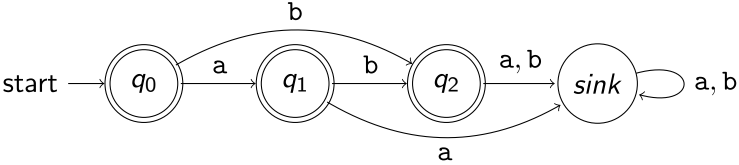
On the state complexity of subword closed languages

Jérôme Guyot

ENS Paris-Saclay - Université Paris-Saclay

Contents

Introduction



This is the canonical automaton for $L = \{\epsilon, \mathtt{a}, \mathtt{b}, \mathtt{ab}\}$.

How large is the canonical automaton for $L^{a \leftarrow L}$?

Motivations

- Finite state automaton are often used as data structure
- Their size is important in the complexity analysis.
- In verification of well structure systems, some algorithms use words ordered by the subword ordering.

Background

Introduction

Background

State complexity

Subword closed languages

Substitution of subword closed languages

Substitution : general / singular

Computing quotients

Bounding the state complexity

State complexity of a regular language

Definition : State complexity

The *state complexity* of L noted $sc(L)$ is the number of states in the canonical automaton for L.

- Easy to visualize but is not practical for formal proofs.

State complexity of a regular language

Definition : State complexity

The *state complexity* of L noted $sc(L)$ is the number of states in the canonical automaton for L.

- Easy to visualize but is not practical for formal proofs.

Definition : Left quotients and quotient complexity

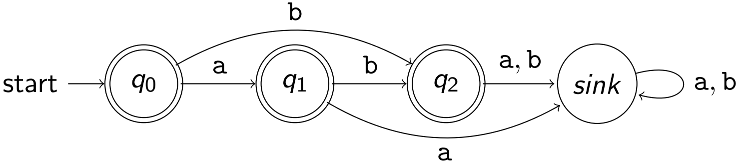
- $L/u = \{v \text{ such that } uv \in L\}$.
- $\mathcal{R}(L)$ is the set of all left quotients of L.
- $\kappa(L) = |\mathcal{R}(L)|$ is the *quotient complexity* of L.

Summary : $\kappa(L)$ is the number of different left quotients of L.

State complexity of a regular language II

Example

Take $L = \{\epsilon, \mathtt{a}, \mathtt{b}, \mathtt{ab}\}$

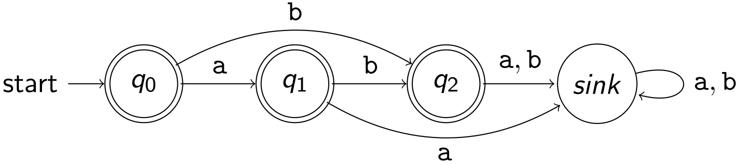


- $L/\epsilon = L$, $L/\mathtt{a} = \{\epsilon, \mathtt{b}\}$, $L/\mathtt{b} = \{\epsilon\}$, $L/\mathtt{aba} = \emptyset$
- $|\mathcal{R}(L)| = \kappa(L) = 4$

State complexity of a regular language II

Example

Take $L = \{\epsilon, \mathtt{a}, \mathtt{b}, \mathtt{ab}\}$



- $L/\epsilon = L, \;\; L/\mathtt{a} = \{\epsilon, \mathtt{b}\}, \;\; L/\mathtt{b} = \{\epsilon\}, \;\; L/\mathtt{aba} = \emptyset$
- $|\mathcal{R}(L)| = \kappa(L) = 4$

Theorem (Myhill, Nerode 1957) (Brzozowski 2009)

$$\kappa(L) = sc(L)$$

State complexity of a function

- Take $f : Reg \longrightarrow Reg$
- How does $\kappa(f(L))$ relate to $\kappa(L)$?

Definition

The *state complexity* of f is the function $\phi_f : \mathbb{N} \longrightarrow \mathbb{N}$ such that

$$\phi_f(n) = \max_{\kappa(L) \leq n} \kappa(f(L))$$

State complexity of a function

- Take $f : Reg \longrightarrow Reg$
- How does $\kappa(f(L))$ relate to $\kappa(L)$?

Definition

The *state complexity* of f is the function $\phi_f : \mathbb{N} \longrightarrow \mathbb{N}$ such that

$$\phi_f(n) = \max_{\kappa(L) \leq n} \kappa(f(L))$$

Examples

$$\phi_\cap(n) = \max_{\kappa(L_1), \kappa(L_2) \leq n} \kappa(L_1 \cap L_2) = n^2$$

$$\phi_{\text{complement}}(n) = \max_{\kappa(L) \leq n} \kappa(\bar{L}) = n$$

$$\phi_{\text{mirror}}(n) = \max_{\kappa(L) \leq n} \kappa(L^R) = 2^n$$

Can we compute or at least bound ϕ_f ?

Subword closed languages

Definition : Subwords

x *subword* of y, noted $x \preccurlyeq y$ iff
$y = u_0 \underline{x_1} u_1 \cdots u_{n-1} \underline{x_n} u_n$ with $\underline{x} = \underline{x_1 \cdots x_n}$ and $\forall i \ u_i \in \Sigma^*$

Example

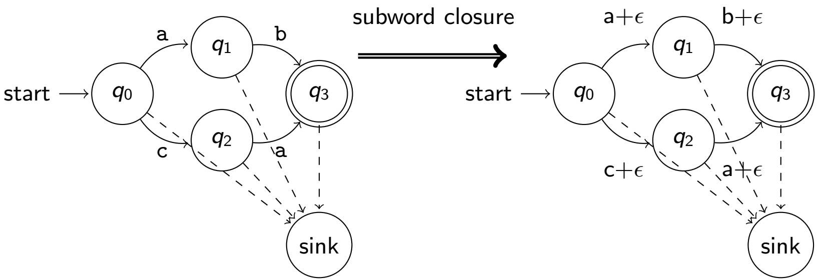
$\underline{\texttt{aab}} \preccurlyeq \underline{\texttt{a}}\texttt{bb}\underline{\texttt{a}}\texttt{a}\underline{\texttt{b}}\texttt{a}$

Definition : Subword closure

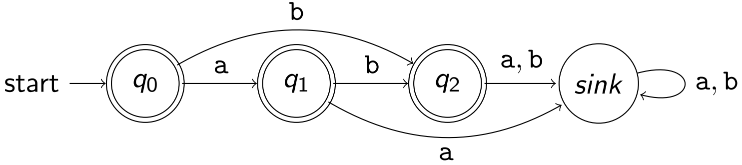
$\downarrow(L) = \{x \mid \exists y \in L, x \preccurlyeq y\}$
L is *subword closed* iff $\downarrow(L) = L$.

L is subword closed if any subword of a word of L is in L.

Subword closed languages II



Automaton of $\downarrow(\{ab\})$



State of the art

State complexity for subword-closed languages
(J. Brzozowski et al. 2014) (Hospodár 2019)

Operation	Upper Bound	Tightness requirement
$L \cap K$	$mn - (m + n - 2)$	$\|\Sigma\| \geq 2$
$L \cup K$	mn	$\|\Sigma\| \geq 4$
$L \setminus K$	$mn - (n - 1)$	$\|\Sigma\| \geq 4$
$L \oplus K$	mn	$\|\Sigma\| \geq 2$
$L \cdot K$	$m + n - 1$	$\|\Sigma\| \geq 2$
L^* (and L^+)	2	$\|\Sigma\| \geq 2$
L^R	$2^{n-2} + 1$	$\|\Sigma\| \geq 2n$
L^k	$k(n - 1) + 1$	$\|\Sigma\| \geq 2$

Substitution of subword closed languages

Substitutions

Definition

Let $S = (K_1, \ldots, K_n)$ a set of languages and $\Sigma = \{a_1, \ldots, a_n\}$ an alphabet, we define ρ such that

$$\rho(\{\epsilon\}) = \{\epsilon\}$$
$$\rho(\{a_i\}) = K_i$$
$$\rho(L_1 \cdot L_2) = \rho(L_2) \cdot \rho(L_2)$$
$$\rho(L_1 + L_2) = \rho(L_1) + \rho(L_2)$$

- We can also write $L^{a_1 \leftarrow K_1, \ldots, a_n \leftarrow K_n}$.

Definition : Singular substitution

If $\rho(\{a_i\}) = \begin{cases} K \text{ if } a_i = a \\ \{a_i\} \text{ otherwise} \end{cases}$

Questions

Proposition

If $K_1, \ldots K_n$ are subword closed, then $\rho(L)$ is subword closed.

Open problems :

$\phi_{\mathsf{sub}} \leq n^{\mathcal{O}(1)}$ for subword closed languages ?

$\kappa(L^{a \leftarrow K}) \leq \kappa(L)\kappa(K)$ for subword closed languages ?

Toy example

Take $L = \{\epsilon, a_1, \ldots, a_n\}$, and $K_i = (\Sigma \backslash \{a_i\})^*$.

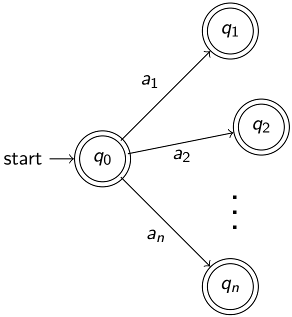


Figure: Automaton of L

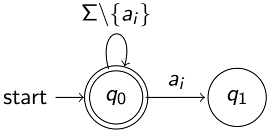


Figure: Automaton of K_i

Toy example II

Let us prove that $\kappa(\rho(L)) = 2^n$:

- $\rho(L) = \bigcup_i K_i =$ words not containing all letters of Σ.
- $\rho(L)/x = \rho(L)/y$ iff $\Sigma(x) = \Sigma(y)$

Proof

If $a \in \Sigma(x)$ and $a \notin \Sigma(y)$, let $w = \cdot_{a_i \neq a} a_i$. Then $xw \notin \rho(L)$ but $yw \in \rho(L)$, thus $\rho(L)/x \neq \rho(L)/y$.

If $\Sigma(x) = \Sigma(y)$ then for all $w \in \Sigma^*$, $\Sigma(xw) = \Sigma(yw)$ thus $xw \in \rho(L) \iff yw \in \rho(L)$. Thus $\rho(L)/x = \rho(L)/y$.

- $\kappa(\rho(L)) = 2^n = \kappa(L) \prod_i \kappa(K_i)$.

General / Singular substitutions

We can answer the first question :

- For subword closed languages, $\phi_{\mathsf{sub}}(n) \geq 2^n$ (in fact $\phi_{\mathsf{sub}}(n) \geq n^n$).

$$\phi_{\mathsf{sub}} \nleq n^{\mathcal{O}(1)} \text{ for subword closed languages}$$

There is still one left :

$$\kappa(L^{a \leftarrow K}) \leq \kappa(L)\kappa(K) \text{ for subword closed languages ?}$$

- Let's focus on singular substitutions

Example of Singular substitution

Let $K = \downarrow(\{\texttt{ab}\}) + \downarrow(\{\texttt{ba}\})$ and $L = \downarrow(\{\texttt{ab}\})$

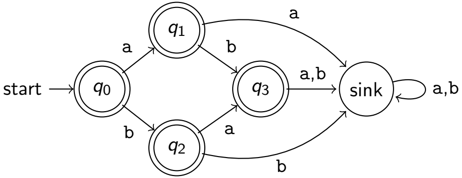


Figure: Automaton of K

$$\rho(L) = L^{a \leftarrow K} = \downarrow(\{\texttt{aab}\}) + \downarrow(\{\texttt{bab}\}) = \{\texttt{a}, \texttt{b}\} \cdot \downarrow(\{\texttt{ab}\})$$

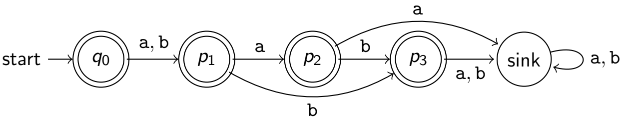


Figure: Automaton of $\rho(L)$

Computing quotients

Definition

$$L^\epsilon = \begin{cases} \emptyset, & \text{if } \epsilon \notin L\,, \\ \{\epsilon\}, & \text{otherwise.} \end{cases}$$

Rules of computation of quotients (Brzozowski et al. 2010)

- $b/a = \begin{cases} \emptyset, & \text{if } b \neq a\,, \\ \epsilon, & \text{otherwise.} \end{cases}$
- $(L + K)/a = L/a + K/a$
- $(L \cdot K)/a = (L/a) \cdot K + L^\epsilon \cdot (K/a)$

Computing quotients of subword closed languages

Proposition

L non-empty subword closed implies $L^{\epsilon} = \{\epsilon\}$

Rules of computation of quotients for non-empty subword closed

$$b/a = \begin{cases} \emptyset, & \text{if } b \neq a\,, \\ \epsilon, & \text{otherwise.} \end{cases}$$

$$(L + K)/a = L/a + K/a$$

$$\boxed{(L \cdot K)/a = (L/a) \cdot K + K/a}$$

Example of quotient computation

Take $K = \downarrow(\{\mathtt{ab}\}) + \downarrow(\{\mathtt{ba}\})$

$$
\begin{aligned}
K/a &= \downarrow(\{\mathtt{ab}\})/a + \downarrow(\{\mathtt{ba}\})/a \\
&= [\downarrow(\{\mathtt{a}\}) \cdot \downarrow(\{\mathtt{b}\})]/a + [\downarrow(\{\mathtt{b}\}) \cdot \downarrow(\{\mathtt{a}\})]/a \\
&= \downarrow(\{\mathtt{a}\})/a \cdot \downarrow(\{\mathtt{b}\}) + \downarrow(\{\mathtt{b}\})/a + \downarrow(\{\mathtt{b}\})/a \cdot \downarrow(\{\mathtt{a}\}) + \downarrow(\{\mathtt{a}\})/a \\
&= \{\epsilon\} \cdot \downarrow(\{\mathtt{b}\}) + \emptyset + \emptyset \cdot \downarrow(\{\mathtt{a}\}) + \{\epsilon\} \\
&= \downarrow(\{\mathtt{b}\})
\end{aligned}
$$

Computing quotients of singular substitutions

Lemma

Let L, K downward closed, $a \neq b \in \Sigma$ and $\rho(L) = L^{a \leftarrow K}$

$$\rho(L)/\epsilon = \rho(L)$$

$$\rho(L)/a = K/a \cdot \rho(L/a)$$

$$\rho(L)/b = \rho(L/b) + K/b \cdot \rho(L/a)$$

We can efficiently compute quotients of singular substitutions.

Application of those formulas

Proposition

If $\rho(L/b)$ or $K/b \cdot \rho(L/a)$ includes the other, then for all word w

$$\boxed{\rho(L)/w = P \cdot \rho(Q)}$$

with $P \in \mathcal{R}(K) \cup \{\{\epsilon\}\}$ and $Q \in \mathcal{R}(L)$.

If $\rho(L/b)$ or $K/b \cdot \rho(L/a)$ includes the other then we have the quadratic bound

The case of disjoint alphabets

Theorem : Disjoint alphabets

Let L, K be downward closed languages based on disjoints alphabets. Then $\boxed{\kappa\big(L^{a \leftarrow K}\big) \leq \kappa(L)\kappa(K)}$.

Proof idea

In this case $\rho(L/b)$ or $K/b \cdot \rho(L/a)$ is equal to $\emptyset$.

The case of disjoint alphabets

Theorem : Disjoint alphabets

Let L, K be downward closed languages based on disjoints alphabets. Then $\boxed{\kappa\big(L^{a \leftarrow K}\big) \leq \kappa(L)\kappa(K)}$.

Proof idea

In this case $\rho(L/b)$ or $K/b \cdot \rho(L/a)$ is equal to $\emptyset$.

Corollary

Let L and $(K_{a_i})_{a_i \in \Sigma}$ downward closed languages such that all $|\Sigma| + 1$ languages have pairwise disjoints alphabets, then we have

$$\boxed{\kappa\big(L^{a_1 \leftarrow K_{a_1}, \ldots, a_n \leftarrow K_{a_n}}\big) \leq \kappa(L) \prod_{1 \leq i \leq n} \kappa(K_{a_i})}$$

Definition SREs

Definition : SRE (Abdulla et al. 2004)

- Atom : $\alpha = \begin{cases} a + \epsilon & \text{with } a \in \Sigma \\ B^* & \text{with } B \subseteq \Sigma \end{cases}$
- Product (of atoms) : $I = \prod_{1 \leq i \leq n} \alpha_i$ with α_i an atom.
- SRE : $E = \sum_{1 \leq j \leq m} I_j$ with I_j a product.

Theorem (Abdulla et al. 2004)

L on a finite alphabet Σ is subword closed if and only if it can be defined by an SRE.

SREs are useful to study subword closed languages.

The case of product of atoms

Theorem : Product of atoms

Let I be a product of atoms and K a downward closed language, then $\boxed{\kappa(I^{a \leftarrow K}) \leq \kappa(K)\kappa(I)}$.

Proof idea

In this case $\rho(L/b)$ or $K/b \cdot \rho(L/a)$ includes the other.

Conclusion

- The general substitution has exponential state complexity on subword closed languages.
- We found formulas to compute quotients of substitutions.
- In some cases, singular substitution has quadratic state complexity on subword closed languages.

Unsolved : $\kappa(L^{a \leftarrow K}) \leq \kappa(L)\kappa(K)$ for subword closed ?

Future works

- Answer the question : $\kappa(L^{a \leftarrow K}) \leq \kappa(L)\kappa(K)$?
- Study other operations (roots, shuffle, ...)
- Study other class of languages (prefix closed, factor closed, ...)

References I

Abdulla et al. (2004). "Using Forward Reachability Analysis for Verification of Lossy Channel Systems". In: *Formal Methods in System Design* 25.1, pp. 39–65. ISSN: 1572-8102.

Brzozowski (2009). "Quotient Complexity of Regular Languages". In: *Proceedings 11th Int. Workshop on Descriptional Complexity of Formal Systems, DCFS 2009*. Vol. 3. EPTCS, pp. 17–28.

Brzozowski, Janusz, Galina Jirásková, and Chenglong Zou (2014). "Quotient Complexity of Closed Languages". In: *Theory of Computing Systems* 54.2, pp. 277–292. ISSN: 1433-0490.

References II

Brzozowski, Jirásková, and Li (2010). "Quotient Complexity of Ideal Languages". In: *Proc. 9th Latin American Symp. Theoretical Informatics (LATIN 2010)*. Springer, pp. 208–221.

Hospodár (2019). "Descriptional Complexity of Power and Positive Closure on Convex Languages". In: *Proc. 24th Int. Conf. Implementation and Application of Automata (CIAA 2019)*. Springer, pp. 158–170. ISBN: 978-3-030-23679-3.