What do Melons and Lemons Have in Common?

Extending Parikh's Theorem to the Weighted Case

Elena Gutiérrez*
IMDEA Software Institute
Madrid, Spain
elena.gutierrez@imdea.org

ABSTRACT

When the ordering of symbols in the words is ignored (and thus "melons" and "lemons" are equivalent words), the celebrated Parikh's Theorem states that every context-free language is equivalent to a regular language. Now we consider languages as sets of words where additionally each word is augmented with a weight. Can we extend Parikh's Theorem to this more general setting?

CCS CONCEPTS

• **Theory of computation → Grammars and context-free languages**;

KEYWORDS

Weighted Context-Free Grammars, Algebraic Language Theory, Parikh Image

ACM Reference Format:
Elena Gutiérrez. 2018. What do Melons and Lemons Have in Common?: Extending Parikh's Theorem to the Weighted Case. In *Proceedings of ACM Celebration of Women in Computer Science (womENcourage 2018).* ACM, New York, NY, USA, Article 4, 1 page. https://doi.org/10.475/123_4

1 INTRODUCTION

In the context of program analysis, *context-free grammars* (CFGs) are natural models for sequential programs with recursive procedures. They consist of a finite set of rules that describe the syntax of the program. CFGs generate the class of the so-called *context-free languages* (CFLs). As a subclass of CFLs we find the *regular* languages which are generated by the so-called *regular* CFGs and, at the loss of expressive power, are much simpler. For instance, it is unsolvable whether the intersection of two general CFLs is empty or not while if we deal with regular CFLs the same question is easy to solve. The celebrated Parikh's Theorem [2] establishes that every CFL is equivalent to a regular CFL when the ordering of symbols is ignored. For instance, the CFL $L = \{a^n b^n \mid n \geq 0\}$ is equivalent to the regular CFL $L' = (ab)^*$ modulo ordering of symbols in the words. Note that each word of the form $a^n b^n$ in L is equivalent to $(ab)^n$ in L' when the ordering of a's and b's is ignored. This result was applied to the analysis of asynchronous programs with procedures [4] where a method call may be postponed by posting a callback in a task buffer and executing it later. This means that one can relax about the ordering in which tasks are served during the execution of the program and use Parikh's Theorem to give a "simpler" asynchronous program without procedures that preserves safety properties.

Now I am interested in generalizing this result for the so-called *weighted CFGs* (WCFGs), where each rule in the CFG has an associated weight. More precisely, the question I study is: given a WCFG, is there always an equivalent regular WCFG modulo ordering of symbols, and such that the weights are also preserved[1]?.

Extending Parikh's Theorem to the weighted case has the potential of enabling richer analysis of event-driven asynchronous programs where each event has a certain probability to occur. By assigning these probabilities to the rules of the CFG, we can perform a probabilistic analysis of programs following this paradigm. Other measurements can be attached to the grammar rules such as energy cost, time of execution, ...

2 OUR CONTRIBUTION TO THE QUESTION

It turns out that the answer to the previous question is, in general, negative. [3]. However, it is well-known that the result is true when the weight domain has the algebraic structure of a *semiring* satisfying two properties: *commutativity* and *idempotence* [1]. In our work we show yet another condition, this time imposed on the derivations of the grammar. Namely, we prove that every *nonexpansive* grammar over any arbitrary weight domain satisfies the theorem. However, these conditions are sufficient but not necessary for the result, so a natural question arises: can we decide whether or not Parikh's Theorem holds in the weighted case? We show an algorithm that solves the question when the weights are defined over the rational numbers. Intuitively, given a WCFG G, the algorithm transforms G by means of algebraic methods into another WCFG G' that produces an equivalent language modulo ordering of symbols preserving the weights; and checks whether G' is regular or not. The algorithm is complete: if G' is not regular, then no regular WCFG with the desired property exists.

REFERENCES

[1] Werner Kuich. 1987. The Kleene and the Parikh Theorem in Complete Semirings. In *ICALP87, Karlsruhe, Germany, July 13-17, 1987, Proceedings.* 212–225.
[2] Rohit Parikh. 1966. On Context-Free Languages. *J. ACM* 13, 4 (1966), 570–581.
[3] Ion Petre. 1999. Parikh's Theorem Does Not Hold for Multiplicities. *Journal of Automata, Languages and Combinatorics* 4, 1 (1999), 17–30.
[4] Koushik Sen and Mahesh Viswanathan. 2006. Model Checking Multithreaded Programs with Asynchronous Atomic Methods. In *CAV 2006, Seattle, WA, USA, August 17-20, 2006, Proceedings.* 300–314.

*The author is also affiliated to Universidad Politécnica de Madrid, Spain

[1] Namely, the sum of the weights of the words that have the same number of occurrences of each symbol must be equal in both languages.

womENcourage 2018, October 2018, Belgrade, Serbia

ACM ISBN 123-4567-24-567/08/06.
https://doi.org/10.475/123_4