Quantified boolean formula (QBF)

A QBF is a boolean formula of the form

$$\underline{Q_1}\,\underline{x_1}\;\underline{Q_2}\,\underline{x_2}\cdots\underline{Q_n}\,\underline{x_n}\;\underline{\phi(x_1 x_2 \cdots x_n)}$$

either $\exists$ or $\forall$

Boolean literals

unquantified boolean formula.

→ In A fully quantified boolean formula every variable is quantified using either existential or universal quantifiers.

→ If such a formula evaluates to "TRUE", then the formula is in the language TQBF

TQBF $= \{\phi \mid \phi$ is a true fully quantified boolean formula$\}$

$\forall x\, \exists y\, \exists z\, ((x \vee z) \wedge y)$

Note: A fully quantified boolean formula is in the prenex normal form (PNF) if it has two basic parts:

* → A portion containing only quantifiers
* → A portion containing unquantified boolean formula.

$$\exists x_1 \forall x_2 \exists x_3 \ldots Q_n x_n\, \phi(x_1, x_2\, x_3 \ldots x_n)$$

where every variable falls within the scope of some quantifier.

Date ______
Page ______

*→ By introducing dummy variables, any formula in prenex normal form can be converted into a sentence where universal and existential quantifiers alternate.

$\exists x_1 \exists x_2 \varphi(x_1, x_2) \mapsto \exists x_1 \forall y_1 \exists x_2 \varphi(x_1 x_2)$.

TQBF is PSPACE Complete

1. TQBF $\in$ PSPACE.

$T(\varphi)$:

1. if φ has no quantifiers, then its an expression with only constants. Evaluate φ. Accept iff φ evaluates to 1.

2. if $\varphi = \exists x \psi$
↳ recursively call T on ψ, first with $x = 0$ and then with $x = 1$
Accept iff either one of the calls accept

3. if $\varphi = \forall x \psi$
↳ recursively call T on ψ, first with $x = 0$ and then with $x = 1$
Accept iff both of the calls accept

Example:

$1 \quad \forall x\, \exists y [(x \lor \neg y) \land (\neg x \lor y)]$

$1 \quad \exists y [(0 \lor \neg y) \land (\neg 0 \lor y)]$ $\qquad$ $1 \quad \exists y [(1 \lor \neg y) \land (\neg 1 \lor y)]$

$(0 \lor \neg 0) \land (\neg 0 \lor 0)$ $\qquad$ $(0 \lor \neg 1) \land (\neg 0 \lor 1)$ $\qquad$ $0 \quad (1 \lor \neg 0) \land (\neg 1 \lor 0)$

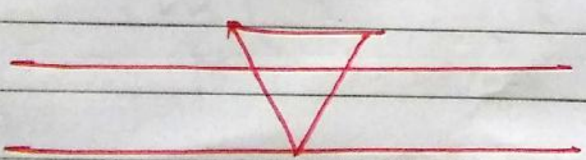
1 $\qquad$ 0

$(1 \lor \neg 1) \land (\neg 1 \lor 1)$

1

2. $\forall L \in PSPACE$

$L \leq_p TQBF$



$\Longrightarrow$ $\boxed{\forall x_1 \exists x_2 \forall x_3 [\,.\,]}$

$\leq_p$

Will the poly-space Machine M accept x $\qquad$ "Is the formula true"

Page

Let $G_{M,x}$ be the configuration graph of the m/c M on input x.

→ suppose we can define a

(unquantified) boolean formula.

$\varphi_i(C_1, C_2)$ such $\varphi_i(C_1, C_2) = 1$ iff

The configuration C_2 is reachable from C_1 in $G_{m,x}$ via a path of the length of at most 2^i.

Let m be the size of $G_{m,x}$.

$$m = O(p|x|).$$

We are trying to capture the formula

$$\varphi_m(C_{start}, C_{Accept})$$

GOAL is to calculate this in polynomial space

Now:

$$\varphi_i(C_1, C_2) = \exists C_3 (\varphi_{i-1}(C_1, C_3) \wedge \varphi_{i-1}(C_3, C_2))$$

size of $\varphi_i(..)$ is at least twice that $\varphi_{i-1}(...)$ and hence, if we inductively expand $\varphi_m(...)$ in the above fashion we would end up in $O(2^m)$ space for φ_m.

Page

Instead we can rewrite ~~ab~~ it as.

$$\varphi_i(C_1 C_2) = \exists C_3 \forall D_1 \forall D_2 \quad \text{--- (1)}$$

$$[(D_1 = C_1 \wedge D_2 = C_3) \vee (D_1 = C_3 \wedge D_2 = C_2) \rightarrow \varphi_{i-1}(D_1 D_2)$$

Note: $D_1 = C_1$ requires $O(m)$ space.

$\rightarrow$ $O(m) + \text{size}(\varphi_{i-1})$.

Inductively.

$$\text{size}(\varphi_m(\ldots)) = \underline{\text{size}(\varphi_{m-1}(\ldots))} + O(m)$$

$$= \text{size}(\varphi_{m-2}(\ldots)) + O(m) + O(m).$$

$$\vdots$$

$$= \underline{\text{size}(\varphi_0(\ldots))} + O(m^2).$$

illustration :

C_1	C_2	$\varphi_0(C_1, C_2)$
.. – – –*– – – – 1 2 3 4 5 6	*– – – – – – 1 2 3 4 5 6	0
– – –*– – –	– –*– – – –	1

"–" represents the cell of the tape and '*' represents the head position.

if the goal is to express $\varphi_0(C_1, C_2)$ using a poly size boolean formula.
Size of above table is $2^{O(m)}$

Since total number of possible configuration for M is $2^{O(m)}$.