On the Complexity of the Syntax of Tree Languages

Symeon Bozapalidis
Aristotle University of Thessaloniki

Antonios Kalampakas
Democritus University of Thrace

Syntactic Complexity on Graphs

The notion of *syntactic complexity* was first introduced for graph languages in

Symeon Bozapalidis, Antonios Kalampakas,
Recognizability of graph and pattern languages,
Acta Informatica 2006.

It is a function mapping any pair of natural numbers (m, n) to the number of syntactic classes of graphs with type (m, n).

Connected graphs have bellian complexity.

The language of Eulerian graphs is syntactically more complicated than that of connected graphs.

In this paper we develop a complexity theory for tree languages.

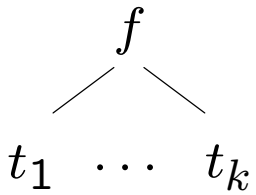
Trees

To construct trees we use a finite ranked alphabet $\Gamma = \bigcup \Gamma_k$ and a set $X = \{x_1, x_2, \dots\}$ of variables, $X_n = \{x_1, x_2, \dots, x_n\}$, $X_0 = \emptyset$.

The set $T_\Gamma(X)$ of trees over Γ and X is inductively defined.

- $\Gamma_0 \cup X \subseteq T_\Gamma(X)$
- $t_1, \dots, t_k \in T_\Gamma(X)$ and $f \in \Gamma_k$ implies $f(t_1, \dots, t_k) \in T_\Gamma(X)$.

Often $f(t_1, \dots, t_k)$ is depicted as



Subsets of $T_\Gamma(X)$ are called *tree languages*.

Substitution

Given $t, t_1, \ldots, t_n \in T_\Gamma(X_n)$, we denote by

$$t[t_1, \ldots, t_n]$$

the result of substituting t_i at every occurrence of x_i, inside t, $1 \leq i \leq n$.

P_Γ the subset of $T_\Gamma(x)$ consisting of all trees with exactly one occurrence of the variable x.

P_Γ becomes a monoid with operation the substitution at x:

$$\text{for } \tau, \pi \in P_\Gamma,\ \tau \cdot \pi = \tau[\pi/x].$$

This monoid is free over the set of trees of the form

$$f(t_1, \ldots, t_{i-1}, x, t_{i+1}, \ldots, t_k)$$

and acts, again by substitution at x, on the set T_Γ:

$$P_\Gamma \times T_\Gamma \to T_\Gamma, \quad (\tau, t) \mapsto \tau \cdot t = \tau[t/x].$$

Deterministic bottom up tree automaton

It is a structure $\mathcal{M} = (Q, \mu, F)$

- Q is the finite set of states,
- $F \subseteq Q$ is the final state set,
- $\mu = (\mu_f : Q^k \to Q)$ moves of $\mathcal{M}$.

The reachability map $\mu_{\mathcal{M}} : T_\Gamma \to Q$ is defined inductively

$$\mu_{\mathcal{M}}(f(t_1, \ldots, t_k)) = \mu_f(\mu_{\mathcal{M}}(t_1), \ldots, \mu_{\mathcal{M}}(t_k)),$$

and the behavior of $\mathcal{M}$ is the tree language

$$|\mathcal{M}| = \{t \mid t \in T_\Gamma, \mu_{\mathcal{M}}(t) \in F\} = \mu_{\mathcal{M}}^{-1}(F).$$

These languages are called *recognizable*.

Derivatives

The right derivative of a tree language $L \subseteq T_\Gamma$ at $t \in T_\Gamma$ is given by

$$Lt^{-1} = \{\tau \mid \tau \in P_\Gamma, \tau \cdot t \in L\},$$

The left derivative of a tree language $L \subseteq T_\Gamma$ at $\tau \in P_\Gamma$ is given by

$$\tau^{-1}L = \{t \mid t \in T_\Gamma, \tau \cdot t \in L\}.$$

The equivalence relation $\sim_L$ on T_Γ

$$t \sim_L t' \quad \text{if} \quad Lt^{-1} = Lt'^{-1}$$

is well known to be a (syntactic) congruence, i.e.,

$$t_1 \sim_L t'_1, \ldots, t_k \sim_L t'_k \text{ and } f \in \Gamma_k$$

imply

$$f(t_1, \ldots, t_k) \sim_L f(t'_1, \ldots, t'_k).$$

Characterization of Recognizability

Proposition. *The folowing conditions are equivalent for a language* $L \subseteq T_\Gamma$

i) L *is recognizable*

ii) $card\{Lt^{-1} \mid t \in T_\Gamma\} < \infty$

iii) $card\{\tau^{-1}L \mid \tau \in P_\Gamma\} < \infty$

iv) *The syntactic congruence* $\sim_L$ *has finite index (i.e., a finite number of classes).*

Syntactic Complexity of Tree Languages

It is a tool to study the syntax of a tree language.

It counts the number of distinct syntactic classes of trees with a fixed yield length.

The syntactic complexity of a tree language $L \subseteq T_\Gamma$ is the function $SC_L : \mathbb{N} \to \mathbb{N}$

$$SC_L(n) = card\{\overline{t} \mid t \in T_\Gamma, |y(t)| = n\}, \quad n \in \mathbb{N}$$

where $\overline{t}$ stands for the $\sim_L$-class of t and the function yield $y : T_\Gamma \to \Gamma_0^*$, is inductively defined by

$$y(c) = c, \quad y(f(t_1, \ldots, t_k)) = y(t_1) \cdots y(t_k).$$

Alternatively we have

$$SC_L(n) = card\{Lt^{-1} \mid t \in T_\Gamma, |y(t)| = n\}, \quad n \in \mathbb{N}.$$

We say that a language $L \subseteq T_\Gamma$ has bounded, polynomial or exponential syntactic complexity if the explicit formula defining the function SC_L is upper bounded by a constant, polynomial or exponential function respectively.

The syntactic complexity of a tree language $L \subseteq T_\Gamma$ does not depend on Γ.

According to the previous Proposition every recognizable tree language has bounded syntactic complexity

$$SC_L(n) \leq k \text{ for a fixed } k \text{ and all } n \in \mathbb{N}.$$

Does bounded syntactic complexity characterize recognizability?

Example

Take the alphabet $\Gamma = \{f, \alpha\}$ with $rank(f) = 2$, $rank(\alpha) = 0$ and consider the tree languages L_{bal} of all balanced trees

$$L_{bal} = \{t_k \mid t_0 = \alpha,\ t_{k+1} = f(t_k, t_k),\ k \geq 1\},$$

and L_{fib} of all Fibonacci trees

$$L_{fib} = \{s_k \mid s_0 = s_1 = a,\ s_{k+2} = f(s_{k+1}, s_k)\}.$$

Observe that $|y(t_k)| = 2^k$ while $|y(s_k)| = f_k$, the k-th Fibonacci number.

The trees $\tau_k = f(t_k, x)$ have the property

$$\tau_k \cdot t_k \in L_{bal},\ \text{but}\ \tau_k \cdot t \notin L_{bal}\ \text{for}\ t \neq t_k,$$

The trees $\pi_k = f(s_{k+1}, x)$, have the property

$$\pi_k \cdot s_k \in L_{fib},\ \text{but}\ \pi_k \cdot s \notin L_{fib}\ \text{for}\ s \neq s_k.$$

Therefore the derivatives $L_{bal}t_k^{-1}$ and $L_{fib}s_k^{-1}$ are pairwise distinct.

It turns out that

$$card\{L_{bal}t^{-1} \mid t \in T_{\Gamma}\} = \infty$$

and

$$card\{L_{fib}s^{-1} \mid s \in T_{\Gamma}\} = \infty$$

and so both the languages L_{bal} and L_{fib} are not recognizable.

Moreover, it holds

$$\begin{aligned} SC_{L_{bal}}(n) &= 2, \text{ if } n = 2^k \\ &= 1, \text{ otherwise} \end{aligned}$$

and similarly,

$$\begin{aligned} SC_{L_{fib}}(n) &= 2, \text{ if } n = f_k \\ &= 1, \text{ otherwise.} \end{aligned}$$

Thus the languages L_{bal}, L_{fib} are not recognizable but they have bounded syntactic complexity.

Proposition. *The class of tree languages with bounded syntactic complexity properly contains the class of recognizable tree languages.*

Proposition. *Given the ranked alphabet* $\Gamma = \{f_1, \ldots, f_k, \alpha\}$, $rank(f_i) = 2$, $rank(\alpha) = 0$, *the Dyck tree language of order* k

$$D_k = \{t \mid t \in T_\Gamma, |t|_{f_1} = \cdots = |t|_{f_k}\}$$

has polynomial syntactic complexity of degree $k - 1$.

For $t, t' \in T_\Gamma$ we have

$$D_k t^{-1} = D_k t'^{-1} \text{ if and only if } |t|_{f_i} = |t'|_{f_i}.$$

The number of binary symbols occurring in a tree $t \in T_\Gamma$ with yield length n is $n-1$.

The different ways to share the symbols f_1, $\ldots$, f_k in the nodes of t is equal with the number of k-tuples of natural numbers $(x_1, \ldots, x_k)$ verifying the equation

$$x_1 + \cdots + x_k = n - 1$$

It is well known from Combinatorics,

$$\binom{n-1+k-1}{k-1} = \binom{n+k-2}{k-1}$$
$$= \frac{1}{(k-1)!}\, n(n+1)\cdots(n+k-2).$$

Hence

$$SC_{D_k}(n) = \frac{1}{(k-1)!}\, n(n+1)\cdots(n+k-2).$$

A tree language $L \subseteq T_\Gamma$ such that for every n

$$card\{Lt^{-1} \mid |y(t)| = n\} = card\{t \mid |y(t)| = n\}$$

is called syntactically hard. Of course such a language L has the highest possible syntactic complexity, i.e.,

$$SC_L(n) = card\{t \mid t \in T_\Gamma, |y(t)| = n\}.$$

In the case that $\Gamma = \{f, a\}$, with $rank(f) = 2$, $rank(\alpha) = 0$ the above number is well known from Combinatorics and is the $n-1$-th Catalan number C_{n-1}, where

$$C_n = \frac{1}{n+1}\binom{2n}{n} \simeq \frac{4^n}{n^{3/2}\sqrt{\pi}}.$$

Proposition. *The diagonal language*

$$L_d = \{f(t,t) \mid t \in T_\Gamma\}, \quad \Gamma = \{f, \alpha\},$$

is syntactically hard

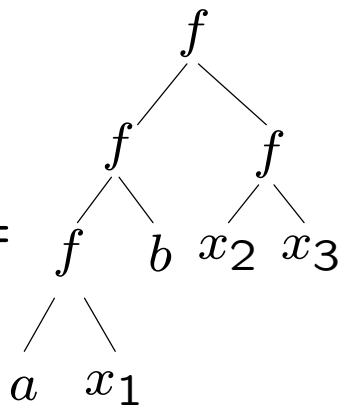
Refined Syntactic Complexity

As we have seen the growth rate of the function SC_L gives no information that allows us to compare recognizable tree languages with respect to their complexity.

We will provide an efficient complexity measure for recognizable tree languages.

Let us denote by $P_\Gamma^{(n)}$ the subset of $T_\Gamma(X_n)$ formed by all trees where $x_1, \ldots, x_n$ occur in the yield of the tree (in this order from left to right) exactly once. For instance the tree

$$\tau = \quad \in P_\Gamma^{(3)}.$$



For every $n \geq 1$ there is a function

$$P_\Gamma^{(n)} \times T_\Gamma^n \to T_\Gamma, \quad (\tau, t_1, \ldots, t_n) \mapsto \tau[t_1, \ldots, t_n].$$

With respect to $L \subseteq T_\Gamma$, two dual notions of derivatives can be defined:

$$\tau^{-1}L = \{(t_1, \ldots, t_n) \mid \tau[t_1, \ldots, t_n] \in L\},$$

$$L(t_1, \ldots, t_n)^{-1} = \{\tau \mid \tau \in P_\Gamma^{(n)},\ \tau[t_1, \ldots, t_n] \in L\},$$

for all $\tau \in P_\Gamma^{(n)}$ and $t_1, \ldots, t_n \in T_\Gamma$.

Theorem. *For $L \subseteq T_\Gamma$, the following conditions are equivalent*

i) *L is recognizable*

ii) $card\{\tau^{-1}L \mid \tau \in P_\Gamma^{(n)}\} < \infty$

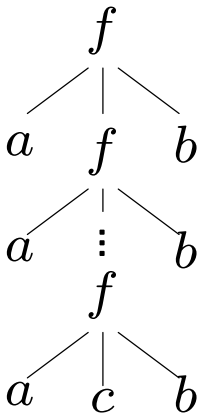
iii) $card\{L(t_1, \ldots, t_n)^{-1} \mid t_1, \ldots, t_n \in T_\Gamma\} < \infty.$

The refined syntactic complexity of a recognizable tree language $L \subseteq T_\Gamma$ is the function $RSC_L : \mathbb{N} \to \mathbb{N}$ sending every natural number n to the number of the distinct left derivatives $\tau^{-1}L$ when τ ranges over $P_\Gamma^{(n)}$, i.e.,

$$RSC_L(n) = card\{\tau^{-1}L \mid \tau \in P_\Gamma^{(n)}\}.$$

Example

Consider the recognizable tree language L consisting of all trees t_m of the form



m occurrences of f, $m \geq 1$.

We prove that the language L has linear refined syntactic complexity.

$$RSC_L(n) = 2(n+1)$$

Example

Consider the regular tree grammar G :

$$y_1 \to f(y_1, y_2), \quad y_2 \to g(y_1, y_2),$$

$$y_1 \to a, \quad y_2 \to b,$$

and the tree language $L(G, y_1)$ generated by G starting from the variable y_1.

Its refined syntactic complexity is

$$RSC_{L(G,y_1)}(n) = 2^n + 1.$$

Constructive Complexity of a Tree Automaton

We display a way to measure how complicated the structure of a tree automaton is.

Given an automaton $\mathcal{M} = (Q, \mu, F)$, for every $t \in T_\Gamma(X_n)$ and $q_1, \ldots, q_n \in Q$, the element $t[q_1, \ldots, q_n] \in Q$ is inductively defined as follows

- for $t = x_i$, $x_i[q_1, \ldots, q_n] = q_i$, $1 \leq i \leq n$;
- for $t = c \in \Gamma_0$, $c[q_1, \ldots, q_n] = \mu_c$;
- for $t = f(t_1, \ldots, t_k)$, $f \in \Gamma_k$, $t_i \in T_\Gamma(X_n)$

$$f(t_1, \ldots, t_k)[q_1, \ldots, q_n] = \mu_f(t_1[q_1, \ldots, q_n], \ldots, t_k[q_1, \ldots, q_n]).$$

The constructive complexity of the automaton

$$\mathcal{M} = (Q, \mu, F)$$

is a function

$$CC_{\mathcal{M}} : \mathbb{N} \to \mathbb{N}$$

defined by the formula

$$CC_{\mathcal{M}}(n) = card\{\tau^{-1}F \mid \tau \in P_{\Gamma}^{(n)}\}$$

where

$$\tau^{-1}F = \{(q_1, \ldots, q_n) \mid \tau[q_1, \ldots, q_n] \in F\}.$$

Since for all n we have $\tau^{-1}F \subseteq Q^n$, we get that

$$CC_{\mathcal{M}}(n) \leq 2^{(cardQ)^n}$$

Example

Let Γ be a finite ranked alphabet and consider the automaton

$$\mathcal{M} = (\mathbb{Z}_m, \mu, \{0\})$$

- $\mathbb{Z}_m = \{0, 1, \ldots, m-1\}$ is the additive group of integers $mod\, m$,

- The moves

$$\mu_f : \mathbb{Z}_m^k \to \mathbb{Z}_m$$

  are given by

  $\mu_c = 1, \quad \mu_f(\alpha_1, \ldots, \alpha_k) = 1 + \alpha_1 + \cdots + \alpha_k,$

  where at the right hand side the designated addition is the $mod\, m$ addition.

- F=$\{0\}$

The reachability map $\mu_{\mathcal{M}} : T_\Gamma \to \mathbb{Z}_m$ sends every tree t to its $mod\, m$ size, i.e.,

$$\mu_{\mathcal{M}}(t) = |t|(mod\, m)$$

and the behavior of $\mathcal{M}$ consists of all trees whose size is divisible by m.

For $\tau, \tau' \in P_\Gamma^{(n)}$, we have

$$\tau^{-1}F = \tau'^{-1}F \text{ if and only if } |\tau| \equiv |\tau'|(mod\, m).$$

Consequently, there are exactly m distinct classes, that is $CC_{\mathcal{M}}(n) = m$ for all n and thus $\mathcal{M}$ has constant constructive complexity.

A naturally arising question concerns the comparison of $CC_{\mathcal{M}}$ and $RSC_{|\mathcal{M}|}$.

Proposition. *Let $\mathcal{M}, \mathcal{M}'$ be reachable tree automata. If there is a simulation $h : \mathcal{M} \to \mathcal{M}'$ then both $\mathcal{M}$ and $\mathcal{M}'$ have the same constructive complexity*

$$CC_{\mathcal{M}} = CC_{\mathcal{M}'}.$$

Proposition. *If $\mathcal{M}$ is a reachable automaton with behavior L, then*

$$CC_{\mathcal{M}} = RSC_L.$$

Proposition. *For every recognizable tree language $L \subseteq T_\Gamma$ it holds*

$$RSC_L(n) \leq 2^{(CardQ_L)^n}, \quad \text{for all } n.$$