Pregroup Grammars are Turing Equivalent

Greg Kobele and Marcus Kracht, UCLA

1 Algebraic Grammars

If $f : X \to Y$ is a function and $U \subseteq X$ then we write $f[U] := \{f(x) : x \in U\}$ for the image of U under f. An **signature** Ω over a set F of function symbols is a function $\Omega : F \to \mathbb{N}$ (where $0 \in \mathbb{N}$). An Ω–**algebra** is a pair $\langle B, \mathfrak{I}\rangle$ such that $\mathfrak{I}$ assigns to each $f \in F$ a function $B^{\Omega(f)} \to B$. $\mathfrak{B}$ is **partial** if $\mathfrak{I}(f)$ may also be a partial function. We shall also write $f^{\mathfrak{B}}$ in place of $\mathfrak{I}(f)$. For example, let $F = \{1, \otimes\}$ and $\Omega(1) = 0$ and $\Omega(\otimes) = 2$. An Ω–algebra is a pair $\langle B, \mathfrak{I}\rangle$ such that $\mathfrak{I}(1) : \{\varnothing\} \to B$ and $\mathfrak{I}(\otimes) : B^2 \to B$ (recall that $B^0 = \{\varnothing\}$). Thus we may also view $\mathfrak{I}(1)$ as an element of B instead of a zeroary function. A particular example of an Ω–algebra is the algebra $\mathfrak{S}(A)$ of strings over an alphabet A. Here, the underlying set of $\mathfrak{S}(A)$ is the set A^* of strings over A and $1^{\mathfrak{S}(A)} = \varepsilon$ as well as $\otimes^{\mathfrak{S}(A)} = {}^\frown$, the concatenation of strings. Notice that concatenation is associative, that is, for all strings x, y and z.

(1) $\quad x^\frown(y^\frown z) = (x^\frown y)^\frown z$

Given two algebras $\mathfrak{B} = \langle B, \mathfrak{I}\rangle$ and $\mathfrak{C} = \langle C, \mathfrak{J}\rangle$, we put $\mathfrak{B} \times \mathfrak{C} = \langle A \times C, \mathfrak{K}\rangle$ where

(2) $$\mathfrak{K}(f)(\langle b_1, c_1\rangle, \langle b_2, c_2\rangle, \cdots, \langle b_{\Omega(f)}, c_{\Omega(f)}\rangle) := \langle \mathfrak{I}(f)(b_1, b_2, \cdots, b_{\Omega(f)}, \mathfrak{J}(f)(c_1, c_2, \cdots, c_{\Omega(f)}))\rangle$$

This is undefined if the right hand side is. In turn this is undefined if any of the functions $\mathfrak{I}(f)$ or $\mathfrak{J}(f)$ is undefined on their respective arguments.

Let $\mathfrak{B}$ be an algebra and $X \subseteq B$ a set. Then algebra generated by X in $\mathfrak{B}$, is obtained as follows. First, we call a subset M of A **closed** if whenever for all $f \in F$ and all $i < \Omega(f)$: $a_i \in M$ also $f^{\mathfrak{B}}(a_1, a_2, \cdots, a_{\Omega(f)}) \in M$. We let $\langle X\rangle$ be the least closed set containing M. The algebra $\mathfrak{B}$ defines an algebra $\mathfrak{X}$ on $\langle X\rangle$ via $f^{\mathfrak{X}}(a_1, a_2, \cdots, a_{\Omega(f)}) := f^{\mathfrak{B}}(a_1, a_2, \cdots, a_{\Omega(f)})$. The left hand is defined iff the right hand side is. We can give a more concrete characterisation as follows. Say that a **term** is built from variables using the function symbols of F. Terms with only the binary symbol $\otimes$ as function symbols are x, y, $x \otimes y$, $x \otimes (y \otimes x)$, and so on. If $t(x_1, \cdots, x_n)$ is a term, and c_i, $1 \leq i \leq n$ are elements of the algebra, then $t(c_1, \cdots, c_n)$ denotes the result of substituting the values c_i for the variables

x_i. With this, $\langle X\rangle$ consists of all elements $t(c_1, \cdots, c_n)$, where $t(x_1, \cdots, x_n)$ is a term and for all $i \leq n$, $c_i \in X$. A term $t(x_1, \cdots, x_n)$ defines a term function $t^{\mathfrak{B}} : \langle c_1, \cdots, c_n\rangle \mapsto t(c_1, \cdots, c_n)$ on A^n. We shall henceforth not distinguish between the term t and the term function it induces on B. If f is a term function and U a set, write $f[U] := \{f(\vec{c}) : \vec{c} \in U^n\}$. We can now also say

(3) $\langle X\rangle = \bigcup\{f[X] : f \text{ a term function of } \mathfrak{B}\}$

An **algebraic grammar scheme** is a partial algebra over the signature $\{\otimes\}$ of the form $\mathfrak{C} \times \mathfrak{S}(A)$. Here, $\mathfrak{C}$ is the algebra of **categories** and $\mathfrak{S}(A)$ is the algebra of **exponents**. A **lexicon** is a finite subset of $C \times A^*$. Finally, we select a set $S \subseteq B$ of so–called **designated categories**. We shall require that this set is in some sense finitely specified. For example, in standard categorial grammar S consists of just one element, denoted here by c. In pregroup grammars we take S to be the set of all categories below c; this is possible because pregroups have a partial order. The triple $G = \langle\mathfrak{G}, D, S\rangle$, where $\mathfrak{G}$ is an algebraic grammar scheme, D a lexicon and S a subset of C is called an **algebraic grammar**. Thus the scheme merely provides for the categories and the strings to be manipulated, while the lexicon provides the actual entries that the grammar uses. It is required that the lexicon is finite; hence we can equate algebraic grammars with finitely generated grammar schemes. A string $\vec{x}$ is **accepted** if there is a $c \in C$ such that $\langle c, \vec{x}\rangle \in \langle D\rangle$. We write

(4) $L = L(G)$

Basic categorial grammar can be construed as an algebraic grammar in the following way. Let T be the set of types over a given set of basic types, using the constructor $/$. Now introduce a binary partial function $\cdot$ which satisfies the following law:

(5) $(x/y) \cdot y = x$

That is to say, $u \cdot y$ is defined iff $u = x/y$ for some x, and the then result is x. This forms the algebra $\mathfrak{T} = \langle T, \cdot\rangle$. Then a categorial grammar takes the form $\mathfrak{T} \times \mathfrak{S}(A)$. Now fix a symbol c of T. We say that a string x is a **sentence over the lexicon** L if the least partial algebra containing L also contains $\langle \mathsf{c}, x\rangle$.

2 Some Basic Results

The first theorem is about closure under homomorphisms. Recall that a string homomorphism from A^* to B^* is a map h that satisfies

(6) $h(\vec{x}{}^\frown\vec{y}) = h(\vec{x}){}^\frown h(\vec{y})$

Such a map is uniquely determined by its action on A.

Theorem 1 *Let $G = \langle \mathfrak{B} \times \mathfrak{S}(A), D, S \rangle$ be an algebraic grammar and $h : A^* \to B^*$ a string homomorphism. Put $D^h := \{\langle b, h(\vec{x})\rangle : \langle b, \vec{x}\rangle \in D\}$ and $G^h := \langle \mathfrak{B} \times \mathfrak{S}(B), D^h, S\rangle$. Then $L(G^h) = h[L(G)]$.*

Proof. We extend h to a map $1 \times h : \mathfrak{B} \times \mathfrak{S}(A) \to \mathfrak{B} \times \mathfrak{S}(B)$ by putting $(1 \times h)(\langle b, \vec{x}\rangle) := \langle b, h(\vec{x})\rangle$. This is a homomorphism, as is easily verified. This means that for every term t, and all elements $s_i = \langle b_i, \vec{x}_i\rangle$

(7) $(1 \times h)(t(s_1, \cdots, s_n)) = t((1 \times h)(s_1), \cdots, (1 \times h)(s_n))$

In turn, this means that $(1\times h)[t[X]] = t[(1\times h)[X]]$. Hence, $(1\times h)[t[D]] = f[D^h]$. It follows that

$$\begin{aligned} \langle D^h\rangle &= \bigcup\{t[D^h] : t \text{ a term function}\} \\ (8) \qquad &= \bigcup\{(1 \times h)[t[D]] : t \text{ a term function}\} \\ &= (1 \times h)[\langle D\rangle] \end{aligned}$$

Now, $\vec{y} \in L(G^h)$ iff there is a $c \in S$ such that $\langle c, \vec{y}\rangle \in \langle D^h\rangle$ iff there is a $c \in S$ such that $\langle c, \vec{y}\rangle \in (1 \times h)[\langle D\rangle]$ iff there is a $c \in S$ and a $\vec{x} \in A^*$ such that $\vec{y} = h(\vec{x})$ and $\langle c, \vec{x}\rangle \in D$ iff there is $\vec{x} \in L(G)$ such that $h(\vec{x}) = \vec{y}$. Hence, $L(G^h) = h[L(G)]$, as promised. ⊣

This theorem did not make any assumptions on the algebra of categories. Next we shall exhibit a general construction, namely the **product** of two grammars. This works as follows. Let $G_1 = \langle \mathfrak{B}_1\times\mathfrak{S}(A), D_1, S_1\rangle$ and $G_2 = \langle \mathfrak{B}_2\times\mathfrak{S}(A), D_2, S_2\rangle$ grammars. Put $D_1 \times' D_2 := \{\langle b, b', \vec{x}\rangle : \langle b, \vec{x}\rangle \in D_1, \langle b', \vec{x}\rangle \in D_2\}$. Finally, put $G_1 \times G_2 := \langle \mathfrak{B}_1 \times \mathfrak{B}_2 \times \mathfrak{S}(A), D_1 \times' D_2, S_1 \times S_2\rangle$.

Suppose that the lexicon is such that $\langle b, \vec{x}\rangle \in D$ only if $\vec{x} \in A$. Then an analysis of a string of length n will contain exactly n occurrences of lexical elements. So, $\vec{x} \in \langle D\rangle$ iff there is a term $t(x_1, \cdots, x_n)$ containing exactly $n - 1$ (!) occurrences of $\otimes$ and lexical elements $s_i = \langle b_1, \vec{x}_i\rangle$, $1 \leq i \leq n$, such that

(9) $t(s_1, \cdots, s_n) = \langle c, \vec{x}\rangle$

for some c. If $\otimes$ is associative, we can choose the following term:

(10) $(\cdots((s_1 \otimes s_2) \otimes s_2) \cdots s_n)$

This will be useful for the next theorem.

Theorem 2 *Suppose that G_1 and G_2 are grammars whose algebras of categories are associative. Furthermore, assume that each of the D_i only contains items of the form $\langle b, \vec{x}\rangle$ where $\vec{x} \in A$. Then $L(G_1 \times G_2) = L(G_1) \cap L(G_2)$.*

Proof. Define the following maps. $\pi_1 : \mathfrak{B}_1 \times \mathfrak{B}_2 \times \mathfrak{S}(A) \to \mathfrak{B}_1 \times \mathfrak{S}(A) : \langle b, b', \vec{x}\rangle \mapsto \langle b, \vec{x}\rangle$, and $\pi_2 : \mathfrak{B}_1 \times \mathfrak{B}_2 \times \mathfrak{S}(A) \to \mathfrak{B}_1 \times \mathfrak{S}(A) : \langle b, b', \vec{x}\rangle \mapsto \langle b', \vec{x}\rangle$. These maps are actually homomorphisms. Furthermore, $\pi_1[D_1 \times' D_2] = D_1$ as well as $\pi_2[D_1 \times' D_2] = D_2$. From this we can already deduce that if $\vec{x} \in L(G_1 \times G_2)$ then $\vec{x} \in L(G_1) \cap L(G_2)$. For if $\vec{x} \in L(G_1 \times G_2)$ then there are b, b' such that $b \in S_1$ and $b' \in S_2$ and $\langle b, b', \vec{x}\rangle \in \langle D_1 \times' D_2\rangle$, then $\langle b, \vec{x}\rangle = \pi_1(\langle b, b', \vec{x}\rangle \in \pi_1[\langle D_1 \times' D_2\rangle] = \langle \pi_1[D_1 \times' D_2]\rangle = \langle D_1\rangle$. Similarly $\langle b', \vec{x}\rangle \in \langle D_2\rangle$ is established. For the converse we need to make use of our further assumptions. Suppose that $\vec{x} \in L(G_1)$ and $\vec{x} \in L(G_2)$. Then there is a term $t(y_1, \cdots, y_p)$ and elements $s_i \in D_1$, such that $t(s_1, \cdots, s_p) = \langle c, \vec{x}\rangle$ for some $c \in S_1$; and there is a term $t'(z_1, \cdots, z_q)$ and elements $s'_i \in D_2$ such that $t(s'_1, \cdots, s'_q) = \langle c', \vec{x}\rangle$ for some $c' \in S_2$. We are not guaranteed that t and t' are the same term. However, under the assumptions made, as the discussion above has revealed, we do have $p = q = n$, and we can use the same term. Moreover, we have $s_i = \langle b_i, x_i\rangle$ and $s'_i = \langle b'_i, x_i\rangle$ for certain $b_i \in B_1$ and $b'_i \in B_2$. It follows that

(11) $t(\langle b_1, b'_1, x_1\rangle, \cdots, \langle b_n, b'_n, x_n\rangle) = \langle c, c', \vec{x}\rangle$

and since $\langle c, c'\rangle \in S_1 \times S_2$, we now have $\vec{x} \in L(G_1 \times G_2)$. ⊣

The theorem can be improved. It is often customary to allow for the empty string ε in the lexicon. In context free grammars it is possible to eliminate the use of the empty string. However, it is also possible to include the string. One possibility is when both algebras of categories have a unit. A **unit** is an element 1 which satisfies $x = 1 \cdot x$ and $x = x \cdot 1$ for all x. In this case, the product grammar shall contain also the following items: $\langle c, 1, \varepsilon\rangle$ iff $\langle c, \varepsilon\rangle \in D_1$ and $\langle 1, c', \varepsilon\rangle$ iff $\langle c', \varepsilon\rangle \in D_2$. Or, equivalently, we assume that both lexicons contain the entry $\mathbb{I} := \langle 1, \varepsilon\rangle$. Now the argument goes through as follows. Suppose $\vec{x} \in L(G_1 \times G_2)$, and we have terms t and t' as above. Now we want to find a term u in the product. To that end, let $t = s_1 \otimes t^*$ and $t' = s'_1 \otimes t'^*$. and $s_1 = \langle b_1, \vec{x}_1\rangle$, $s'_1 = \langle b', \vec{x}'_1\rangle$. Case

1. $\vec{x}_1 = \vec{x}'_1$. Then on condition we have found the term u^* for t^* and $t^{*\prime}$, we can put $u := \langle b_1, b'_1, \vec{x}_1 \rangle \otimes u^*$. Case 2. $\vec{x}_1 \neq \vec{x}'_1$. Then, since the strings are either empty or a letter, and they cannot be distinct letters, one of the strings is empty and the other is a letter. Without loss of generality let $\vec{x}_1 = a$ and $\vec{x}'_1 = \varepsilon$. Now put $u = \langle b_1, 1, a \rangle \otimes \langle 1, b'_1, \varepsilon \rangle \otimes u^*$. Case 3. One of the terms is exhausted. So, we have — without loss of generality — that t' is empty. Then $s_1 = \langle b_1, \varepsilon \rangle$, because the strings of t and t' multiply to the same string; as t' is empty, this product is empty. Then $u := \langle b_1, 1, \varepsilon \rangle \otimes u^*$. This completes the definition of u.

We shall briefly prove a theorem to the effect that the algebras must be infinite in order to yield nontrivial results.

Theorem 3 *Let $G = \langle \mathfrak{B} \times \mathfrak{S}(A), D, S \rangle$ be a grammar with finite algebra of categories. If $\otimes$ is associative, $L(G_1)$ is regular.*

Proof. Define an automaton $\mathfrak{A}$ as follows. The set of states is $\{q_0\} \cup B$, and the transition table is $q_0 \xrightarrow{x} b$ iff $\langle b, x \rangle \in D$ and $b \xrightarrow{x} b'$ iff there is a c such that $b' = b \cdot c$ and $\langle c, x \rangle \in D$. It is easily shown that $q_0 \xrightarrow{x_1 \cdots x_n} c$ iff there are b_i, $1 \leq i \leq n$, such that for all i $\langle b_i, x_i \in D$ and $b_1 \cdot b_2 \cdot \cdots \cdot b_n = c$ iff

(12) $\langle b_1, x_1 \otimes \langle b_2, x_2 \rangle \otimes \cdots \otimes \langle b_n, x_n \rangle = \langle c, x_1 \cdots x_n \rangle$

Finally, let the set of accepting states be S. Then $x_1 \cdots x_n \in L(\mathfrak{A})$ iff $q_0 \xrightarrow{x_1 \cdots x_n} c$ for some $c \in S$ iff there is a term $t(y_1, \cdots, y_n)$ and elements $s_i = \langle b_i, x_i \rangle \in D$ such that $t(s_1, \cdots, s_n) = \langle c, x_1 \cdots x_n \rangle$ for some $c \in S$ iff $x_1 \cdots x_n \in L(G)$. ⊣

3 Pregroups

A **pregroup** is a structure $\langle G, 1, \cdot, {}^r, {}^\ell, \leq \rangle$ such that the following holds.

① $x \cdot 1 = 1 \cdot x = x$,

② $x \cdot (y \cdot z) = (x \cdot y) \cdot z$,

③ $x \cdot x^r \leq 1 \leq x^r x$,

④ $x^\ell x \leq 1 \leq x x^\ell$, and

⑤ if $x \leq y$ then $xz \leq yz$ and $zx \leq zy$.

The set of designated categories is the set of all $x \leq c$, where c is the category of sentences. Thus, a **pregroup grammar** is a triple $\langle \mathfrak{B} \times \mathfrak{S}(A), D, C\rangle$, where $\mathfrak{B}$ is a pregroup and $C = \{x : x \leq c\}$, $c \in B$. For a class $\mathcal{V}$ of pregroups and a language L, we say that L is $\mathcal{V}$–**definable** if there is a pregroup grammar G whose algebra is from $\mathcal{V}$ such that $L = L(G)$.

Theorem 4 (Buszkowski) *Let $\mathcal{F}$ be the variety of free pregroups. The class of languages definable by $\mathcal{F}$ is the class of context free languages.*

We shall note that the product of two pregroups also is a pregroup, where $\langle x, y\rangle \leq \langle x', y'\rangle$ iff $x \leq x'$ and $y \leq y'$. Let $\mathfrak{P}$ be the class of direct products of two pregroups. Our main theorem is this.

Theorem 5 *The class of languages definable by $\mathcal{P}$ is the class of recursively enumerable languages.*

Proof. It is known that every recursively enumerable language is the homomorphic image of an intersection of two context free languages. By Proposition 1 it is enough to show that every intersection of two CF languages is $\mathcal{P}$–definable. By Proposition 2 and Theorem 4 this is shown if we can see to it that there is a pregroup grammar with a free pregroup for a CF language whose lexicon consists only of letters. This can be achieved as follows. Suppose that the lexicon contains an entry $\langle b, x_1^\frown\vec{y}\rangle$, where $\vec{y} = x_2 \cdots x_n$, $n > 1$. The algebra of categories is the free pregroup $\mathfrak{F}(X)$ over some set X of generators. Then adjoin to X a new element u, and let the new algebra of categories be $\mathfrak{F}(X \cup \{u\})$. Then the lexicon D is embedded into this new algebra in the natural way. We eliminate the entry $\langle b, \vec{x}\rangle$ and add instead the following entries: $\langle bu, x_1\rangle$ and $\langle u^r, \vec{y}\rangle$. It is a matter of direct verification that the new grammar accepts the same strings. ⊣