Contents lists available at ScienceDirect
Information and Computation
www.elsevier.com/locate/yinco

On the compressibility of finite languages and formal proofs ☆

Sebastian Eberhard, Stefan Hetzl *

Institute of Discrete Mathematics and Geometry, Vienna University of Technology, Wiedner Hauptstraße 8-10, 1040 Wien, Austria

ARTICLE INFO

Article history:
Received 15 December 2015
Available online xxxx

ABSTRACT

We consider the minimal number of productions needed for a grammar to cover a finite language L as the grammatical complexity of L. We study this measure for several types of word and tree grammars and show that it is closely connected to well-known measures for the complexity of formal proofs in first-order predicate logic.
We construct an incompressible sequence of finite word languages and transfer this and several other results about the complexity of word and tree languages to formal proofs.

1. Introduction

In grammar-based compression, context-free grammars that generate exactly one word are used for representing the input text. The smallest grammar problem asks for the smallest context-free grammar that generates a given word. Its decision version is known to be NP-complete [1], see [2] for the case of a bounded alphabet. However, there is a number of fast algorithms which are practically useful [3–5] or achieve good approximation ratios [6–10]. Grammar-based compression also has the considerable practical advantage that many operations can be performed directly on the compressed representation; see [11].

We are interested in the problem of simultaneously compressing a finite set of words by a single grammar. Traditionally, the grammatical complexity of a finite language L is defined as the minimal number of productions of a grammar G with $L(G) = L$. Each class of grammars thus gives rise to a measure of descriptional complexity. The study of the grammatical complexity of finite languages has been initiated in [12] and continued in [13–16].

Our motivation for investigating this problem is rooted in proof theory and automated deduction: as shown in [17], there is an intimate relationship between a certain class of formal proofs (those with Π_1-cuts) in first-order predicate logic and a certain class of grammars (totally rigid acyclic tree grammars). In particular, the number of production rules in the grammar characterises the number of certain inference rules in the proof. This relationship has been exploited in a method for proof compression whose central combinatorial step is a grammar-based compression of a finite tree language [18–20].

The proof-theoretic application of our work requires a modification of the traditional problem: we are looking for a minimal grammar G s.t. $L(G) \supseteq L$ where L is the finite input language. This is the case because L describes a disjunction which is required to be a tautology (a so-called Herbrand-disjunction, see [21,22]) and if $L' \supseteq L$, then L' also describes a tautology. This condition is similar to (but different from) the one imposed on cover automata [23]: there an automaton A is sought s.t. $L(A) \supseteq L$, but in addition it is required that $L(A) \setminus L$ consists only of words longer than any word in L.

In this paper we consider the minimal number of productions needed for a grammar to cover a finite language L as the grammatical complexity of L. We study this measure for several types of word and tree grammars and show that

☆ Research supported by the Vienna Science Fund (WWTF) project VRG12-004.

* Corresponding author.
E-mail addresses: sebastian.eberhard84@gmail.com (S. Eberhard), stefan.hetzl@tuwien.ac.at (S. Hetzl).

http://dx.doi.org/10.1016/j.ic.2017.09.001
0890-5401/© 2017 Elsevier Inc. All rights reserved.

it is strongly related to well-known measures for the complexity of formal proofs. The central technical results are the construction of an incompressible sequence of finite (word) languages and its use for obtaining a lower bound on the complexity of proofs with $\overline{\Pi}_1$-cuts in terms of the complexity of the shortest cut-free proof. The interest in such a result is motivated by the experience that the length of proofs with cuts is notoriously difficult to control (for propositional logic this is considered the central open problem in proof complexity [24]).

This paper extends [25] in the following respects: we prove the lower bound on an enlarged class of proofs: instead of proofs with Π_1-cuts (i.e., lemmas of the form $\forall x\, A$ for A quantifier-free) we treat proofs with $\overline{\Pi}_1$-cuts here (i.e., lemmas of the form $\forall x_1 \cdots \forall x_n\, A$ for A quantifier-free). This necessitates the introduction of a more general class of tree grammars: vectorial totally rigid acyclic tree grammars. In this paper, we also carry out a thorough investigation of the various types of tree grammars involved, including several results on the relative complexity of them. In contrast to [25], this paper also contains an introduction to the proof-theoretic background and complete proofs of the proof-theoretic results.

In Section 2 we introduce some basic notions concerning the grammatical complexity of finite languages. In Section 3 we construct an incompressible sequence of word languages. In Section 4 we study tree grammars of proof-theoretic relevance. We investigate their relationship to each other and to the word grammars of Section 3. In Section 5 we introduce the basic notions and results of proof theory which are relevant to this paper. In Section 6 we establish the relationship between the complexity of formal proofs and the grammatical complexity of finite languages. Finally, in Section 7, we transfer several results about grammatical complexity, including the incompressibility-result, to proof theory.

2. Grammatical complexity of finite languages

Definition 1. A *context-free grammar (CFG)* is a 4-tuple $G = (N, \Sigma, P, S)$ where N is a finite set of nonterminals, Σ is a finite alphabet, $S \in N$ is the starting symbol and P is a finite set of productions of the form $A \to w$ where $A \in N$ and $w \in (\Sigma \cup N)^*$.

As usual, the one-step derivation relation $\Longrightarrow_G$ of G is defined by $u \Longrightarrow_G v$ iff there is a production $A \to w$ in G s.t. v is obtained from u by replacing an occurrence of A by w. The derivation relation $\Longrightarrow_G^*$ is the reflexive and transitive closure of $\Longrightarrow_G$ and the language of G is $L(G) = \{w \in \Sigma^* \mid S \Longrightarrow_G^* w\}$. We omit the subscript G if the grammar is clear from the context.

Definition 2. A *right-linear grammar* is a context-free grammar (N, Σ, P, S) s.t. all productions in P are of the form $A \to vB$ or $A \to v$ for $A, B \in N$ and $v \in \Sigma^*$.

It is well-known, see e.g., [26], that the languages generated by right-linear grammars are exactly the regular languages.

Definition 3. Let $G = (N, \Sigma, P, S)$ be a context-free grammar. The relation $<_G^1$ on N is defined as follows: $A <_G^1 B$ iff there is a production $A \to w$ in P s.t. B occurs in w. The relation $<_G$ is defined as the transitive closure of $<_G^1$. We say that G is cyclic (respectively acyclic) iff $<_G$ is.

We abbreviate "right-linear acyclic grammar" as "RLAG". Let $A \in N$; then a production whose left hand side is A is called A-production. We write P_A for the set of A-productions in P. For $N' \subseteq N$ we define $P_{N'} = \bigcup_{A \in N'} P_A$. For a language L and a CFG G we say that G covers L if $L(G) \supseteq L$. The size of a CFG $G = (N, \Sigma, P, S)$ is defined as $|G| = |P|$. The length of a right-linear production rule $A \to wB$ or $A \to w$ for $w \in \Sigma^*$ is defined as $|w|$.

Definition 4. The *RLA-complexity* of a finite language L is defined as $\mathrm{RLAc}(L) = \min\{|G| \mid G \text{ RLA s.t. } L(G) \supseteq L\}$. A finite language L is called *RLA-compressible* if $\mathrm{RLAc}(L) < |L|$ and *RLA-incompressible* otherwise.

Note that $\mathrm{RLAc}(L) \leq |L|$ for all finite languages L since L can be generated by a trivial grammar with $|L|$ production rules. All descriptional complexity measures in this paper will be written as $\mathrm{Xc}(\cdot)$ for some formalism X, e.g., $X = \mathrm{RLA}$ as above.

Definition 5. A sequence $(L_n)_{n\geq 1}$ of finite languages is called *RLA-incompressible* if there is an $M \in \mathbb{N}$ s.t. for all $n \geq M$ the language L_n is RLA-incompressible. A sequence $(L_n)_{n\geq 1}$ of finite languages is called *RLA-compressible* if for every $M \in \mathbb{N}$ there is an $n \geq M$ s.t. L_n is RLA-compressible.

We will use the above definition of X-compressibility of a sequence based on the X-compressibility of an element of the sequence also for descriptional complexity measures other than $X = \mathrm{RLA}$.

A variant of our measure of grammatical complexity, the equality formulation, consists in asking for a minimal grammar G with $L(G) = L$. As explained in the introduction, the cover formulation is motivated by our proof-theoretic application; see Section 6. However, the main result on incompressibility also applies to the equality formulation; see Corollary 1. Incompressible finite languages in the sense of the equality formulation have been studied before: [13] considers the sequence