On Models for Quantified Boolean Formulas

Hans Kleine Büning
and
(Zhao Xishun, Zongshan University
Guangzhou, China)

Paris, November 2003

Contents:

Notation

QBF ^{free}: quantified Boolean formulas including formulas with free variables

$\Phi(z) = \forall x \exists y (x \vee (\neg z \wedge y) \vee (\neg x \vee \neg z))$

QBF : quantified Boolean formulas **without** free variables

QX^{free}: quantified Boolean formulas in prefix normal form and kernel in X

QX: formulas in QX^* **without** free variables

QCNF ^{free}: quantified Boolean formulas in prefix NF and kernel in CNF

$\Phi(x_1) = \forall x \exists y \ \ (\neg x \vee y \vee x_1) \wedge (x \vee \neg y)$

QHORN : *QCNF* with Horn-kernel

$\Phi = \forall x \exists y \ \ (\neg x \vee y) \wedge (x \vee \neg y)$

Semantics

1. Quantified Boolean Formulas QBF

$\Phi = \forall x \exists y \ \ (x \vee y) \wedge (\neg x \vee \neg y)$ is true (satisfiable)

$f_y(x) = \neg x$ satisfies Φ

$\forall x \ \ (x \vee y) \wedge (\neg x \vee \neg y)[y/f_y(x)] = \forall x \ \ (x \vee \neg x) \wedge (\neg x \vee x)$ is true

(f_y) is a *model* for Φ.

$\Phi = \forall x \forall z \exists y \ \ (\neg x \vee y) \wedge (z \vee \neg y)$ is not true (satisfiable)

2. Quantified Boolean formulas *QBF* ^{free}

$\Phi(x, z) = \exists y : (x \to y) \wedge (y \to z)$ is equivalent to the propositional formula $x \to z$.

For $f_y(x, z) = (x \wedge z)$

$$(x \to y) \wedge (y \to z)[y/f_y(x, z)] =$$

$$(x \to f_y(x, z)) \wedge (f_y(x, z) \to z) =$$

$$(x \to (x \wedge z)) \wedge ((x \wedge z) \to z) \approx (x \to z)$$

Boolean Functions $f : \{0,1\}^n \longrightarrow \{0,1\}$

Representations:

1. truth table
2. propositional formula (in CNF)

   $f(x,z,y) = (\neg x \vee y) \wedge (\neg z \vee x)$
3. quantified Boolean formula with free variables

   $f(x,z) = \exists y((x \rightarrow y) \wedge (y \rightarrow z)) \approx (x \rightarrow z)$
4. .....
5. .....

Definition 1 *Let* $\Phi = \forall z_1 \exists y_1 \cdots \forall z_m \exists y_m \;\; \phi$ *be a satisfiable QBF .*

For Boolean functions $f_{y_i}(z_1, \cdots z_i)$, *we say* $M = (f_{y_1}, \cdots, f_{y_m})$ *is a* model *for* Φ

if, and only if

$\forall z_1 \cdots \forall z_t : \phi[y_1/f_{y_1}(z_1), \cdots, y_m/f_{y_m}(z_1, \cdots, z_m)]$ *is true.*

Example: $\forall x \exists y : (\neg x \vee y) \wedge (x \vee \neg y)$

$f(x) = x$ is a model.

$\forall x : (\neg x \vee y) \wedge (x \vee \neg y)[y/f(x)] = \forall x(\neg x \vee x) \wedge (x \vee \neg x)$ is true.

$(\neg x \vee x) \wedge (x \vee \neg x)$ is a tautology.

$g(x) = \neg x$ is not a model

$\forall x : (\neg x \vee y) \wedge (x \vee \neg y)[y/g(x)] = \forall x(\neg x \vee \neg x) \wedge (x \vee x)$

$\neg x \wedge x$ is not a tautology.

K-Model Checking Problem for $X \subseteq$ *QCNF* :

Instance: A formula $\Phi \in X$ and a sequence $M = (f_1, \cdots, f_n)$, ($f_i \in K$) of propositional formulas.

Query: Is M a K-model for Φ?

K-Model Problem for $X \subseteq$ *QCNF* :

Instance: A formula $\Phi \in X$

Query: Does there exist a K-model for Φ?

Example: $X = \exists^*$*CNF*

Propositional Logic : Satisfiability Problem *SAT*

$\phi = (x \vee y) \wedge (\neg x \vee y) \wedge (\neg z \vee \neg y)$

Does there exists a satisfying truth assignment for ϕ?

$\exists x \exists y \exists z : \phi$ is true?

$\exists \cdots \exists \phi$ has a model?

Example: $\exists x \exists y \exists z (x \vee y) \wedge (\neg x \vee y) \wedge (\neg z \vee \neg y)$

$M = (y = 1, z = 0)$ is a model

Quantified Boolean Formulas

Satisfiability Problem / Evaluation Problem / Model Problem

Example: $\forall x \exists y (x \wedge (y \vee \neg x))$ is satisfiable?

Complexity: *PSPACE* –complete

$\Phi \in$ *QCNF* : $\quad \Phi = \forall z_1 \exists y_1 \cdots \forall z_n \exists y_n \phi$

$M = (f^1_{y_1}, \cdots, f^n_{y_n})$ is a model for Φ

$$\Longleftrightarrow$$

$\forall z_1 \cdots \forall z_n \phi[y_1/f_{y_1}(z_1), \cdots, y_n/f_{y_n}(z_1, \cdots, z_n)]$ is true

$$\Longleftrightarrow$$

$\phi[y_1/f_{y_1}(z_1), \cdots, y_n/f_{y_n}(z_1, \cdots, z_n)]$ is a tautology

Lemma 1 *(MCH-P)*
The model checking problem for QCNF is coNP–complete.

For example $K = 2 - CNF$

Characterization Problem:

For a given class of propositional formulas K determine the class S with $\mathcal{Z}(S, K)$

1. $\forall \Phi \in S$: Φ has a K–model
2. $\forall \Phi \in S \quad \forall \Phi' \subset_{cl} \Phi : \quad \Phi' \notin S$
3. $\forall \Phi : (\quad \Phi$ has a K-model $\Longrightarrow \exists \Phi' \subseteq_{cl} \Phi : \quad \Phi' \in S$).

Lemma 2 *(Uniqueness)*

1. $\forall S_1, S_2 \subseteq$ *QCNF* $\forall K \subseteq$ *propositional formulas:*

$\mathcal{Z}(S_1, K)$ *and* $\mathcal{Z}(S_2, K) \Longrightarrow S_1 = S_2$

2. $\forall S \subseteq$ *QCNF* $\forall K_1, K_2 \subseteq$ *propositional formulas:*

$\mathcal{Z}(S, K_1)$ *and* $\mathcal{Z}(S, K_2) \Longrightarrow K_1 \approx K_2$

$K_0 = \{f \mid f \text{ is } 0 \text{ or } 1\}$
$K_1 := \{f \mid \exists i \exists \epsilon_i : f(x_1, ..., x_n) = x_i^{\epsilon_i} \} \cup K_0$
$K_2 := \{f \mid \exists I : f(x_1, ..., x_n) = \wedge_{i \in I} x_i\} \cup K_0$

Boolean functions K	model checking	model	$\mathcal{Z}(\mathbf{S}, K)$
CNF	*coNP* -c	*PSPACE* –c	*QCNF* $\cap$ *MINSAT*
K_0	linear time	NP-c	*Q1-CNF* $\cap$ *MINSAT*
K_1	linear time	NP-c	*Q2-CNF* $\cap$ *MINSAT*
K_2	quadratic time	NP-c	*QHORN* $\cap$ *MINSAT*
D_2	quadratic time	NP-c	*QHORN* [co–$\forall, \exists$] $\cap$ *MINSAT*
1-CNF	co-NP-c	Σ_2^P	
1-CNF for *Qk-CNF*	polynomial time	*NP*	
1-DNF	*coNP* -c	Σ_2^P	
2-CNF	*coNP* -c	Σ_2^P	
HORN	*coNP* -c		
2-HORN	*coNP* -c	Σ_2^P	
QCNF ^{free}	*PSPACE* -c	*PSPACE* -c	
QHORN ^{free}	*coNP* -c		

$K_0 = \{f \mid f \text{ is } 0 \text{ or } 1\}$

1. The K_0-model checking problem for *QCNF* is solvable in linear time.

$$\forall x_1 \exists y_1 \cdots \forall x_n \exists y_n : \phi \text{ is true}$$
$$\Longleftrightarrow$$
$$\forall x_1 \cdots \forall x_n : \phi[y_1/0, \cdots y_n/1] \text{ is true}$$
$$\Longleftrightarrow$$
$$\phi[y_1/0, \cdots y_n/1] \in \textit{CNF} \text{ is tautology}$$

2. The K_0-model problem for *QCNF* is *NP*-complete.

$K_1 := \{f \mid \exists i \exists \epsilon \in \{0,1\} : f(x_1, ..., x_n) = x_i^\epsilon\} \cup K_0$

Theorem 1 *1. For QCNF ,*
the K_1-model checking problem is solvable linear time
the K_1-model problem is NP-complete.

2. Any formula $\Phi \in$ Q2-CNF $\cap$ SAT has a K_1-model.

3. A formula $\Phi \in$ QCNF has a K_1-model $\Longleftrightarrow \exists \Phi' \subseteq_{cl} \Phi : \Phi' \in$ Q2-CNF $\cap$ SAT .

Example $\forall x \forall z \exists y_1 \forall x_2 \exists y_2 \exists y_3 : (\neg x \vee y_1) \wedge (x \vee y_2) \wedge (\neg y_1 \vee \neg y_2) \wedge (z \vee y_3)$

$M = (f_1, f_2, f_3)$ is a model , where $f_1(x,z) = x$, and $f_2(x,z) = \neg x$, and $f_3(x,z) = 1$

$\Phi = \forall x \forall z \exists y : (\neg x \vee z \vee y) \wedge (\neg z \vee \neg y)$

$M = (f)$, where $f(x,z) = \neg z$

$\Phi' = \forall x \forall z \exists y : (z \vee y) \wedge (\neg z \vee \neg y)$

QHORN = prefix + kernel in Horn form
The satisfiability problem is solvable in polynomial time

.

$K_2 := \{f \mid \exists I \subseteq \{1, \cdots, n\}, f(x_1, ..., x_n) = \wedge_{i \in I} x_i\} \cup K_0$

Theorem 2 *1. The K_2–model checking problem for QCNF is solvable in quadratic time.*

2. The K_2–model problem for QCNF is NP –complete.

3. Any formula $\Phi \in$ QHORN $\cap$ SAT has a K_2–model and a K_2–model can be computed in polynomial time.

4. A formula $\Phi \in$ QCNF has a K_2–model $\Longleftrightarrow \exists \Phi' \subseteq_{cl} \Phi : \Phi' \in$ QHORN $\cap$ SAT .

Theorem 3 *For $\Phi \in$ QHORN , if $M_1 = (f_1, \cdots, f_r)$ and $M_2 = (g_1, \cdots, g_r)$ are models for Φ,*
then $M_1 \cap M_2 = (f_1 \wedge g_1, \cdots, f_r \wedge g_r)$ is a model for Φ.

Lemma 3 *For* $X \in \{$ *1-CNF , 1-DNF , 2-CNF , HORN , 2-HORN* $\}$*,*
the X*–model checking problem is coNP –complete,*
whereas for $Y \in \{$ *1-CNF , 1-DNF , 2-CNF , 2-HORN* $\}$ *the* Y*–model problem is in* Σ_2^p*.*

Open Problem: Whether there is a polynomial q
such that any *QCNF* formula Φ with a *HORN* –model
has always a *HORN* –model of length $q(|\Phi|)$.

QCNF ^{free}–Models

A Boolean function f over n variables can be represented
as quantified Boolean formula
with *CNF* kernel and with at most n free variables.

$$\Phi(x_{1,1}, \cdots, x_{n,n}) = \exists y_1 \cdots \exists y_n((\neg y_1 \vee \cdots \vee \neg y_n) \wedge \bigwedge_{1 \leq i,j \leq n} (y_i \vee \neg x_{i,j}))$$

is equivalent to

$$\varphi = \bigwedge_{1 \leq j_1, \cdots, j_n \leq n} (\neg x_{1,j_1} \vee \cdots \vee \neg x_{n,j_n})$$

The length of Φ is $n^2 + 2n$, whereas φ has length n^{n+1}.

Proposition 1 *(Horn equivalence)*

1. $\forall \Phi \in$ *QHORN* ^{free} $\exists F \in$ *HORN* $: \Phi \approx F$,
2. *There exists formulas* $\Phi_n \in$ *QHORN* ^{free} *for which any equivalent CNF formula has superpolynomial length.*

Lemma 4 *The QCNF^{free}–model checking problem is PSPACE –complete.*

Theorem 4 *(QHORN^{free}–models)*

1. *The problem whether a QCNF has a QHORN^{free}–model is as hard as the problem whether a Horn–model exists.*
2. *The QHORN^{free}–model checking problem is as hard as the HORN –model checking problem.*