Compressibility of finite languages by grammars

Stefan Hetzl

Institute of Discrete Mathematics and Geometry
Vienna University of Technology

joint work with Sebastian Eberhard

Descriptional Complexity of Formal Systems (DCFS) 2015
Waterloo, Ontario, Canada

June 26, 2015

Introduction

- Grammar based compression
- Smallest grammar problem
  (compression of a single word by a CFG)
- This talk: compression of a finite language by a grammar
  incompressible sequence of finite languages
- Motivation: application in proof theory

Outline

- The smallest grammar problem(s)
- Incompressible languages
- Trees and proofs

The smallest grammar problem

- **Problem.**
  Given $w \in \Sigma^*$, find minimal CFG G with $\mathsf{L}(G) = \{w\}$
  here: minimal w.r.t. sum of lengths of rhs of production rules
- **Decision Problem.**
  Given $w \in \Sigma^*$ and $k \in \mathbb{N}$, is there a CFG G with $\mathsf{L}(G) = \{w\}$ and $\mathrm{size}(G) \leq k$?
- Decision problem **NP**-complete [Storer, Szymanski '78]
- Approximation: linear-time algorithms with logarithmic approximation ratio
  [Charikar et al. '02], [Rytter '03], [Sakomoto '05], [Charikar et al. '05], [Jeż '13], [Jeż '14]
- Practically efficient approximation algorithms
  - Sequitur [Nevill-Manning, Witten '97]
  - Re-Pair [Larsson, Moffat '99]

Our variant of the smallest grammar problem

- A grammar $G = (N, \Sigma, P, S)$ is called *right-linear* if all productions are of the form $A \to wB$ or $A \to w$ for $w \in \Sigma^*$.
- **Definition.** $A <^1_G B$ if there is $A \to u \in P$ s.t. B occurs in u. Define $<_G$ as transitive closure of $<^1_G$.
- **Definition.** RLAG: right-linear acyclic grammar
- **Problem.**
  Given finite $L \subseteq \Sigma^*$, find minimal RLAG G with $\mathrm{L}(G) \supseteq L$.
  here: minimal w.r.t. number of production rules
- $|G|$ is number of production rules

Many smallest grammar problems

- **Problem (traditional).**
  Given $w \in \Sigma^*$, find minimal CFG G with $\mathrm{L}(G) = \{w\}$
  here: minimal w.r.t. sum of lengths of rhs of production rules
- **Problem (this talk).**
  Given finite $L \subseteq \Sigma^*$, find minimal RLAG G with $\mathrm{L}(G) \supseteq L$.
  here: minimal w.r.t. number of production rules

Many smallest grammar problems

- **Problem (traditional).**
  Given $w \in \Sigma^*$, find minimal CFG G with $\mathrm{L}(G) = \{w\}$
  here: minimal w.r.t. sum of lengths of rhs of production rules
- **Problem (this talk).**
  Given finite $L \subseteq \Sigma^*$, find minimal RLAG G with $\mathrm{L}(G) \supseteq L$.
  here: minimal w.r.t. number of production rules

- Many smallest grammar problems:
  - RLAG / ACFG / TRATG / ...
  - Size / number of production rules / ...
  - $\mathrm{L}(G) \supseteq L$, $\mathrm{L}(G) = L$

- Compression of a *finite language*
- Emphasis on formalism for compression
- Operations on compressed representation

Outline

- ✓ The smallest grammar problem(s)
- Incompressible languages
- Trees and proofs

Incompressibility

- **Definition.** Finite L is called incompressible if every RLAG G with $\mathrm{L}(G) \supseteq L$ satisfies $|G| \geq |L|$.
- **Definition.** A sequence $(L_n)_{n \geq 1}$ is called incompressible if there is an $M \in \mathbb{N}$ s.t. for all $n \geq M$ the language L_n is incompressible.

Incompressibility

- **Definition.** Finite L is called incompressible if every RLAG G with $\mathrm{L}(G) \supseteq L$ satisfies $|G| \geq |L|$.
- **Definition.** A sequence $(L_n)_{n \geq 1}$ is called incompressible if there is an $M \in \mathbb{N}$ s.t. for all $n \geq M$ the language L_n is incompressible.
- $L_n = \{a\}$ is incompressible.
- $L_n = \{a_1, \ldots, a_n\}$ is incompressible.
- Is there incompressible $(L_n)_{n \geq 1}$ s.t.
  - alphabet is finite and
  - $|L_n|$ is unbounded ?

Incompressible languages

- $\Sigma = \{\mathbf{0}, \mathbf{1}, \mathbf{s}\}$
- Write $\mathrm{b}_l(i) \in \{\mathbf{0}, \mathbf{1}\}^l$ for l-bit binary representation of i.
- For $n \geq 1$ define

$$l(n) = \lceil \log_2(n) \rceil$$
$$k(n) = \lceil \frac{9n}{l(n) + 1} \rceil$$
$$L_n = \{(\mathbf{s}\mathrm{b}_{l(n)}(i))^{k(n)} \mid 0 \leq i \leq n - 1\}$$

- $|L_n| = n$
- Length of all $w \in L_n$ is $O(n)$

Incompressible languages – Example

For $n = 10$ we have $l(n) = 4$ and $k(n) = 18$ and $L_n =$

$$\begin{array}{ccc}
\mathbf{s0000s0000} & \cdots & \mathbf{s0000} \\
\mathbf{s0001s0001} & \cdots & \mathbf{s0001} \\
\vdots & \vdots & \vdots \\
\mathbf{s1001s1001} & \cdots & s\mathbf{1001}
\end{array}$$

Definition. Building block, segment.

Incompressible languages – Result

Theorem. $(L_n)_{n\geq 1}$ is incompressible.

Proof Sketch.

1. W.r.t. compressibility: reduced RLAGs enough
2. Reduced RLAG that covers L_n has only short productions
3. Short productions cannot cover many segments
4. Compressing grammar must cover many segments per production

3 and 4 contradict.

- **Corollary.** There is no sequence $(G_n)_{n\geq 1}$ of RLAGs and $M \in \mathbb{N}$ s.t. $\mathrm{L}(G_n) = L_n$ and $|G_n| < |L_n|$ for all $n \geq M$.
- **Theorem.** There is a sequence $(G_n)_{n\geq 1}$ of acyclic CFGs which compresses $(L_n)_{n\geq 1}$.

  Proof. Let P_n be

$$\begin{aligned} S &\to (\mathbf{s}A_1)^{k(n)}, \\ A_1 &\to \mathbf{0}A_2 \mid \mathbf{1}A_2, \\ &\vdots \\ A_{l(n)} &\to \mathbf{0} \mid \mathbf{1}. \end{aligned}$$

  Then $|P_n| = 2\lceil \log(n) \rceil + 1 < n = |L_n|$.

Outline

- ✓ The smallest grammar problem(s)
- ✓ Incompressible languages
- ▶ Trees and proofs

TRAT grammars

- Rigid tree languages [Jacquemard, Clay, Vacher '09]
- **Definition.** A *regular tree grammar* is a tuple (N, Σ, P, S) s.t. all productions are of the form $A \to t$ with $t \in \mathrm{T}(\Sigma \cup N)$.
- **Definition.** $<_G$ on N as for word grammars.
- **Definition.** A derivation $S \Longrightarrow_G^* t$ satisfies *rigidity condition* if it uses at most one A-production for every nonterminal A.
- **Definition.** A totally rigid acyclic tree (TRAT) grammar is an acyclic regular tree grammar $G = (N, \Sigma, P, S)$. Define $\mathrm{L}(G) = \{t \in \mathrm{T}(\Sigma) \mid S \Longrightarrow_G^* t \text{ satisfying rigidity condition}\}$.
- *Example.* $S \to f(A, B),\ A \to g(B),\ B \to c \mid d$
  as regular tree grammar:
  $$L = \{f(g(c), c), f(g(c), d), f(g(d), c), f(g(d), d)\}$$
  as TRATG:
  $$L = \{f(g(c), c), f(g(d), d)\}$$

From word languages to tree languages

- For alphabet Σ define $\Sigma^{\mathsf{T}} = \{f_x \mid x \in \Sigma\} \cup \{e\}$
- Map words to trees, e.g.: $(abaac)^{\mathsf{T}} = f_a(f_b(f_a(f_a(f_c(e)))))$
- $\cdot^{\mathsf{T}}$ maps RLAG to TRATG
- **Lemma.** If L is RLA-incompressible, then L^{T} is TRAT-incompressible.
- **Corollary.** $(L_n^{\mathsf{T}})_{n \geq 1}$ is TRAT-incompressible.

A corollary in proof theory

- Inference rule "cut": use of a lemma in a proof
- **Theorem** [H '12].

cut-free proof	...	trivial tree grammar: tree language
proof with Π_1-cuts	...	(non-trivial) TRAT grammar

- Cut-elimination gives trivial bounds on compressibility

  $\Rightarrow$ Π_1-compression: exponential

- We construct formulas ψ_n in first-order predicate logic s.t.
  - cut-free complexity $O((2^n)^2)$
  - Π_1-cut complexity 2^n

  $\Rightarrow$ only quadratic

Conclusion

- Sequence of incompressible languages
- Compressing finite languages is interesting

Open Questions / Future Work

- Complexity of smallest grammar problem(s) for finite languages

  We know: Decision problem for TRATG(2), number of production rules, $\mathrm{L}(G) \supseteq L$ is **NP**-complete.

- Approximation ratios?
- Practically efficient algorithms?