Quantified Boolean Formulas

Note for IAIP Fall 2005
Rune M. Jensen
IT University of Copenhagen

August 11, 2005

Quantified Boolean Formulas (QBF) [1] provides a concise notation for complex operations on Boolean formulas which we will use extensively to define BDD operations. QBF is ordinary propositional logic extended with quantification of Boolean variables.

Definition 1 (QBF syntax) *Given a set $V = \{v_1, \cdots, v_n\}$ of propositional variables, QBF(V) formulas are inductively defined by*

- *every variable in V is a formula,*
- *if f and g are formulas, then so are $\neg f$, $f \wedge g$, and $f \vee g$, and*
- *if f is a formula and $v \in V$, then $\exists v \,.\, f$ and $\forall v \,.\, f$ are formulas.*

A *truth assignment* for QBF(V) is a function $\sigma : V \rightarrow \mathbb{B}$. We will use the notation $\sigma\langle v \leftarrow a\rangle$ for the truth assignment defined by

$$\sigma\langle v \leftarrow a\rangle(w) = \begin{cases} a & : \text{ if } v = w \\ \sigma(w) & : \text{ otherwise.} \end{cases}$$

Definition 2 (QBF Semantics) *If f is a formula in QBF(V) and σ is a truth assignment, we will write $\sigma \models f$ to denote that f is true under the assignment σ. The relation $\models$ is defined inductively in the obvious manner*

- $\sigma \models v$ *iff* $\sigma(v) = true$,
- $\sigma \models \neg f$ *iff* $\sigma \not\models f$,
- $\sigma \models f \vee g$ *iff* $\sigma \models f$ *or* $\sigma \models g$,
- $\sigma \models f \wedge g$ *iff* $\sigma \models f$ *and* $\sigma \models g$,
- $\sigma \models \exists v \,.\, f$ *iff* $\sigma\langle v \leftarrow false\rangle \models f$ *or* $\sigma\langle v \leftarrow true\rangle \models f$,
- $\sigma \models \forall v \,.\, f$ *iff* $\sigma\langle v \leftarrow false\rangle \models f$ *and* $\sigma\langle v \leftarrow true\rangle \models f$.

For a vector $\vec{v} = (v_1, \cdots, v_m)$ of propositional variables in V, we define the abbreviations

$$\begin{aligned} \exists \vec{v} \,.\, f &\equiv \exists v_1 \,.\, (\cdots (\exists v_m \,.\, f) \cdots) & (1) \\ \forall \vec{v} \,.\, f &\equiv \forall v_1 \,.\, (\cdots (\forall v_m \,.\, f) \cdots). & (2) \end{aligned}$$

The *support* of a formula f is the set of variables that f depends on $\{v \in V \mid f|_{v \leftarrow \mathit{true}} \neq f|_{v \leftarrow \mathit{false}}\}$.

References

[1] A. V. Aho, J. E. Hopcroft, and J. D. Ullman. *The Design and Analysis of Computer Algorithms*. Addison Wesley, 1974.