The Tautologies over a finite set are context-free

Michael Bertol
Universität Stuttgart, Institut für Informatik
Breitwiesenstraße 20/22, D-70565 Stuttgart

Klaus Reinhardt
Eberhard-Karls-Universität Tübingen
Wilhelm-Schickhard Institut für Informatik
Sand 13, D-72076 Tübingen

Abstract

In this note we show, that the set of valid boolean formulae over a finite set of variables is a natural example for a context-free language.

Context-free languages are very important in computer science. Thus it would be nice to enlarge the collection of natural and attractive examples. One interesting example is obtained by the question whether the set of tautologies over a fixed finite set of variables is context-free[1]. We show that the answer of this question is positive. The proof of this fact is very simple. However, up to our knowledge, this fact is not available in the (standard) literature. Therefore this note might be of interest.

Let $\mathcal{F}_n$ be the set of boolean formulae containing at most the variables $x_1, \ldots, x_n$ and $\mathcal{A}_n$ the set of all assignments $\alpha : \{x_1, \ldots, x_n\} \to \{0,1\}$. The set of tautologies is $\mathcal{T}_n = \{\varphi \in \mathcal{F}_n \mid \forall \alpha \in \mathcal{A}_n : \alpha(\varphi) = 1\}$.

Let $\mathcal{G}_n = (N_n, T_n, P_n, S)$ be the context-free grammar where each subset of $A \subseteq \mathcal{A}_n$ is identified with a nonterminal $V_A \in N_n$; the variables $x_1, \ldots, x_n$, the brackets and the logical junctors $\neg$ and $\vee$ are the terminal symbols, and $V_{\mathcal{A}_n}$ is the axiom S. The set of productions P_n consists of

$$V_A \longrightarrow \neg(V_{\mathcal{A}_n \setminus A}) \tag{1}$$

$$V_{A \cup B} \longrightarrow V_A \vee V_B \tag{2}$$

$$V_{\{\alpha \mid \alpha(x)=1\}} \longrightarrow x \tag{3}$$

for $A, B \in N_n$ and $x \in \{x_1, \ldots, x_n\}$. (Note that each formula defines a set of rules.)

[1]This problem has been raised by W. Brauer, and communicated to us by M. Jantzen.

It remains to prove, that $L(\mathcal{G}_n) = \mathcal{T}_n$. We show that $V_A \stackrel{*}{\Rightarrow} \varphi$ iff $A = \{\alpha \mid \alpha(\varphi) = 1\}$.

(i) Let $V_A \stackrel{*}{\Rightarrow} \varphi$. If $\varphi = x$, for some $x \in \{x_1, \ldots, x_n\}$, then x can only be derived by (3). If $V_A \Rightarrow \neg(V_B) \stackrel{*}{\Rightarrow} \varphi$, then $\varphi = \neg(\psi)$ where by induction, $B = \{\beta \mid \beta(\psi) = 1\}$ and therefore, by (1) $V_B \stackrel{*}{\Rightarrow} \Psi$; $A = (\mathcal{A}_n \setminus B) = \{\alpha \mid \alpha(\neg(\psi)) = 1\}$. Else, if $V_A \Rightarrow V_B \vee V_C \stackrel{*}{\Rightarrow} \varphi$ then $\varphi = \psi \vee \theta$ and $A = B \cup C$; by (2), $B = \{\beta \mid \beta(\psi) = 1\}$, $C = \{\gamma \mid \gamma(\theta) = 1\}$, hence $A = \{\alpha \mid \alpha(\psi \vee \theta) = 1\}$.

(ii) It remains to show, that $V_{\{\alpha|\alpha(\varphi)=1\}} \stackrel{*}{\Rightarrow} \varphi$. If $\varphi = x$ for some $x \in \{x_1, \ldots, x_n\}$, then with (3), $V_{\{\alpha|\alpha(\varphi)=1\}} \Rightarrow \varphi$. If $\varphi = \neg(\psi)$, by induction $V_{\{\beta|\beta(\psi)=1\}} \stackrel{*}{\Rightarrow} \psi$, hence by (1), $V_{\{\alpha|\alpha(\neg(\psi))=1\}} \Rightarrow \neg(V_{\{\beta|\beta(\psi)=1\}}) \stackrel{*}{\Rightarrow} \neg(\psi)$. Else, if $\varphi = \psi \vee \theta$, again by induction, $V_{\{\beta|\beta(\psi)=1\}} \stackrel{*}{\Rightarrow} \psi$ and $V_{\{\gamma|\gamma(\theta)=1\}} \stackrel{*}{\Rightarrow} \theta$, hence, by (2) and the fact that $\{\alpha \mid \alpha(\psi \vee \theta) = 1\} = \{\beta \mid \beta(\psi) = 1\} \cup \{\gamma \mid \gamma(\theta) = 1\}$, this case is proven.

As a consequence, $S \stackrel{*}{\Rightarrow} \varphi$ iff $\varphi \in \mathcal{T}_n$.

Remarks: It suffices to consider only the junctor base $\neg, \vee$, since analogously one can consider all (binary) junctors.

As an immediate consequence we get that the set of all satisfiable formulae over the set of variables $\{x_1, \ldots, x_n\}$ is context-free, too. We only need to take as axioms all non-empty subsets of $\mathcal{A}_n$.

The resulting grammar $\mathcal{G}_n$ is of course ambiguous, but the language $\mathcal{T}_n$ is deterministic context-free.

These results are a consequence of the well known (more general) principle that one can simulate the evaluation for a s-attributed context-free grammar (only synthesized attributes) with finite attribute domains by an (unattributed) context-free grammar since the term-algebra over a finite signature (set of well-formed terms) is context-free.

References

[1] A.V. Aho, R. Sethi, J.D. Ullman. *Compilers, Principles, Techniques, and Tools.* Addison-Wessley Publishing Company, 1986.

[2] J.E. Hopcroft, J.D. Ullman. *Introduction to Automata Theory, Languages, and Computation.* Addison-Wessley Publishing Company, 1979.

[3] U. Schöning. *Logik für Informatiker.* BI Wissenschaftsverlag, 1987.