

- A quantifier prefix: $\mathcal{Q}$
- The current assignment (sequence of literals): A
- A set of cubes: $\mathcal{D}$
- A set of clauses: $\mathcal{C}$

Possible steps of a QBF solver:

- Transition relation over the states defined by conditional transition rules
- Different specializations, heuristics can be seen as refinements of the relations

State Transition System as QBF Solver

Possible states of a QBF solver: $A \parallel \mathcal{D} \parallel \mathcal{C}$

- A quantifier prefix: $\mathcal{Q}$
- The current assignment (sequence of literals): A
- A set of cubes: $\mathcal{D}$
- A set of clauses: $\mathcal{C}$

Possible steps of a QBF solver:

- Transition relation over the states defined by conditional transition rules
- Different specializations, heuristics can be seen as refinements of the relations

Describe the solving process as a derivation in the calculus.

Calculus Rules

$$\frac{A \parallel \mathcal{D} \parallel \mathcal{C} \wedge C}{A\ \ell_\exists \parallel \mathcal{D} \parallel \mathcal{C} \wedge C}\ \ell_\exists \text{ existential unit in } C[A] \qquad (\mathrm{Unit}_\exists)$$

$$\frac{A \parallel \mathcal{D} \vee C \parallel \mathcal{C}}{A\ \neg\ell_\forall \parallel \mathcal{D} \vee C \parallel \mathcal{C}}\ \ell_\forall \text{ universal unit in } C[A] \qquad (\mathrm{Unit}_\forall)$$

$$\frac{A \parallel \mathcal{D} \parallel \mathcal{C}}{A\ \ell_\exists \parallel \mathcal{D} \parallel \mathcal{C}}\ \ell_\exists \in \mathcal{R}_\exists^\mathcal{Q}(\mathcal{D}[A]) \text{ is pure} \qquad (\mathrm{Pure}_\exists)$$

$$\frac{A \parallel \mathcal{D} \parallel \mathcal{C}}{A\ \neg\ell_\forall \parallel \mathcal{D} \parallel \mathcal{C}}\ \ell_\forall \in \mathcal{R}_\forall^\mathcal{Q}(\mathcal{C}[A]) \text{ is pure} \qquad (\mathrm{Pure}_\forall)$$

$$\frac{A \parallel \mathcal{D} \parallel \mathcal{C}}{A \parallel \mathcal{D} \parallel \mathcal{C} \wedge C}\ \mathcal{C} \vDash_\mathcal{Q} C \qquad (\mathrm{Learn}_{\mathrm{CNF}})$$

$$\frac{A \parallel \mathcal{D} \parallel \mathcal{C}}{A \parallel \mathcal{D} \vee C \parallel \mathcal{C}}\ \mathcal{D} \vDash^\mathcal{Q} C \qquad (\mathrm{Learn}_{\mathrm{DNF}})$$

$$\frac{A\ \ell_\exists^d A' \parallel \mathcal{D} \parallel \mathcal{C}}{A \parallel \mathcal{D} \parallel \mathcal{C}} \qquad (\mathrm{Undo}_\exists)$$

$$\frac{A\ \ell_\forall^d A' \parallel \mathcal{D} \parallel \mathcal{C}}{A \parallel \mathcal{D} \parallel \mathcal{C}} \qquad (\mathrm{Undo}_\forall)$$

$$\frac{A \parallel D \parallel \mathcal{C} \wedge \emptyset}{\bot} \qquad (\mathrm{Final}_{\mathrm{CNF}})$$

$$\frac{A \parallel \mathcal{D} \vee \emptyset \parallel \mathcal{C}}{\top} \qquad (\mathrm{Final}_{\mathrm{DNF}})$$

$$\frac{A \parallel \mathcal{D} \parallel \mathcal{C}}{A\ \ell^d \parallel \mathcal{D} \parallel \mathcal{C}}\ \ell \text{ is unassigned and all } \ell' \text{ with } \ell' <_\mathcal{Q} \ell \text{ are assigned in } A \qquad (\mathrm{Decide})$$

Remarks

- Strategy: Additional constraints in order to guarantee termination and to make the solver more realistic.

Remarks

- Strategy: Additional constraints in order to guarantee termination and to make the solver more realistic.
- Extendable: further rules to represent functionalities of practical solvers (e.g. forget, restart).

Remarks

- Strategy: Additional constraints in order to guarantee termination and to make the solver more realistic.
- Extendable: further rules to represent functionalities of practical solvers (e.g. forget, restart).
- If duality can not be assumed, it is possible to easily adopt the system for PCNF-based solvers.

Example (1)

- Given the following formula

$$\exists x \forall y . x \Leftrightarrow y$$

Example (1)

- Given the following formula

$$\exists x \forall y . x \Leftrightarrow y$$

- in PCNF ($\mathcal{C}$):

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

Example (1)

- Given the following formula

$$\exists x \forall y . x \Leftrightarrow y$$

- in PCNF ($\mathcal{C}$):

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

- in PDNF ($\mathcal{D}$):

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

Example (1)

- Given the following formula

$$\exists x \forall y . x \Leftrightarrow y$$

- in PCNF ($\mathcal{C}$):

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

- in PDNF ($\mathcal{D}$):

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

- Merged Prefix:
  $\mathcal{Q} = \exists x \forall y \exists p \forall q$

Example (1)

- Given the following formula

$$\exists x \forall y . x \Leftrightarrow y$$

- in PCNF ($\mathcal{C}$):

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

- in PDNF ($\mathcal{D}$):

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

- Merged Prefix:
  $\mathcal{Q} = \exists x \forall y \exists p \forall q$

- Starting state:
  $S = \emptyset \parallel \mathcal{D} \parallel \mathcal{C}$

Example (2)

$\mathcal{Q} = \exists x \forall y \exists p \forall q$

A	$\mathcal{D}$	$\mathcal{C}$
	q	p
$\emptyset$	$\neg x \wedge \neg y \wedge \neg q$	$\neg x \vee y \vee \neg p$
	$x \wedge y \wedge \neg q$	$x \vee \neg y \vee \neg p$

$\emptyset \parallel \mathcal{D} \parallel \mathcal{C}$

Example (2)

$\mathcal{Q} = \exists x \forall y \exists p \forall q$

A	$\mathcal{D}$	$\mathcal{C}$
	q	p
$\emptyset$	$\neg x \wedge \neg y \wedge \neg q$	$\neg x \vee y \vee \neg p$
	$x \wedge y \wedge \neg q$	$x \vee \neg y \vee \neg p$

$\emptyset \parallel \mathcal{D} \parallel \mathcal{C}$

Unit$_\exists$:

$$\frac{A \parallel \mathcal{D} \parallel \mathcal{C} \wedge C}{A\ \ell_\exists \parallel \mathcal{D} \parallel \mathcal{C} \wedge C}$$

$\ell_\exists$ existential unit in $C[A]$

JKU

Example (2)

$\mathcal{Q} = \exists x \forall y \exists p \forall q$

A	$\mathcal{D}$	$\mathcal{C}$
	q	p
p	$\neg x \wedge \neg y \wedge \neg q$	$\neg x \vee y \vee \neg p$
	$x \wedge y \wedge \neg q$	$x \vee \neg y \vee \neg p$

$$\frac{\emptyset \parallel \mathcal{D} \parallel \mathcal{C}}{p \parallel \mathcal{D} \parallel \mathcal{C}}\ \text{Unit}_\exists$$

Unit$_\exists$:

$$\frac{A \parallel \mathcal{D} \parallel \mathcal{C} \wedge C}{A\ \ell_\exists \parallel \mathcal{D} \parallel \mathcal{C} \wedge C}$$

$\ell_\exists$ existential unit in $C[A]$

Example (2)

$\mathcal{Q} = \exists x \forall y \exists p \forall q$

A	$\mathcal{D}[A]$	$\mathcal{C}[A]$
	q	p
p	$\neg x \wedge \neg y \wedge \neg q$	$\neg x \vee y \vee \neg p$
	$x \wedge y \wedge \neg q$	$x \vee \neg y \vee \neg p$

$$\frac{\emptyset \parallel \mathcal{D} \parallel \mathcal{C}}{p \parallel \mathcal{D} \parallel \mathcal{C}} \, \mathrm{Unit}_{\exists}$$

JKU

Example (2)

$\mathcal{Q} = \exists x \forall y \exists p \forall q$

A	$\mathcal{R}^{\mathcal{Q}}_{\exists}(\mathcal{D}[A])$	$\mathcal{R}^{\mathcal{Q}}_{\forall}(\mathcal{C}[A])$
	q	p
p	$\neg x \wedge \neg y \wedge \neg q$	$\neg x \vee y \vee \neg p$
	$x \wedge y \wedge \neg q$	$x \vee \neg y \vee \neg p$

$$\frac{\emptyset \parallel \mathcal{D} \parallel \mathcal{C}}{p \parallel \mathcal{D} \parallel \mathcal{C}} \text{Unit}_{\exists}$$

Example (2)

$\mathcal{Q} = \exists x \forall y \exists p \forall q$

A	$\mathcal{R}^{\mathcal{Q}}_{\exists}(\mathcal{D}[A])$	$\mathcal{R}^{\mathcal{Q}}_{\forall}(\mathcal{C}[A])$
	q	p
p	$\neg x \wedge \neg y \wedge \neg q$	$\neg x \vee y \vee \neg p$
	$x \wedge y \wedge \neg q$	$x \vee \neg y \vee \neg p$

$$\frac{\emptyset \parallel \mathcal{D} \parallel \mathcal{C}}{p \parallel \mathcal{D} \parallel \mathcal{C}} \text{ Unit}_{\exists}$$

$\text{Unit}_{\forall}$:

$$\frac{A \parallel \mathcal{D} \vee C \parallel \mathcal{C}}{A\ \neg\ell_{\forall} \parallel \mathcal{D} \vee C \parallel \mathcal{C}}$$

$\ell_{\forall}$ universal unit in $C[A]$

Example (2)

$\mathcal{Q} = \exists x \forall y \exists p \forall q$

A	$\mathcal{R}^{\mathcal{Q}}_{\exists}(\mathcal{D}[A])$	$\mathcal{R}^{\mathcal{Q}}_{\forall}(\mathcal{C}[A])$
	q	p
$p\ \neg q$	$\neg x \wedge \neg y \wedge \neg q$	$\neg x \vee y \vee \neg p$
	$x \wedge y \wedge \neg q$	$x \vee \neg y \vee \neg p$

$$\dfrac{\dfrac{\emptyset \parallel \mathcal{D} \parallel \mathcal{C}}{p \parallel \mathcal{D} \parallel \mathcal{C}}\ \mathrm{Unit}_{\exists}}{p\ \neg q \parallel \mathcal{D} \parallel \mathcal{C}}\ \mathrm{Unit}_{\forall}$$

$\mathrm{Unit}_{\forall}$:

$$\frac{A \parallel \mathcal{D} \vee C \parallel \mathcal{C}}{A\ \neg \ell_{\forall} \parallel \mathcal{D} \vee C \parallel \mathcal{C}}$$

$\ell_{\forall}$ universal unit in $C[A]$

Example (2)

$\mathcal{Q} = \exists x \forall y \exists p \forall q$

A	$\mathcal{R}^{\mathcal{Q}}_{\exists}(\mathcal{D}[A])$	$\mathcal{R}^{\mathcal{Q}}_{\forall}(\mathcal{C}[A])$
	q	p
$p \; \neg q$	$\neg x \wedge \neg y \wedge \neg q$	$\neg x \vee y \vee \neg p$
	$x \wedge y \wedge \neg q$	$x \vee \neg y \vee \neg p$

$$\dfrac{\dfrac{\emptyset \parallel \mathcal{D} \parallel \mathcal{C}}{p \parallel \mathcal{D} \parallel \mathcal{C}}\,\text{Unit}_\exists}{p\;\neg q \parallel \mathcal{D} \parallel \mathcal{C}}\,\text{Unit}_\forall$$

$\text{Unit}_\exists$:

$$\dfrac{A \parallel \mathcal{D} \parallel \mathcal{C} \wedge C}{A\;\ell_\exists \parallel \mathcal{D} \parallel \mathcal{C} \wedge C}$$

$\ell_\exists$ existential unit in $C[A]$

Example (2)

$\mathcal{Q} = \exists x \forall y \exists p \forall q$

A	$\mathcal{R}^{\mathcal{Q}}_{\exists}(\mathcal{D}[A])$	$\mathcal{R}^{\mathcal{Q}}_{\forall}(\mathcal{C}[A])$
	q	p
$p \; \neg q \; x$	$\neg x \wedge \neg y \wedge \neg q$	$\neg x \vee y \vee \neg p$
	$x \wedge y \wedge \neg q$	$x \vee \neg y \vee \neg p$

$$\cfrac{\cfrac{\cfrac{\emptyset \parallel \mathcal{D} \parallel \mathcal{C}}{p \parallel \mathcal{D} \parallel \mathcal{C}}\ \mathrm{Unit}_{\exists}}{p \; \neg q \parallel \mathcal{D} \parallel \mathcal{C}}\ \mathrm{Unit}_{\forall}}{p \; \neg q \; x \parallel \mathcal{D} \parallel \mathcal{C}}\ \mathrm{Unit}_{\exists}$$

$\mathrm{Unit}_{\exists}$:

$$\frac{A \parallel \mathcal{D} \parallel \mathcal{C} \wedge C}{A \; \ell_{\exists} \parallel \mathcal{D} \parallel \mathcal{C} \wedge C}$$

$\ell_{\exists}$ existential unit in $C[A]$

Example (2)

$\mathcal{Q} = \exists x \forall y \exists p \forall q$

A	$\mathcal{R}_\exists^\mathcal{Q}(\mathcal{D}[A])$	$\mathcal{R}_\forall^\mathcal{Q}(\mathcal{C}[A])$
	q	p
$p\ \neg q\ x$	$\neg x \wedge \neg y \wedge \neg q$	$\neg x \vee y \vee \neg p$
	$x \wedge y \wedge \neg q$	$x \vee \neg y \vee \neg p$

$$\cfrac{\cfrac{\cfrac{\emptyset \parallel \mathcal{D} \parallel \mathcal{C}}{p \parallel \mathcal{D} \parallel \mathcal{C}}\ \text{Unit}_\exists}{p\ \neg q \parallel \mathcal{D} \parallel \mathcal{C}}\ \text{Unit}_\forall}{p\ \neg q\ x \parallel \mathcal{D} \parallel \mathcal{C}}\ \text{Unit}_\exists$$

Learn$_{\text{CNF}}$:

$$\frac{A \parallel \mathcal{D} \parallel \mathcal{C}}{A \parallel \mathcal{D} \parallel \mathcal{C} \wedge C}$$

$$\mathcal{C} \models_\mathcal{Q} C$$

Example (2)

$\mathcal{Q} = \exists x \forall y \exists p \forall q$

A	$\mathcal{R}^{\mathcal{Q}}_{\exists}(\mathcal{D}[A])$	$\mathcal{R}^{\mathcal{Q}}_{\forall}(\mathcal{C}[A])$
	q	p
$p\ \neg q\ x$	$\neg x \wedge \neg y \wedge \neg q$	$\neg x \vee y \vee \neg p$
	$x \wedge y \wedge \neg q$	$x \vee \neg y \vee \neg p$
		$\emptyset$

$$\cfrac{\cfrac{\cfrac{\cfrac{\emptyset \parallel \mathcal{D} \parallel \mathcal{C}}{p \parallel \mathcal{D} \parallel \mathcal{C}}\ \text{Unit}_{\exists}}{p\ \neg q \parallel \mathcal{D} \parallel \mathcal{C}}\ \text{Unit}_{\forall}}{p\ \neg q\ x \parallel \mathcal{D} \parallel \mathcal{C}}\ \text{Unit}_{\exists}}{p\ \neg q\ x \parallel \mathcal{D} \parallel \mathcal{C} \wedge \emptyset}\ \text{Learn}_{\text{CNF}}$$

Learn$_{\text{CNF}}$:

$$\frac{A \parallel \mathcal{D} \parallel \mathcal{C}}{A \parallel \mathcal{D} \parallel \mathcal{C} \wedge C}$$

$$\mathcal{C} \vDash_{\mathcal{Q}} C$$

Example (2)

$\mathcal{Q} = \exists x \forall y \exists p \forall q$

A	$\mathcal{R}_{\exists}^{\mathcal{Q}}(\mathcal{D}[A])$	$\mathcal{R}_{\forall}^{\mathcal{Q}}(\mathcal{C}[A])$
	q	p
$p\ \neg q\ x$	$\neg x \wedge \neg y \wedge \neg q$	$\neg x \vee y \vee \neg p$
	$x \wedge y \wedge \neg q$	$x \vee \neg y \vee \neg p$
		$\emptyset$

$$\cfrac{\cfrac{\cfrac{\cfrac{\emptyset \parallel \mathcal{D} \parallel \mathcal{C}}{p \parallel \mathcal{D} \parallel \mathcal{C}}\ \text{Unit}_{\exists}}{p\ \neg q \parallel \mathcal{D} \parallel \mathcal{C}}\ \text{Unit}_{\forall}}{p\ \neg q\ x \parallel \mathcal{D} \parallel \mathcal{C}}\ \text{Unit}_{\exists}}{p\ \neg q\ x \parallel \mathcal{D} \parallel \mathcal{C} \wedge \emptyset}\ \text{Learn}_{\text{CNF}}$$

$\text{Final}_{\text{CNF}}$:

$$\frac{A \parallel D \parallel \mathcal{C} \wedge \emptyset}{\bot}$$

Example (2)

$\mathcal{Q} = \exists x \forall y \exists p \forall q$

A	$\mathcal{R}^{\mathcal{Q}}_{\exists}(\mathcal{D}[A])$	$\mathcal{R}^{\mathcal{Q}}_{\forall}(\mathcal{C}[A])$
	q	
$p \ \neg q \ x$	$\neg x \wedge \neg y \wedge \neg q$	$\bot$
	$x \wedge y \wedge \neg q$	

$$\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\emptyset \parallel \mathcal{D} \parallel \mathcal{C}}{p \parallel \mathcal{D} \parallel \mathcal{C}}\ \text{Unit}_{\exists}}{p\ \neg q \parallel \mathcal{D} \parallel \mathcal{C}}\ \text{Unit}_{\forall}}{p\ \neg q\ x \parallel \mathcal{D} \parallel \mathcal{C}}\ \text{Unit}_{\exists}}{p\ \neg q\ x \parallel \mathcal{D} \parallel \mathcal{C} \wedge \emptyset}\ \text{Learn}_{\text{CNF}}}{\bot}\ \text{Final}_{\text{CNF}}$$

$\text{Final}_{\text{CNF}}$:

$$\frac{A \parallel D \parallel \mathcal{C} \wedge \emptyset}{\bot}$$

Conclusion

Abstract search-based QBF solvers

- Simple formalism to describe the behavior of search-based QBF solvers without the technical details
- Provides better understanding of individual solving techniques
- Flexible representation: specialization of calculus rules to describe e.g. different decision heuristics
- Step towards verified QBF solvers

Future work

- Formalize preprocessing techniques
- Comparison to non-QCDCL solvers