An Infinite Sequence of Full AFL-Structures, Each of Which Possesses an Infinite Hierarchy

Peter R.J. Asveld

Department of Computer Science, Twente University of Technology
P.O. Box 217, 7500 AE Enschede, the Netherlands
e-mail: infprja@cs.utwente.nl

Abstract. We investigate different sets of operations on languages which results in corresponding algebraic structures, viz. in different types of full AFL's (full Abstract Family of Languages). By iterating control on ETOL-systems we show that there exists an infinite sequence $\mathcal{C}_m$ ($m \geq 1$) of classes of such algebraic structures (full AFL-structures): each class is a proper superset of the next class ($\mathcal{C}_m \supset \mathcal{C}_{m+1}$). In turn each class $\mathcal{C}_m$ contains a countably infinite hierarchy, i.e., a countably infinite chain of language families $K_{m,n}$ ($n \geq 1$) such that (i) each $K_{m,n}$ is closed under the operations that determine $\mathcal{C}_m$, and (ii) each $K_{m,n}$ is properly included in the next one: $K_{m,n} \subset K_{m,n+1}$.

1. Introduction

Usually, each newly introduced family of formal languages will be studied sooner or later with respect to its closure properties. In the early days of formal language theory, this meant that (non)closure under each known operation has to be established separately. Then one realised that some operations are more fundamental that others, and that some other operations can be expressed in these fundamental ones: they are "polynomials" over those fundamental operations. In short, a more algebraic view on this part of formal language theory emerged.

An important step in this algebraic approach to families of languages has been the introduction of the notion of full Abstract Family of Languages (full AFL), being a nontrivial family of languages closed under the operations: union, concatenation, Kleene $\star$, homomorphism, inverse homomorphism, and intersection with regular sets [9]. Similar as in ordinary algebra —where one went from groups to semigroups, rings, and fields— full AFL's gave rise to weaker structures (full trios, full semi-AFL's [9]) and more powerful ones: full substitution-closed AFL's [10], full super-AFL's [11] and full hyper-AFL's [1].

For each class $\mathcal{C}$ of these full AFL-like structures, it has been shown that $\mathcal{C}$ is not trivial in the sense that it does not solely consists of a few "isolated" language families but, to the contrary, that $\mathcal{C}$ is infinite. This latter fact is usually established by showing the existence of an infinite hierarchy i.e., a countably infinite chain of language families K_n $(n \geq 1)$ such that (i) each K_n is closed under the operations that determine $\mathcal{C}$, and (ii) each K_n is properly included in the next one: $K_n \subset K_{n+1}$.

In this paper we show that by iterating control on ETOL-systems, as studied in [7, 8], we obtain an infinite sequence of full AFL-structures. Each class $\mathcal{C}_m$ $(m \geq 1)$ in this sequence is a proper superset of the next class: $\mathcal{C}_m \supset \mathcal{C}_{m+1}$. So the full AFL-structures in $\mathcal{C}_{m+1}$ are more powerful than those in $\mathcal{C}_m$. And each class $\mathcal{C}_m$ is nontrivial, since it contains an infinite hierarchy of language families $K_{m,n}$ $(n \geq 1)$, each of which is properly included in the next one: $K_{m,n} \subset K_{m,n+1}$. The proofs of these results heavily rely on the main results of [4] and [7]. Many properties of full substitution-closed AFL's, full super-AFL's and full hyper-AFL's (quoted in §5) have their counterparts for the classes $\mathcal{C}_m$; see §7.

The remaining part of this paper is organized as follows. §2 consists of Preliminaries. The definitions and properties of some generalized grammatical models (controlled K-iteration grammar, context-free K-grammar, regular K-grammar) are in §3 and §4, respectively. In §5 we recall the corresponding full AFL-structures. In §6 we quote two fundamental theorems which enable us to establish the main result in §7. Some concluding remarks are in §8.

2. Preliminaries

We already mentioned the standard text [9] on full AFL's and related concepts. Some other books on formal language theory, like [12, 13, 15], also treat the relevant issues to the extent we use in this paper. For Lindenmayer or L systems we refer to [14].

Henceforth, Σ_ω denotes a countably infinite set of symbols. A *family of languages*, or a *family* for short, K is a set of languages L with $L \subseteq \Sigma_L^\star$ such that each Σ_L is a finite subset of Σ_ω. As usual, we assume that for each language L in the family K, the alphabet Σ_L is minimal, i.e., a symbol α belongs to Σ_L if and only if there exists a word w of L in which α occurs. A family K is called *nontrivial* if K contains a nonempty language L with $L \neq \{\lambda\}$, where λ denotes the empty word. We also assume that each family is closed under isomorphism.

We will use well-known families like FIN (family of finite languages), REG (regular languages), CF (context-free languages), as well as the family ONE of singleton languages: ONE $= \{\{w\} \mid w \in \Sigma_\omega^\star\}$, the family ALPHA of alphabets: ALPHA $= \{\Sigma \mid \Sigma \subset \Sigma_\omega,\ \Sigma \text{ is finite}\}$, and the family SYMBOL of singleton alphabets: SYMBOL $= \{\{\sigma\} \mid \sigma \in \Sigma_\omega\}$.

We often need the concept of substitution and a few of its generalizations (Definitions 2.1 and 5.1).

Definition 2.1. Let K be a family and V an alphabet. A K-*substitution* is a mapping $\tau : V \to K$; it is extended to words over V by $\tau(\lambda) = \{\lambda\}$, and $\tau(\alpha_1 \dots \alpha_n) = \tau(\alpha_1) \dots \tau(\alpha_n)$ where $\alpha_i \in V$ $(1 \leq i \leq n)$, and to languages L over V by $\tau(L) = \bigcup\{\tau(w) \mid w \in L\}$. If K equals FIN or REG, τ is called a *finite* or a *regular substitution*, respectively.

Given families K and K', let $\hat{\text{Sub}}(K, K')$ be defined by $\hat{\text{Sub}}(K, K') = \{\tau(L) \mid \tau \text{ is a } K'\text{-substitution}; L \in K\}$. A family K is *closed* under K'-substitution if $\hat{\text{Sub}}(K, K') \subseteq K$, and K is *closed under substitution*, if K is closed under K-substitution.

$\tau : V \to K$ is a K-*substitution over* V if $\tau(\alpha) \subseteq V^\star$ for each $\alpha \in V$. A K-substitution τ over V is *nested*, if $\alpha \in \tau(\alpha)$ for each $\alpha \in V$. □

Definition 2.2. A *prequasoid* K is a nontrivial family that is closed under finite substitution and under intersection with regular languages. For each family K, let $\Pi(K)$ denote the smallest prequasoid that includes K. A *quasoid* is a prequasoid that contains an infinite language. □

It is easy to see that each [pre]quasoid includes the smallest [pre]quasoid REG [FIN, respectively], whereas FIN is the only prequasoid that is not a quasoid; cf. [1, 2].

Definition 2.3. A *full Abstract Family of Languages* or *full AFL* is a nontrivial family of languages closed under union, concatenation, Kleene $\star$, homomorphism, inverse homomorphism, and intersection with regular languages. A *full substitution-closed AFL* is a full AFL closed under substitution. □

Frequently, the following characterization of full AFL's is useful; cf. Theorem 5.4(1) below.

Proposition 2.4. [10, 9, 2] *A family K of languages is a full AFL if and only if K is a prequasoid closed under regular substitution (i.e., $\hat{\text{Sub}}(K, \text{REG}) \subseteq K$), and under substitution in the regular languages (i.e., $\hat{\text{Sub}}(\text{REG}, K) \subseteq K$).* □

3. Some Generalized Grammars

In this section we recall the definitions of some grammar types with a countably infinite number of rules rather than a finite number. These generalizations are based on the concepts of ETOL-system (Definition 3.1), controlled ETOL-system (3.2), context-free grammar (3.4), and non-self-embedding context-free grammar (3.5).

Definition 3.1. [1, 2] Let K be a family. A K-*iteration grammar* $G = (V, \Sigma, U, S)$ consists of an alphabet V, a terminal alphabet Σ $(\Sigma \subseteq V)$, an initial symbol S $(S \in V)$, and a finite set U of K-substitutions over V.

The language $L(G)$ generated by G is defined by $L(G) = U^\star(S) \cap \Sigma^\star = \bigcup\{\tau_p(\dots(\tau_1(S))\dots) \mid p \geq 0;\ \tau_i \in U,\ 1 \leq i \leq p\} \cap \Sigma^\star$.

The family of languages generated by K-iteration grammars is denoted by $H(K)$. For $m \geq 1$, $H_m(K)$ is the family generated by K-iteration grammars that contain at most m K-substitutions in U. □

Definition 3.2. [1, 2] Let Γ and K be a families of languages. A *Γ-controlled K-iteration grammar* or *(Γ, K)-iteration grammar* is a pair (G, M) that consists of a K-iteration grammar $G = (V, \Sigma, U, S)$ and a *control language* M, i.e., M is a language over U, and $M \in \Gamma$. The language $L(G, M)$ generated by (G, M) is defined by $L(G, M) = M(S) \cap \Sigma^\star = \bigcup\{\tau_p(\dots(\tau_1(S))\dots) \mid p \geq 0;\ \tau_i \in U,\ \tau_1 \dots \tau_p \in M\} \cap \Sigma^\star$.

The family generated by (Γ, K)-iteration grammars is denoted by $H(\Gamma, K)$. Similarly, $H_m(\Gamma, K)$ is the family generated by (Γ, K)-iteration grammars that contain at most m K-substitutions in U $(m \geq 1)$. □

Clearly, $H(K) = \bigcup_{m \geq 1} H_m(K)$ and $H(\Gamma, K) = \bigcup_{m \geq 1} H_m(\Gamma, K)$.

Example 3.3. By taking concrete values for the parameter K we obtain some families of Lindenmayer languages; viz. $H(\text{ONE}) = \text{EDTOL}$, $H_1(\text{ONE}) = \text{EDOL}$, $H(\text{FIN}) = \text{ETOL}$, and $H_1(\text{FIN}) = \text{EOL}$. Readers unfamiliar with L systems are referred to [14]. Alternatively, they may view these equalities as definitions. □

Definition 3.4. [16, 2, 6] Let K be a family. A *context-free K-grammar* G is a K-iteration grammar $G = (V, \Sigma, U, S)$ of which each substitution τ from U is a nested K-substitution over V; so $\alpha \in \tau(\alpha)$ for each $\alpha \in V$ and each $\tau \in U$.

The family of languages generated by context-free K-grammars is denoted by $A(K)$. For $m \geq 1$, $A_m(K)$ is the family generated by context-free K-grammars that contain at most m K-substitutions in U. □

Definition 3.5. [2, 5] Let K be a family and let U be a finite set of nested K-substitutions over an alphabet V. Then U is called *not self-embedding* if for all $u \in U^\star$ and for all α in V, the implication $w_1 \alpha w_2 \in u(\alpha) \ \Rightarrow\ (w_1 = \lambda \text{ or } w_2 = \lambda)$ holds for all $w_1, w_2 \in V^\star$.

A *regular K-grammar* $G = (V, \Sigma, U, S)$ is a context-free K-grammar where U is a non-self-embedding set of nested K-substitutions over V.

The family of languages generated by regular K-grammars is denoted by $R(K)$. For each $m \geq 1$, $R_m(K)$ is the family generated by regular K-grammars that contain at most m K-substitutions in U. □

Example 3.6. When we take K equal to FIN, we have $A(\text{FIN}) = \text{CF}$ and $R(\text{FIN}) = \text{REG}$. □

4. Some Properties of These Generalized Grammars

This section consists of some useful properties of the grammatical devices that we discussed in the previous section.

First, we remember that regular control does not extend the generating power of K-iteration grammars.

Theorem 4.1. [1, 2] *If* $K \supseteq$ ONE*, then* $H(\text{REG}, K) = H(K)$. □

The number of K-substitutions in a Γ-controlled K-iteration grammar can be reduced to two in case the parameters Γ and K satisfy some very simple conditions, since we have

Proposition 4.2. [1, 2] *Let* K *be a family with* $K \supseteq$ SYMBOL*.*
(1) *If* Γ *is a family closed under* λ*-free homomorphism, then* $H_2(\Gamma, K) = H_m(\Gamma, K) = H(\Gamma, K)$ *for each* $m \geq 2$*.*
(2) *For each* $m \geq 2$, $H_2(K) = H_m(K) = H(K)$*.* □

For [non-self-embedding] context-free K-grammars a reduction to a single, equivalent [non-self-embedding] K-substitution is possible.

Proposition 4.3. [2, 6, 5] *Let* K *be a family closed under union with languages from* SYMBOL*. If* $K \supseteq$ SYMBOL*, then* $A_1(K) = A_m(K) = A(K)$ *and* $R_1(K) = R_m(K) = R(K)$ *for each* $m \geq 1$*.* □

Comparing Propositions 4.2(2) and 4.3 reveals that providing regular or context-free K-grammars with control does not lead to interesting results.

We conclude this section with a few useful inclusion properties for which we need some additional terminology.

Definition 4.4. A family Γ is closed under *left marking* [*right marking*] if for each language L in Γ with $L \subseteq \Sigma^\star$ for some Σ, and for each symbol c not in Σ, the language $\{c\}L$ [$L\{c\}$, respectively] belongs to Γ. And Γ is closed under *full marking* if Γ is closed under both left and right marking. □

Proposition 4.5. [1, 4]
(1) *Let* Γ *be a family closed under right marking, and let* K *be a family with* $K \supseteq$ ONE*. Then* $\Gamma \subseteq H(\Gamma, K)$ *and* $K \subseteq H(\Gamma, K)$*.*
(2) *Let* Γ *be a family closed under (i) left or right marking, (ii) union or concatenation, and (iii) Kleene star. If* K *is a family with* $K \supseteq$ SYMBOL*, then* $H(K) \subseteq H(\Gamma, K)$*.* □

Proposition 4.6. [2, 4, 5, 6] *Let* K *be a family closed under union with languages from* SYMBOL*. If* $K \supseteq$ SYMBOL*, then* $K \subseteq H(K)$*,* $K \subseteq A(K)$*, and* $K \subseteq R(K)$*.* □

Proposition 4.7. [1, 2, 4, 5, 6] *Let Γ be a family closed under full marking. If the family K is a prequasoid, then so are the families $R(K)$, $A(K)$, $H(K)$ and $H(\Gamma, K)$.* □

5. Some Full AFL-Structures

In §2 we already encountered full AFL's and full substitution-closed AFL's. For full AFL-like structures weaker than full AFL, we refer to [9]. The present section is devoted to structures stronger than full AFL, which are related to the generalized grammars of §3.

Definition 5.1. A family K is closed under *iterated substitution* if for each language L from K with $L \subseteq V^\star$ for some alphabet V, and for each finite set U of K-substitutions over V, the language $U^\star(L)$, defined by

$$U^\star(L) = \bigcup\{\tau_p \ldots \tau_1(L) \mid p \geq 0,\ \tau_i \in U\ (1 \leq i \leq p)\},$$

belongs to K. In case each substitution in U is nested, then K is called *closed under nested iterated substitution*.

A *full hyper-AFL* [1] is a full AFL closed under iterated substitution; a *full super-AFL* [11] is a full AFL closed under nested iterated substitution. □

Definition 5.2. Let K be a family. By $\hat{\mathcal{F}}(K)$ [$\hat{\mathcal{R}}(K)$, $\hat{\mathcal{A}}(K)$, and $\hat{\mathcal{H}}(K)$] we denote the smallest full AFL [full substitution-closed AFL, full super-AFL, and full hyper AFL, respectively] that includes K. □

Theorem 5.3. [2, 4, 5, 6] *Let K be a family. Then K is a*
(1) *full substitution-closed AFL, if and only if K is a prequasoid and $R(K) = K$.*
(2) *full super-AFL, if and only if K is a prequasoid and $A(K) = K$.*
(3) *full hyper-AFL, if and only if K is a prequasoid and $H(K) = K$.* □

Theorem 5.4. [2, 4, 5, 6] *Let K be a family. Then*
(1) $\hat{\mathrm{Sub}}(\mathrm{REG}, \hat{\mathrm{Sub}}(\Pi(K), \mathrm{REG})) = \hat{\mathrm{Sub}}(\hat{\mathrm{Sub}}(\mathrm{REG}, \Pi(K)), \mathrm{REG})$ *is a full AFL that includes K.*
(2) *$R\Pi(K)$ is a full substitution-closed AFL that includes K.*
(3) *$A\Pi(K)$ is a full super-AFL that includes K.*
(4) *$H\Pi(K)$ is a full hyper-AFL that includes K.* □

Theorem 5.3(3) says that K is a full hyper-AFL if and only if $\Pi(K) = K$ and $H(K) = K$. Consequently, the smallest full hyper-AFL $\hat{\mathcal{H}}(K)$, that includes a family K, equals $\hat{\mathcal{H}}(K) = \bigcup\{w(K) \mid w \in \{\Pi, H\}^\star\}$ or, equivalently, $\hat{\mathcal{H}}(K) = \{\Pi, H\}^\star(K)$. According Theorem 5.5 below, this infinite set of strings over $\{\Pi, H\}$ can be reduced to the single string $H\Pi$. Obviously, a similar remark applies to the other full AFL-structures.

Theorem 5.5. [2, 4, 5, 6] *Let K be a family of languages. Then* $\hat{\mathcal{F}}(K) = \hat{\mathrm{Sub}}(\mathrm{REG}, \hat{\mathrm{Sub}}(\Pi(K), \mathrm{REG})) = \hat{\mathrm{Sub}}(\hat{\mathrm{Sub}}(\mathrm{REG}, \Pi(K)), \mathrm{REG})$, $\hat{\mathcal{R}}(K) = R\Pi(K)$, $\hat{\mathcal{A}}(K) = A\Pi(K)$, *and* $\hat{\mathcal{H}}(K) = H\Pi(K)$. □

Theorem 5.6. REG [REG, CF, ETOL, *respectively*] *is the smallest full AFL* [*full substitution-closed AFL, full super-AFL, full hyper-AFL*]. □

Each full hyper-AFL is a full super-AFL, and each full super-AFL is a full substitution-closed AFL. But none of the converse implications hold; cf. Theorem 5.6.

6. Two Fundamental Results

This section contains two results (Theorems 6.1 and 6.4) that constitute the principal steps in obtaining the main result of this paper; cf. §7. The first one is a direct consequence of a more general statement from [4].

Theorem 6.1. [4] *Let Γ_1, Γ_2 and K be families of languages and let Γ_2 be closed under full marking, union or concatenation, and Kleene $\star$. If $K \supseteq$ ALPHA, then* $H(\Gamma_1, H(\Gamma_2, K)) \subseteq H(\hat{\mathrm{Sub}}(\Gamma_1, \Gamma_2), K)$. □

Corollary 6.2. [4] (1) *Let Γ be a family of languages closed under full marking and under substitution that satisfies $\Gamma \supseteq$ REG. If K is a family with $K \supseteq \mathrm{ALPHA} \cup \mathrm{ONE}$, then $H(\Gamma, H(\Gamma, K)) = H(\Gamma, K)$.*
(2) *Let Γ be a family of languages that is closed under full marking, union, concatenation, and Kleene $\star$. If K is a family with $K \supseteq \mathrm{ALPHA} \cup \mathrm{ONE}$, then $H(H(\Gamma, K)) = H(\Gamma, K)$.* □

Corollary 6.2(2) has been used to show that certain families $H(\Gamma, K)$ are full hyper-AFL's. In particular, there exist infinite chains of full hyper-AFL's [7, 8]; see also Theorem 6.4 below. In §7 we will use Corollary 6.2(1) to obtain related results.

Corollary 6.3. [4] *If $K \supseteq \mathrm{ALPHA} \cup \mathrm{ONE}$, then $HH(K) = H(K)$.* □

In establishing results like Theorems 5.3(3), 5.4(4) and 5.5, Corollary 6.3 plays an important rôle.

The strictness of our infinite sequence of full AFL-structures, as well as the properness of our hierarchies rely on the following theorem that stems from a rich collection [7, 8] of similar hierarchies.

Theorem 6.4. [7, 8] *Let $K_0 =$ REG and $K_{i+1} = H(K_i, \mathrm{FIN})$ for each $i \geq 0$. Then $\{K_i\}_{i\geq 1}$ is an infinite hierarchy of full hyper-AFL's, i.e.,*

- *for each $i \geq 1$, K_i is a full hyper-AFL, and*
- *for each $i \geq 1$, K_i is properly included in K_{i+1}: $K_i \subset K_{i+1}$.* □

7. An Infinite Sequence of Full AFL-Structures

In this section we define a family of full AFL-structures (Definition 7.1). Then we show some properties of these full AFL-structures (Proposition 7.2, Theorems 7.3 and 7.4) and our main result (Theorem 7.5).

In this section we frequently write $H_\Gamma(K)$ instead of $H(\Gamma, K)$ in order to distinguish between the two arguments of $H(\Gamma, K)$: K will play the rôle of "ordinary" argument, whereas Γ is an additional parameter over which we proceed inductively (Theorems 6.4, 7.4 and 7.5).

In view of Theorem 5.3, the following definition is a natural extension of the notion of full hyper-AFL.

Definition 7.1. Let Γ be a fixed family of languages. An arbitrary family K is a *full Γ-hyper-AFL* if K is a prequasoid with $H(\Gamma, K) = K$, or equivalently, with $H_\Gamma(K) = K$. For each family K, let $\hat{\mathcal{H}}_\Gamma(K)$ denote the smallest full Γ-hyper-AFL that includes K. □

Ordinary full hyper-AFL's are now obtained as a special instance of Definition 7.1, since Theorem 4.1 implies

Proposition 7.2. *A family K of languages is a full hyper-AFL if and only if K is a full* REG*-hyper-AFL.* □

Theorem 7.3. *Let Γ be a full substitution-closed AFL.*
(1) *Each full Γ-hyper-AFL is a full hyper-AFL.*
(2) *If K is a family, then $H_\Gamma\Pi(K)$ is a full Γ-hyper-AFL that includes K.*
(3) *For each family K, $\hat{\mathcal{H}}_\Gamma(K) = H_\Gamma\Pi(K)$.*
(4) $H_\Gamma(\mathrm{FIN})$ *is the smallest full Γ-hyper-AFL.*

Proof. (1) It suffices to show that $H(\Gamma, K) = K$ implies $H(K) = K$. Since $\Gamma \supseteq \mathrm{REG}$, we have by Propositions 4.6 and 4.5(2): $K \subseteq H(K) \subseteq H(\Gamma, K) = K$. Hence $H(K) = K$.

(2) The result follows from Propositions 4.7, 4.5(1), Corollary 6.2(1), and the fact that Γ is closed under substitution.

(3) By the inclusion $K \subseteq \hat{\mathcal{H}}_\Gamma(K)$ and the monotonicity of both H_Γ and Π, we have $H_\Gamma\Pi(K) \subseteq H_\Gamma\Pi\hat{\mathcal{H}}_\Gamma(K)$. According to Definition 7.1 this yields $H_\Gamma\Pi(K) \subseteq \hat{\mathcal{H}}_\Gamma(K)$. Now Theorem 7.3(2) and Proposition 4.5(1) imply that $H_\Gamma\Pi(K)$ is a full Γ-hyper-AFL that includes K. Hence we obtain that $\hat{\mathcal{H}}_\Gamma(K) = H_\Gamma\Pi(K)$.

(4) FIN is the smallest prequasoid, and Theorem 7.3(3). □

Compare Theorem 7.3(2), (3) and (4) with their "uncontrolled counterparts": Theorems 5.4(4), 5.5, and 5.6, respectively.

Theorem 7.4. *Let the family K be a prequasoid, let $Q_0 = \mathrm{REG}$ and*

$Q_{i+1} = H(Q_i, K)$ for each $i \geq 0$. Then for each $i \geq 0$, Q_j is a full Q_i-hyper-AFL provided that $j > i$.

Proof. A simple proof by induction on i using Theorem 4.1, Propositions 4.5(1) and 4.7, and Corollary 6.2(2), yield the following facts: (F1) Q_i is a full hyper-AFL for each $i \geq 1$, and (F2) $Q_i \subseteq Q_j$ provided $j \geq i$.

Next we prove by induction on i ($i \geq 0$) that Q_j ($j > i$) is a full Q_i-hyper-AFL.

Basis ($i = 0$): We have to show that Q_j is a full Q_0-hyper-AFL for each $j \geq 1$. Since $Q_0 = \text{REG}$ and each Q_j is a full REG-hyper-AFL if and only if Q_j is a full hyper-AFL (Proposition 7.2), the statement follows from (F1).

Induction step: Assume that for each $j > i$, Q_j is a full Q_i-hyper-AFL.

We have to show that each family Q_j with $j > i + 1$ is a full Q_{i+1}-hyper-AFL.

Consider an arbitrary Q_j with $j > i + 1$; then $Q_j = H(Q_{j-1}, K)$. As $j - 1 > i$, the induction hypothesis implies that Q_{j-1} is a full Q_i-hyper-AFL. Now by Theorem 7.3(1) and Proposition 4.7, Q_j is a prequasoid.

So it remains to show that $H(Q_{i+1}, Q_j) \subseteq Q_j$, since the converse inclusion follows from Proposition 4.5(2) and (F1).

From the definition of Q_j and Theorem 6.1 respectively, we obtain

$$H(Q_{i+1}, Q_j) = H(Q_{i+1}, H(Q_{j-1}, K)) \subseteq H(\hat{\text{Sub}}(Q_{i+1}, Q_{j-1}), K).$$

We already remarked that the induction hypothesis implies that Q_{j-1} is a full Q_i-hyper-AFL. By Theorem 7.3(1), Q_{j-1} is a full hyper-AFL and so Q_{j-1} is closed under substitution. As $j - 1 \geq i + 1$, we have $Q_{i+1} \subseteq Q_{j-1}$ by (F2), and consequently, $\hat{\text{Sub}}(Q_{i+1}, Q_{j-1}) \subseteq Q_{j-1}$. Hence we have $H(Q_{i+1}, Q_j) \subseteq H(\hat{\text{Sub}}(Q_{i+1}, Q_{j-1}), K) \subseteq H(Q_{j-1}, K) = Q_j$, which completes the induction. □

We are now ready for the main result.

Theorem 7.5. *Let $K_0 = \text{REG}$ and $K_{m+1} = H(K_m, \text{FIN})$ for $m \geq 0$, and let $\mathcal{C}_m$ be the class of all full K_m-hyper-AFL's. Then for each $m \geq 1$,*
(1) *the class $\mathcal{C}_m$ is a proper superset of $\mathcal{C}_{m+1}$: $\mathcal{C}_m \supset \mathcal{C}_{m+1}$,*
(2) *the class $\mathcal{C}_m$ contains an infinite hierarchy of full K_m-AFL's, i.e., a countably infinite chain of language families $K_{m,n}$ ($n \geq 1$) such that* (i) *$K_{m,n}$ is a full K_m-AFL, and* (ii) *for each $n \geq 1$, $K_{m,n}$ is properly included in the next one: $K_{m,n} \subset K_{m,n+1}$.*

Proof. (1) follows from Theorems 6.4, 7.4 (with $K = \text{FIN}$) and 7.3(4).

(2) For fixed m ($m \geq 1$), we define $\{K_{m,n}\}_{n \geq 1}$ by $K_{m,n} = K_{m+n}$ for each $n \geq 1$. By Theorems 6.4 and 7.4 this is an infinite hierarchy of full K_m-hyper-AFL's. □

8. Concluding Remarks

We extended the finite sequence "full AFL, full substitution-closed AFL, full super-AFL, full hyper-AFL" to a countably infinite sequence of full AFL-structures, each of which possesses properties (Theorem 7.3) similar to those of the members of the initial, finite sequence (Theorems 5.4, 5.5 and 5.6). And each new class of full AFL-structures is nontrivial in the sense that it contains a countably infinite hierarchy (Theorem 7.5).

The concept of full AFL abstracts the regular languages in case they are characterized by nondeterministic finite automata and regular expressions. Full substitution-closed AFL's generalize the regular languages when they are viewed as the languages generated by non-self-embedding context-free grammars. Similarly, full super-AFL's, full hyper-AFL's and full Γ-hyper-AFL's correspond to context-free grammars, ETOL-systems and Γ-controlled ETOL-systems, respectively.

Actually, a full Γ-hyper-AFL is a full AFL closed under Γ-controlled iterated substitution. A family K is closed under *Γ-controlled iterated substitution*, if for each language L from K with $L \subseteq V^\star$ for some alphabet V, for each finite set U of K-substitutions over V, and for each language M over U from the family Γ, the language $M(L)$, defined by

$$M(L) = \bigcup\{\tau_p \dots \tau_1(L) \mid p \geq 0,\ \tau_i \in U\ (1 \leq i \leq p),\ \tau_1 \dots \tau_p \in M\},$$

belongs to K; cf. Definition 5.1.

We could take the obvious, next step: a family K is closed under *controlled iterated substitution* if K is closed under K-controlled iterated substitution. And a family K is a *full $\top$-hyper-AFL* if K is a prequasoid and $H(K, K) = K$.

Up to now there are only a few full $\top$-hyper-AFL's known. Of course there are the smallest full $\top$-hyper-AFL $K_\omega = \bigcup_{m \geq 1} K_m$ (cf. Theorem 6.4), and the largest effective one: the family RE of recursively enumerable languages (since we have $H(\mathrm{RE}, \mathrm{RE}) = \mathrm{RE}$ by Church's thesis). A less trivial example is the family $\mathcal{L}_{\star\mathrm{OI}}$ studied in [17]. In particular, it is an open question whether these exist infinitely many full $\top$-hyper-AFL's.

In this context we quote an interesting result (Proposition 8.1) for which we need some additional terminology. A family K is closed under *removal of right endmarker*, if for each language L ($L \subseteq \Sigma^\star$) and each symbol c with $c \notin \Sigma$, $L\{c\}$ in K implies L in K.

Proposition 8.1. [3]. *Let K be a family closed under removal of right endmarker. If* $\mathrm{DSPACE}(\log n) \subseteq K \subset \mathrm{RE}$*, then K is not closed under controlled iterated (λ-free) substitution.* □

(Originally, this statement has been formulated for λ-free substitutions only. Obviously, it applies to arbitrary substitutions as well.) Proposition 8.1 implies that DSPACE($\log n$) is neither a subfamily of K_ω nor of $\mathcal{L}_{\star\mathrm{OI}}$.

On the other hand it is known that the hierarchy of Theorem 6.4, and consequently K_ω, is situated within the family of context-sensitive languages; see [3, 7, 8]. Therefore, the smallest elements in the classes $\mathcal{C}_m$ ($m \geq 1$), as well as the infinite hierarchies $\{K_{m,n}\}_{n\geq 1}$ ($m \geq 1$) (Theorem 7.5) are in the family of context-sensitive languages.

References

1. P.R.J. Asveld: Controlled iteration grammars and full hyper-AFL's, *Inform. Contr.* **34** (1977) 248–269.
2. P.R.J. Asveld: *Iterated Context-Independent Rewriting – An Algebraic Approach to Formal Languages* (1978), Ph.D. Thesis, Dept. of Appl. Math., Twente University of Technology, Enschede, the Netherlands.
3. P.R.J. Asveld: Space-bounded complexity classes and iterated deterministic substitution, *Inform. Contr.* **44** (1980) 282–299.
4. P.R.J. Asveld: Controlled fuzzy parallel rewriting, pp. 49–70 in: Gh. Păun & A. Salomaa (eds.): *New Trends in Formal Languages — Control, Cooperation, and Combinatorics* (1997), Lect. Notes. in Comp. Sci. **1218**, Springer, Berlin, etc.
5. P.R.J. Asveld: The non-self-embedding property for generalized fuzzy context-free grammars, *Publ. Math. Debrecen* **54** *Suppl.* (1999) 553–573.
6. P.R.J. Asveld: Fuzzy context-free languages — Part I: Generalized fuzzy context-free grammars (2000), CTIT Tech. Rep. No. 00-03, Enschede, the Netherlands.
7. J. Engelfriet: Three hierarchies of transducers, *Math. Syst. Th.* **15** (1982) 95-125.
8. J. Engelfriet: Hierarchies of hyper-AFL's, *J. Comp. Syst. Sci.* **30** (1985) 86–115.
9. S. Ginsburg: *Algebraic and Automata-Theoretic Properties of Formal Languages* (1975), North-Holland, Amsterdam.
10. S. Ginsburg & E.H. Spanier: Substitution in families of languages. *Inform. Sci.* **2** (1970) 83–110.
11. S.A. Greibach: Full AFL's and nested iterated substitution, *Inform. Contr.* **16** (1970) 7–35.
12. M.A. Harrison: *Introduction to Formal Language Theory* (1978), Addison-Wesley, Reading, Mass.
13. J.E. Hopcroft & J.D. Ullman: *Introduction to Automata Theory, Languages, and Computation* (1979), Addison-Wesley, Reading, Mass.
14. G. Rozenberg & A. Salomaa: *The Mathematical Theory of L Systems* (1980), Academic Press, New York.
15. A. Salomaa: *Formal Languages* (1973), Academic Press, New York.
16. J. van Leeuwen: A generalization of Parikh's theorem in formal language theory, pp. 17–26 in: J. Loeckx (ed.): *2nd ICALP*, Lect. Notes in Comp. Sci. **14** (1974), Springer-Verlag, Berlin, etc.
17. H. Vogler: The OI-hierarchy is closed under control, *Inform. Comp.* **78** (1988) 187–204.