Notes on Validated Model Counting

Randal E. Bryant

Computer Science Department
Carnegie Mellon University, Pittsburgh, PA, United States

1 Notation

Consider Boolean formulas over a set of variables X. An assignment σ is a function mapping each variable to truth value 1 (true) or 0 (false). We can extend σ to Boolean formulas in the normal way, such that $\sigma(F)$ will be 1 (respectively, 0) if the evaluation of formula F yields 1 (resp., 0) when its variables are assigned values according to σ.

For Boolean formula F, we define its set of *models* Σ_F as

$$\Sigma_F = \{\sigma | \sigma(F) = 1\} \tag{1}$$

The task of model counting is, given formula F, to determine the size of its set of models $\mu(F) = |\Sigma_F|$. Ideally, this should be done without actually enumerating the set.

For interpretation σ and a Boolean formula E over X, we use the notation $\sigma[E/x]$ to denote the intepretation σ', such that $\sigma'(y) = \sigma(y)$ for all $y \neq x$ and $\sigma'(x) = \sigma(E)$. In particular, the notation $\sigma[\overline{x}/x]$ indicates the interpretation in which the value assigned to x is complemented, while others remain unchanged.

A Boolean formula F is said to be *independent* of variable x if every $\sigma \in \Sigma_F$ has $\sigma[\overline{x}/x] \in \Sigma_F$.

Lemma 1. *If Boolean formula F is independent of variable x, then:*

$$|\Sigma_F| \;=\; 2 \cdot |\Sigma_{F \wedge x}| =\; 2 \cdot |\Sigma_{F \wedge \overline{x}}| \tag{2}$$

That is, the set of models of F is split evenly between those assigning 1 to x and those assigning 0 to x.

2 ITE Operation

We consider a single Boolean operation, known as "If-Then-Else," or simply "ITE." For Boolean values a, b, c, the operation is defined as $\mathit{ITE}(a, b, c) = (a \wedge b) \vee (\overline{a} \wedge c)$. This single operation can be used to express several common Boolean operations:

$$\begin{aligned}
\overline{a} &= \mathit{ITE}(a, 0, 1) \\
a \wedge b &= \mathit{ITE}(a, b, 0) \\
a \vee b &= \mathit{ITE}(a, 1, b) \\
a \rightarrow b &= \mathit{ITE}(a, b, 1)
\end{aligned}$$

For variable x and Boolean formulas F and G, the set of models for $\mathit{ITE}(x, F, G)$ is given by the formula:

$$\Sigma_{\mathit{ITE}(x,F,G)} = \{\sigma \in \Sigma_F | \sigma(x) = 1\} \cup \{\sigma \in \Sigma_G | \sigma(x) = 0\} \tag{3}$$

3 ITE Graphs

Model counting is especially simple for a class of formulas we call "free ITE graphs." This representation can be obtained directly from an Ordered Binary Decision Diagram (OBDD) representation of a Boolean function [1].

An *ITE graph* is defined to be a directed acyclic graph, consisting of three node types:

Constant: Corresponds to value 1 or 0. A constant node has no incoming arcs.
Input: Corresponds to one of the variables in X. A variable node has no incoming arcs.
Operator: Represents an application of the *ITE* operation. An operator node has three incoming arcs, labeled **I**, **T**, and **E**, corresponding to the three arguments of the *ITE* operation.

An *input-controlled* ITE graph is one for which the incoming **I** arc for every operator node is from an input node. That is, the result of one *ITE* operation can serve as the "then" or "else" argument to another *ITE* operation, but not for the "if" argument.

When describing input-controlled ITE graphs, we refer to constant and input nodes by their associated value or variable. An operator node is described by an expression of the form $\mathit{ITE}(x, v_t, v_e)$, where v_t and v_e are the nodes corresponding to the **T** and **E** inputs, respectively.

We can define the function D mapping each node v in an input-controlled ITE graph to the set of variables on which it logically depends. This can be expressed recursively as

$$D(1) = \emptyset \tag{4}$$
$$D(0) = \emptyset \tag{5}$$
$$D(x) = \{x\} \tag{6}$$
$$D(\mathit{ITE}(x, v_t, v_e)) = \{x\} \cup D(v_t) \cup D(v_e) \tag{7}$$

Lemma 2. *Any node v in an ITE graph is independent of any variable $y \in X$ such that $y \notin D(v)$.*

Proof: Follows by the recursive definition of D (4–7) and by (3).

A *free* ITE graph is an input-controlled ITE graph satisfying the added property that for every operator node $v = \mathit{ITE}(x, v_t, v_e)$, its incoming **T** and **E** arcs must be from nodes such that $x \notin D(v_t)$ and $x \notin D(v_e)$.

Free ITE graphs correspond directly to a class of binary decision diagrams known as *free* BDDs (FBDDs) [2]. Ordered BDDs are a subclass of FBDDs.

4 Model Counting Free ITE Graphs

While model counting is a difficult problem for arbitrary formulas, it readily be computed when F takes the form of a free ITE graph.

Theorem 1. *Letting* $n = |X|$*, the following recursive formula holds for any node in a free ITE graph*

$$\mu(1) = 2^n \tag{8}$$

$$\mu(0) = 0 \tag{9}$$

$$\mu(x) = 2^{n-1} \tag{10}$$

$$\mu(\mathit{ITE}(x, v_t, v_e)) = \frac{\mu(v_t) + \mu(v_e)}{2} \tag{11}$$

Proof: The key result here is (11). It follows from Lemma 1, (3), Lemma 2, and (7). That is, both v_t and v_e are independent of x. That implies that half of the satisfying assignments σ for v_t have have $\sigma(x) = 1$. Similarly, half of the satisfying assignments σ for v_e have $\sigma(x) = 0$. Equation (11) then follows by (3).

References

1. Bryant, R.E.: Graph-based algorithms for Boolean function manipulation. IEEE Trans. Computers **35**(8), 677–691 (1986)
2. Gergov, J., Meinel, C.: Efficient Boolean manipulation with OBDD's can be extended to FBDD's. IEEE Transactions on Computers **43**(10), 1197–1209 (October 1994)