HYPER-AFL's AND ETOL SYSTEMS

by

Per Ahlgren Christensen

DAIMI PB-35
August 1974

Institute of Mathematics University of Aarhus
DEPARTMENT OF COMPUTER SCIENCE
Ny Munkegade - 8000 Aarhus C - Denmark
Phone 06-12 83 55

Hyper-AFL's and ETOL-systems

Introduction:

This paper deals with relations between substitutions and parallel rewriting in the sense of Lindenmayer-systems. We are especially interested in iterated substitution, which was introduced by Jan van Leeuwen [10] and Arto Salomaa [12], and which is a generalization of the EOL- and the ETOL-system. In a natural way these iterated substitutions lead to the notion of a hyper-AFL, and it will be proved that the family of ETOL-languages is the smallest hyper-AFL.

1. DEFINITIONS

We will first define the different types of Lindenmayer-systems, which will be used in what follows.

Definition 1.1

An extended table Lindenmayer-system without interaction (ETOL-system) is a 4-tuple $G = (V_N, V_T, T, \sigma)$, where

(i) V_N and V_T are finite disjoint sets, the nonterminal and terminal alphabets,

(ii) T is a finite nonempty set of tables, $T = \{t_1, \ldots, t_n\}$, where $n \geq 1$ and for $i = 1, \ldots, n$: $t_i \subseteq (V_N \cup V_T) \times (V_N \cup V_T)^*$, t_i is finite and t_i is complete, i.e. $\forall a \in (V_N \cup V_T) \exists w \in (V_N \cup V_T)^*$: $(a, w) \in t_i$. Instead of $(a, w) \in t_i$ we will often write $a \to w \in t_i$ and call it a production in t_i.

(iii) $\sigma \in (V_N \cup V_T)^+$ is named the axiom of G.

Definition 1.2

An ETOL-system $G = (V_N, V_T, T, \sigma)$ is called

(i) a TOL-system if $V_N = \emptyset$

(ii) an EPTOL-system if for each table t in T $t \subseteq (V_N \cup V_T) \times (V_N \cup V_T)^+$

(iii) a PTOL-system if it is TOL and EPTOL

(iv) an EOL-system if $T = \{t_1\}$, i.e. T consists of exactly one table

(v) an OL-system if it is EOL and TOL

(vi) an EPOL-system if it is EOL and EPTOL

(vii) a POL-system if it is OL and EPOL.

Definition 1.3

If $G = (V_N, V_T, T, \sigma)$ is an ETOL-system and if $a_1, \ldots, a_k \in V_N \cup V_T$ and $y_1, \ldots, y_k \in (V_N \cup V_T)^*$ and if there exists a t in T such that $(a_j, y_j) \in t$ for $j = 1, \ldots, k$ then we will write $a_1 \ldots a_k \underset{G}{\Rightarrow} y_1 \ldots y_k$. The transitive reflexive closure of $\underset{G}{\Rightarrow}$ is denoted by $\overset{*}{\underset{G}{\Rightarrow}}$.

The language generated by G is denoted by $L(G)$ and defined as $L(G) = \{w \in V_T^* \mid \sigma \overset{*}{\underset{G}{\Rightarrow}} w\}$.

If $X \in \{\lambda, ET, EPT, E, EP, T, PT, P\}$ we will call a language generated by an XOL-system an XOL-language. The family of all XOL-languages is also denoted by XOL.

Note, if $X \in \{ET, E, EP\}$ we may assume, without loss of generality, that the axiom of the XOL-system is a nonterminal letter, and furthermore, the tables need not fulfil the completeness condition in (ii) of def. 1.1.

We will use the following names for well.known families of languages:

F	- the finite languages
F_0	- the finite λ-free languages
R	- the regular languages
R_0	- the regular λ-free languages
CF	- the context-free languages
CS	- the context-sensitive languages.

In the sequel we will use the following notion of a synchronized version of an ETOL-system:

Definition 1.4

Given an ETOL-system $G = (V_N, V_T, T, \sigma)$ then define the ETOL-system $G' = (V_N', V_T, T', \sigma')$ by :
$V_N' = V_N \cup \{a' \mid a \in V_T\} \cup \{¢\}$, where a' and $¢$ are new symbols.

Define the homomorphism $h : (V_N \cup V_T)^* \to V_N'^*$ by

$$h(a) = \begin{cases} a \text{ if } a \in V_N \\ a' \text{ if } a \in V_T \end{cases}$$

If $T = \{t_1, \ldots, t_n\}$ then $T' = \{t_1', \ldots, t_n'\}$ where for $i = 1, \ldots, n$:

$$t_i' = \{(h(a), h(w)) \mid (a, w) \in t_i\} \cup \{(a, ¢) \mid a \in V_T \cup \{¢\}\}$$
$$\cup \{(a', a) \mid a \in V_T\}.$$

The axiom of G' is $\sigma' = h(\sigma)$. It is now obvious that $L(G) = L(G')$. In deriving a terminal word in G', all the terminal letters are produced in the last step. That is the reason for calling G' a synchronized version of G.

2. ITERATION-GRAMMARS

We will start this section by listing some definitions and theorems - without proving them - from A. Salomaa [12].

Definition 2.1

Let K be a non-trivial family of languages. A K-iteration grammar is a 4-tuple $G = (V_N, V_T, S, U)$, where V_N and V_T are finite disjoint sets (the nonterminal and the terminal alphabets). $V = V_N \cup V_T$ is called G's alphabet, $S \in V^+$, S is G's axiom. $U = \{\mathcal{T}_1, \ldots, \mathcal{T}_n\}$ is a finite set of K-substitutions defined on V^* such that for each a in V and for each $\mathcal{T}$ in U: $\mathcal{T}(a) \subseteq V^*$.

The language generated by such a grammar is denoted by L(G) and defined as:

$$L(G) = (\mathcal{T}_{i_1} \ldots \mathcal{T}_{i_k} (S)) \cap V_T^*,$$

where the union is taken over all integers $k \geq 0$ and all k-tuples $(i_1, \ldots, i_k)$ with $1 \leq i_j \leq n$ for $j = 1, \ldots, k$.

Definition 2.2

The family of all languages generated by K-iteration grammars is denoted by K_{iter}.

If $n \geq 1$ is an integer then the family of all languages generated by such K-iteration grammars, where U consists of at most n K substitutions, is denoted by $K_{iter}^{(n)}$.

Theorem 2.3

If K is a non-trivial family of languages such that

i) $\exists L \in K: L \neq \emptyset \wedge L \neq \{\lambda\}$

ii) K is closed under finite substitution,

iii) K is closed under intersection with regular sets,

and if K is a language generated by a λ-free K-iteration grammar (i.e., $\forall a \in V \ \forall \mathcal{T} \in U: \lambda \notin \mathcal{T}(a)$), and if h is an arbitrary homomorphism, then $h(L)\setminus\{\lambda\}$ is also generated by a λ-free K-iteration grammar with the same number of substitutions.

Corollary 2.4

Assume K is a nontrivial family of languages fulfilling i), ii) and iii) of Theorem 2.3.

If L is generated by a K-iteration grammar, then $L\setminus\{\lambda\}$ is generated by a λ-free K-iteration grammar with the same number of substitutions.

The next lemma shows that these iteration-grammars are extensions of the usual Lindenmayer-systems.

Lemma 2.5

$F^{(1)}_{iter}$ = EOL and F_{iter} = ETOL.

Proof

Let $G = (V_N, V_T, T, \sigma)$ be an ETOL-system. Define the F-iteration-grammar $G' = (V_N, V_T, \sigma, U)$, where V_N, V_T and σ are as in G, and if $T = \{t_1, \ldots, t_n\}$ then $U = \{\mathcal{T}_1, \ldots, \mathcal{T}_n\}$, where for each a in $V_N \cup V_T$ and each i $\mathcal{T}_i(a) = \{w \mid (a, w) \in t_i\}$. Then obviously $L(G) = L(G')$.

Therefore EOL $\subseteq F^{(1)}_{iter}$ and ETOL $\subseteq F_{iter}$.

Now if $G = (V_N, V_T, S, U)$ is a F-iteration grammar, then define the the ETOL-system $G' = (V_N \cup \{\$\}, V_T, T, S)$, where V_N, V_T and S are as in G, \$ is a new symbol and if $U = \{\mathcal{T}_1, \ldots, \mathcal{T}_n\}$ then $T = \{t_1, \ldots, t_n\}$, where t_i is defined by $t_i = \{(a, w) \mid w \in \mathcal{T}_i(a)\} \cup \{(a, \$) \mid \mathcal{T}_i(a) = \emptyset\} \cup \{(\$, \$)\}$.

Since \$ is a new non-terminal symbol, which only derives \$, the only task of \$ is to block the words in which \$ appears from being terminal, and this obviously simulates $\mathcal{T}_i(a) = \emptyset$.

Therefore $L(G) = L(G')$ and the lemma is proved.

From the proof of this lemma it is immediately seen that: EPTOL is the family of languages generated by λ-free F-iteration grammars, and EPOL is the family of languages generated by λ-free F-iteration grammars with one substitution. Thus, since F fulfills i), ii), and iii) in Theorem 2.3 we have proved that:

a) $L \in ETOL \Leftrightarrow L \setminus \{\lambda\} \in EPTOL$

b) $L \in EOL \Leftrightarrow L \setminus \{\lambda\} \in EPOL$

<u>Definition 2.6</u>

A family of languages is said to be <u>quasoid</u> iff

(1) K is closed under finite substitution

(2) K is closed under intersection with regular sets

(3) $R \subseteq K$

<u>Note</u>: Because of (1) and (2), (3) could be exchanged by (3)': $\{a\}^* \in K$.

Finally we mention the general theorem about iteration grammars from [12].

<u>Theorem 2.7</u>

If K is a quasoid then $K^{(1)}_{iter}$ and K_{iter} are full AFL's.

Thus, since full AFL's obviously are quasoids families like $R^{(1)}_{iter}$, R_{iter} and CF_{iter} are full AFL's.

3. SUBSTITUTIONS INTO LINDENMAYER - LANGUAGES.

In this section we will pay attention to families of languages in the form LMOL or LMTOL, where M stands for substitution into and L is some family of languages i. e. LMTOL = $\{\mathcal{T}(L) \mid L \in TOL,$ is a L-substitution$\}$. These families were introduced by Culik and Opatrný [in 2 and 3]. We will begin with a general theorem about these families.

Theorem 3. 1.

If K is a quasoid which is closed under regular substitution then KMOL and KMTOL are full AFL's.

proof:

(i) Since $R \subseteq K$ and $\{a\} \subseteq OL \subseteq TOL$ we conclude: $R \subseteq KMOL$, $R \subseteq KMTOL$.

(ii) Let $L_1, L_2 \in KMTOL$, i. e. there exist TOL-systems $G_i = (\emptyset, V_T^i, t_1^i, \ldots\ldots, t_{n_i}^i\}, \sigma^i)$ for i = 1, 2 and K-substitutions $\mathcal{T}_1$ and $\mathcal{T}_2$ such that $L_1 = \mathcal{T}_1(L(G_1))$ and $L_2 = \mathcal{T}_2(L(G_2))$. We can assume, without loss of generality, $V_T^1 \cap V_T^2 = \emptyset$. Let S be a new symbol and assume for instance $n_1 \leq n_2$. To prove $L_1 \cup L_2 \in KMTOL$ define the TOL-system $G = (\emptyset, \{S\} \cup V_T^1 \cup V_T^2, \{t_1, \ldots\ldots\ldots t_{n_2}\}, S)$. Where the table t_i consists of the productions:
if $1 \leq i \leq n_1$ then: $S \to \sigma^1$ and $S \to \sigma^2$ plus all productions in t_i^1 and t_i^1 and t_{i2}^2.
if $n_1 \leq i \leq n_2$ then: $S \to \sigma^1$ and $S \to \sigma_2^2$ plus all productions in t_i^2 and $a \to a$ for each a in V_T^1.
Define the substitution $\mathcal{T}$ by: $\mathcal{T}(S) = \emptyset$
$\forall a \in V_T^1: \mathcal{T}(a) = \mathcal{T}_1(a)$
$\forall a \in V_T^2: \mathcal{T}(a) = \mathcal{T}_2(a)$
Then it is obvious that $L_1 \cup L_2 = (L(G)) \in KMTOL$, and since the number of tables in G is n_2 then $L_1 \cup L_2 \in KMOL$ if $L_1, L_2 \in KMOL$.
To prove $L_1{}^* \in KMOL$ if $L_1 \in KMOL$, we define the OL-system $G = (\emptyset, \{S, S_1\} \cup V_T^1, \{t\}, S)$, where t consists of the productions: $S \to \lambda$; $S \to S\,S_1$; $S_1 \to \sigma^1$ plus all productions from t_1^1.

The substitution $\mathcal{T}$ is defined by $\mathcal{T}(a) = \emptyset$ if $a \in \{S, S_1\}$; $\mathcal{T}(a) = \mathcal{T}_1^1(a)$ if $a \in V_T^1$. Then obviously $L_1{}^* = \mathcal{T}(L(G)) \in$ KMOL.

If $L_1 \in$ KMTOL we define the TOL-system $G = (\emptyset, \{S, S_1\} \cup V_T^1 \cup \overline{V_T^1}, \{t_1, \ldots, t_{n_1+1}\}, S)$, where $\overline{V_T^1} = \{\bar{a} \mid a \in V_T^1\}$, Define the homomorphism h by $\forall a \in V_T^1 : h(a) = \bar{a}$.

For $1 \leq i \leq n_1$ t_i consists of the productions: $S \to S$; $S_1 \to S_1$ and for each $a \in V_T^1$ $a \to a$ and $\bar{a} \to h(w)$ iff $a \to w \in t_i^1$. t_{n_1+1} consists of the productions:

$S \to \lambda$; $S \to S S_1$; $S_1 \to S_1$; $S_1 \to h(\sigma 1)$ for each $a \in V_T^1$: $a \to a$ and $\bar{a} \to a$.

The substitution $\mathcal{T}$ is defined by

$$\mathcal{T}(a) = \begin{cases} \emptyset & \text{if } a \in \{S, S_1\} \cup \overline{V_T^1} \\ \mathcal{T}_1(a) & \text{if } a \in V_T^1 \end{cases}$$

Then it is obvious that $L_1{}^* = \mathcal{T}(L(G)) \in$ KMTOL. Thus we have so far proved that both KMOL and KMTOL are closed under $\cup$ and $*$.

(iii) To prove that both families are closed under intersection with regular sets let $G = (\emptyset, L, \{t_1, \ldots, t_n\}, \mathcal{T})$ be a TOL-system, let $\mathcal{T}$ be a K-substitution and finally let $M = (Q, \Sigma_1, \delta, q_0, F)$ be a finite automaton. Now define the TOL-system $G' = (\emptyset, \{\$, S\} \cup Q \times \Sigma \times Q, \{t_1', \ldots, t_n'\}, S)$, where S and \$ are new symbols. If G is an OL-system G' is too.

Assume $\sigma = a_{i_1} \ldots a_{i_k}$, where $a_{i_j} \in \Sigma$ for $j = 1, \ldots, k$.

For $1 \leq i \leq n$ t_i' consists of the productions:

$S \to (q_0, a_{i_1}, q_{i_1})(q_{i_1}, a_{i_2}, q_{i_2}) \ldots \ldots (q_{i_{k-1}}, a_{i_k}, q_{i_h})$ for all possible choices of $q_{i_1}, \ldots, q_{i_{k-1}}$ in Q and q_{i_k} in F.

And if $a \to b_1 \ldots b_n \in t_i$ with $n \geq 1$ and $b_1, \ldots, b_n \in \Sigma$ then:

$(q_i, a, q_j) \to (q_i, b_1, q_{i_1})(q_{i_1}, b_2, q_{i_2}) \ldots \ldots (q_{i_{n-1}}, b_n, q_j)$ for all possible choices of $q_i, q_j, q_{i_1}, \ldots \ldots q_{i_{n-1}}$ in Q.

Finally if $a \to \lambda \in t_i$ then:

for each q_i, q_j in Q:

if $i \neq j$ then $(q_i, a, q_j) \to \$$

if $i = j$ then $(q_i, a, q_j) \to \lambda$

plus the production $\$ \to \$$. Define the regular sets $R(q_i, q_j)$ for each q_i and q_j in Q by:

$R(q_i, q_j) = \{ w \in \Sigma_1{}^* \mid \delta(q_i, w) = q_j \}$.

Now we can define the K-substitution $\mathcal{T}'$ by: $\mathcal{T}'(\$) = \emptyset$

$\forall\, q_i, q_j \in Q\ \forall\, a \in \Sigma_1$:

$\mathcal{T}'((q_i, a, q_j)) = \mathcal{T}(a) \cap R(q_i, q_j)$. Then obviously: $\mathcal{T}(L(G)) \cap T(M) = \mathcal{T}'(L(G')) \in$ KMTOL. Therefore we have proved that both KMOL and KMTOL are closed under intersection with regular sets.

(iv) Finally since K is closed under regular substitution then it is obvious that KMOL and KMTOL also are closed under regular substitution.

(i), (ii), (iii), and (iv) proves the theorem.

Corollary 3.2

If K is a full AFL then both KMOL and KMTOL are full AFL's.

proof:

Obvious from 3.1 since a full AFL is a quasoid.

We have thus proved that families of languages like RMOL and RMTOL are full AFL's. Since full AFL's are closed under regular substitution RMOL (RMTOL) is obviously the smallest full AFL containing OL (TOL). Since each EOL (ETOL) - language is obtained from an OL-(TOL)- language by intersection with a regular set, it is obvious that RMOL (RMTOL) is the smallest full AFL containing EOL (ETOL).

In fact we can prove a little stronger theorem:

Lemma 3.3.

RMOL (RMTOL) is the smallest AFL containing OL or EOL (TOL or ETOL).

proof:

As mentioned above the smallest AFL containing OL (TOL) is the same as the smallest AFL containing EOL (ETOL). Since AFL's are closed under R_0-substitution, this smallest AFL will contain $R_0M(EOL) = R_0M\,F^{(1)}_{iter}$ $(R_0M(ETOL) = R_0\,M\,F_{iter})$.

Thus to prove the lemma we just need to prove $RMOL \subseteq R_0 MF^{(1)}_{iter}$ ($RMTOL \subseteq R_0 MF_{iter}$). Let $L \in OL(TOL)$ and let ρ be a regular substitution, since $OL \leq EOL = F^{(1)}_{iter}$ ($TOL \leq ETOL = F_{iter}$)
we know from corollary 2.4 that $L \setminus \{\lambda\}$ is generated by a λ-free $F^{(1)}$ (F) - iteration - grammar.
Define the finite substitutions $\mathcal{T}_1$ and $\mathcal{T}_2$ by:
if $L \leq \Sigma^*$ then let $\bar{\Sigma} = \{\bar{a} \mid a \in \Sigma\}$ be a set of new symbols and:

$$\forall a \in \Sigma: \mathcal{T}_1(a) = \begin{cases} \{a, \bar{a}\} & \text{if } \lambda \in \rho(a) \\ \{a\} & \text{if } \lambda \notin \rho(a) \end{cases}$$

$$\mathcal{T}_2(a) = \{a\} \text{ and } \mathcal{T}_2(\bar{a}) = \{\lambda\}.$$

It is obvious that $\mathcal{T}_1(L \setminus \{\lambda\})$ is generated by a λ-free $F^{(1)}$ (F)-iteration-grammar, since $L \setminus \{\lambda\}$ is. From theorem 2.3 follows that $\mathcal{T}_2(\mathcal{T}_1(L \setminus \{\lambda\})) \setminus \{\lambda\} \in F^{(1)}_{iter} = EOL$ ($F_{iter} = ETOL$) and thus $\mathcal{T}_2(\mathcal{T}_1(L)) \in EOL$ (ETOL).
Define the R_0 - substitution $\bar{\rho}$ by:

$$\forall a \in \Sigma : \bar{\rho}(a) = \rho(a) \setminus \{\lambda\}$$

Thus $\bar{\rho}(\mathcal{T}_2(\mathcal{T}_1(L)) \in R_0M(EOL)$ ($R_0M(ETOL)$) but obviously $\rho(L) = \bar{\rho}(\mathcal{T}_2(\mathcal{T}_1(L)))$ and the lemma is proved.

Lemma 3.4.

RMOL is not closed under substitution

proof:

Let $L_1 := \{a^{2^n} \mid n \geq o\}$ and $L_2 := \{a b^{2^n} \mid n \geq o\}$ then $L_1, L_2 \in OL \subseteq RMOL$. Define the substitution $\mathcal{T}$ by $\mathcal{T}(a) = L_2$, if we can prove $\mathcal{T}(L_1) \notin RMOL$ the lemma is proved. Since regular languages fulfils a pumping-lemma (i.e. $\forall A \in R \ \exists n_A > O \ \forall w \in A: |w| \geq n_A \Rightarrow (\exists x, y, z: (|yz| \leq n_A) \wedge (|y| > O) \wedge (w = xyz) \wedge (\forall i \geq o : x y^i z \in A)))$ the n because of the powers of 2 $\mathcal{T}(L_1) \notin RMOL \setminus FMOL$, therefore it is enough to prove $\mathcal{T}(L_1) \notin FMOL$.
Define the finite substitution $\mathcal{T}'$ by:
$\mathcal{T}'(a) = \{a\}$ and $\mathcal{T}'(b) = \{\lambda, b\}$, and define the hormorphism h by: $h(a) = a$, $h(\sigma) = \lambda$, then $b^* \mathcal{T}'((L_1)) = h^{-1}(L_1)$, Herman [8] has proved that $h^{-1}(L_1) \notin EOL$, but the proof shows that $\mathcal{T}'((L_1)) \notin EOL$, and since EOL is closed under finite substitution and intersection with regular sets the lemma is proved.

4. HYPER - AFL's.

The notions of a hyper $^{(1)}$- AFL, and a hyper - AFL were introduced in Jan van Leeuwen [10] and A. Salomaa [12].

Definition 4. 1.

If K is a quasoid such that $K^{(1)}_{iter}$ = K then K is said to be a hyper$^{(1)}$-AFL.

If K is quasoid such that K_{iter}= K then K is said to be a hyper-AFL.

From theorem 2. 7 follows immediately that each hyper$^{(1)}$-AFL and each hyper-AFL are full AFL's.

The notion of a super-AFL was introduced in Greibach [6].

Definition 4. 2.

A family of languages is said to be a super-AFL, iff

(i) $\exists$ L $\in$ K $\exists$ w $\in$ L: $|w| > 1$.
i. e. there exists a language in K with a word of lenght greater than one.

(ii) K is closed under intersection with regular sets.

(iii) For any language L in K and any language L_1, satisfying. $\forall$ w$\in L_1$: $|w| \leq 1$, $L \cup L_1 \in K$.

(iv) If L $\in$ K and if $\mathcal{T}$ is a K-substitution, such that L $\leq \Sigma_1$ * and $\mathcal{T}$ is defined on Σ_1 * and for each a in Σ_1 : a $\in \mathcal{T}$(a). Moreover if $\mathcal{T}(L) \leq \Sigma_2$* we extend $\mathcal{T}$ to be defined on $(\Sigma_1 \cup \Sigma_2)^*$ by for each $b \in \Sigma_2 \setminus \Sigma_1$ $\mathcal{T}(b) = \{b\}$ (because of (i) and (ii) $\mathcal{T}$ is still a K-substitution). We now define $\mathcal{T}^{o}(L) = L$ and for each n $\geq$ o $\mathcal{T}^{n+1}(L) = \mathcal{T}(\mathcal{T}^{n}(L))$
Then $\bigcup\limits_{n=}^{\infty} \mathcal{T}^{n}(L) \in K$.

In [6] the following theorem is also proved.

Theorem 4.3.

Each super-AFL is a full substitution-closed AFL (i.e. K is a full AFL and K M K = K).

Remark:

This theorem shows that the difference between a super-AFL and a hyper$^{(1)}$-AFL is, that in the definition of a super-AFL we just require iteration-closure under such substitutions $\mathcal{T}$ where for each a $a \in \mathcal{T}(a)$. This difference is obviously due to the difference between parallel and non-parallel rewriting.

Theorem 4.4.

Each hyper$^{(1)}$-AFL is a super-AFL.

proof:

Let K be a hyper$^{(1)}$-AFL. Then we know K is a full AFL and therefore K satisfies (i), (ii), and (iii) in definition 4.2.
To prove that (iv) is satisfied let $L \in K$, $L \le \Sigma_1{}^*$ and let $\mathcal{T}_1$ be the extended K-substitution (i.e. if $\mathcal{T}(L) \subseteq \Sigma_2{}^*$ then $\mathcal{T}_1$ is defined on $(\Sigma_1 \cup \Sigma_2)^*$) still satisfying that for each a : $a \in \mathcal{T}_1(a)$.
Let S be a new symbol, define $\bar{\Sigma} = \{ \bar{a} \mid a \in \Sigma_1 \cup \Sigma_2\}$ as a set of new symbols, finally define the homomorphism h by: $\forall a \in \Sigma_1 \cup \Sigma_2$: $h(a) = \bar{a}$
Now define the $K^{(1)}$- iteration-grammar $G = (\{S\} \cup \bar{\Sigma}, \Sigma_1 \cup \Sigma_2, S, \{\mathcal{T}\})$, where $\mathcal{T}(S) = h(L) \in K$, since K is a full AFL, for each a in $\Sigma_1 \cup \Sigma_2$: $\mathcal{T}(\bar{a}) = h(\mathcal{T}_1(a)) \cup \{a\}$ and $\mathcal{T}(a) = \varphi$
Since $\mathcal{T}_1$ is a K-substitution and K is a full AFL $\mathcal{T}$ is obviously a K-substitution and $\bigcup_{n=0}^{\infty} \mathcal{T}_1{}^n(L) = L(G) \in K^{(1)}_{iter} = K$, which proves the theorem.

Corollary 4.5

Each hyper-AFL is a super-AFL.

proof:

Each hyper-AFL is obviously a hyper$^{(1)}$-AFL.

Corollary 4.6

Each hyper$^{(1)}$-AFL and each hyper-AFL are substitution-closed.

proof:

Immediate from theorem 4.4, corollary 4.5 and theorem 4.3.

Theorem 4.7.

$ETOL = ETOL^{(1)}_{iter} = ETOL_{iter}$

proof:

Let $L \in ETOL$, $L \leq \Sigma^*$, let S be a new symbol i.e. $S \notin \Sigma$.
Define the $ETOL^{(1)}$-iteration-grammar $G = (\{S\}, \Sigma, S, \{\mathcal{T}\})$, where $\mathcal{T}(S) = L$ and for each a in $\Sigma: \mathcal{T}(a) = \{a\}$. Then obviously $L = L(G) \in ETOL^{(1)}_{iter}$. Thus $ETOL \subseteq ETOL^{(1)}_{iter} \subseteq ETOL_{iter}$

To prove $ETOL_{iter} \subseteq ETOL$:
let $G = (V_n, V_T, S, U)$ be an ETOL-iteration grammar with $U = \{\mathcal{T}_1, \ldots . \mathcal{T}_n\}$, where each $\mathcal{T}_j$ is an ETOL-substitution defined on $V_N \cup V_T = \{a_1, \ldots\ldots, a_m\}$ such that for each i and each j $\mathcal{T}_j(a_i) \subseteq (V_N \cup V_T)^*$.
Assume that $\mathcal{T}_j(a_i) = L(G_{i,j})$, where $G_{i,j} = (V_N^{i,j}, V_T \cup V_N, T_{i,j}, S_{i,j})$ are synchronized versions of ETOL-systems, where we obviously without loss of generality can assume that the nonterminal alphabets $V_N^{i,j}$ are pairwise disjoint.
We define a new ETOL-system $G' = (V_N', V_T, T', \bar{\bar{S}})$, where $V_N' = \{\$\} \cup \bigcup_{i,j} (V_N^{i,j} \cup \{\bar{S}_{i,j}\}) \cup \bar{V}_T \cup \bar{V}_N \cup \bar{\bar{V}}_T \cup \bar{\bar{V}}_N$, where $\$$ and all $\bar{\bar{S}}_{i,j}$ are new symbols.
$\bar{V}_X = \{\bar{a} \mid a \in V_X\}$ and $\bar{\bar{V}}_X = \{\bar{\bar{a}} \mid a \in V_X\}$ for $X = N, T$ are sets of new symbols.
If X is a string of symbols $X = b_1, \ldots\ldots b_m$, then we write $\bar{X} = \bar{b}_1 \ldots\ldots \bar{b}_n$. and $\bar{\bar{X}} = \bar{\bar{b}}_1 \ldots\ldots \bar{\bar{b}}_m$.
The axions of G is defined as $\bar{\bar{S}}$ in this way.
Finally T' consists of the tables:
t_o: $\bar{\bar{a}}_i \to \bar{S}_{i,j}$ for each i and each j. $A \to \$$ for each other symbol A.
For $1 \leq j \leq n$ we have the table:
t_j : $\bar{S}_{i,j} \to \bar{S}_{i,j}$; $\bar{S}_{i,j} \to S_{i,j}$ for each i. $\bar{a}_i \to \bar{\bar{a}}_i$; $\bar{\bar{a}}_i \to \bar{\bar{a}}_i$ for each a_i in $V_N \cup V_T$. $A \to \$$ for each other symbol A. For $1 \leq j \leq n$ and $1 \leq i \leq m$ we have the set of tables:

$\tilde{T}_{i,j}$: which consists of all tables from $T_{i,j}$ where we have changed the table with terminal productions ($G_{i,j}$ is a synchronized version of an ETOL-system) to produce barred terminals instead. In all the tables we add the productions: $\bar{\bar{a}}_k \to \bar{\bar{a}}_k$ for each a_k in $V_N \cup V_T$.

$\overline{S}_{k,j} \to \overline{S}_{k,j}$ for each k $\quad A \to \$$ for each other symbol A.

Finally there is the table with terminal productions:

$\overline{\overline{a}} \to a$ for each $a \in V_T$

$A \to \$$ for each other symbol A.

The claim is now that $L(G) = L(G')$.

$\subseteq$: If $S \in V_T^*$, we can obviously derive S from $\overline{\overline{S}}$ in G' by using the table with terminal production. Now assume $t \geq 1$; $1 \leq i_1, \ldots, i_t \leq n$ and $X_1 \in \mathcal{T}_{i_1}(S)$, $X_2 \in \mathcal{T}_{i_2}(X_1), \ldots\ldots, X_t \in \mathcal{T}_{i_t}(X_{t-1})$ and assume $X_t \in V_T^*$, then we want to prove that $X_t \in L(G')$

If $S = a_{l_1} \ldots\ldots a_{l_k}$ then:

$$\overline{\overline{S}} = \overline{\overline{a}}_{l_1} \ldots\ldots \overline{\overline{a}}_{l_k}$$

$$\overset{t_o}{\underset{G'}{\Rightarrow}} \overline{S}_{l_1, i_1} \; \overline{S}_{l_2, i_1} \ldots\ldots \overline{S}_{l_k, i_1}$$

$$\overset{t_{i_1}}{\Rightarrow} S_{l_1, i_1} \; \overline{S}_{l_2, i_1} \ldots\ldots \overline{S}_{l_k, i_1}$$

$$\tilde{T}_{l_1, i_1} : \overset{*}{\underset{G'}{\Rightarrow}} \overline{X}_{1, l_1} \; \overline{S}_{l_2, i_1} \ldots\ldots \overline{S}_{l_k, i_1}$$

$$\overset{t_{i_1}}{\underset{G'}{\Rightarrow}} \overline{\overline{X}}_{1, l_1} \; S_{l_2, i_1} \ldots\ldots \overline{S}_{l_h, i_1}$$

$$\tilde{T}_{l_2, i_1} : \overset{*}{\underset{G'}{\Rightarrow}} \overline{\overline{X}}_{1, l_1} \; \overline{X}_{1, l_2} \ldots\ldots \overline{S}_{l_k, i_1}$$

$$\overset{t_{i_1}}{\underset{G'}{\Rightarrow}} \overline{X}_{1, l_1} \; \overline{\overline{X}}_{1, l_2} \ldots\ldots \overline{S}_{l_k, i_1}$$

$$\overset{*}{\Rightarrow} \overline{\overline{X}}_{1, l_1} \; \overline{\overline{X}}_{1, l_2} \ldots\ldots \overline{\overline{X}}_{1, l_k} = \overline{\overline{X}}_1$$

(in the same way as $\overline{\overline{S}} \overset{*}{\underset{G'}{\Rightarrow}} \overline{\overline{X}}_1$)

$$\overset{*}{\underset{G'}{\Rightarrow}} \overline{\overline{X}}_2$$

$$\overset{*}{\underset{G'}{\Rightarrow}} \overline{\overline{X}}_t$$

terminal table: $\overset{\Rightarrow}{G'}$ X_t

The reason why $\overline{\overline{X}}_1$ is derivable in the stated way is simply that $X_1 \in \mathcal{T}_{i_1}(S) = \mathcal{T}_{i_1}(a_{l_1} \ldots\ldots\ldots a_{l_k}) = \mathcal{T}_{i_1}(a_{l_1}) \ldots \mathcal{T}_{i_1}(a_{l_k})$ and thus $X_1 = X_{1,l_1} \ldots\ldots\ldots X_{1,l_k}$ where $X_{1,l_j} \in \mathcal{T}_{i_1}(a_{l_j})$ for $j=1,\ldots,k$. Therefore $\overline{X}_{1,l_j}$ is derivable from S_{l_j,i_1} via the tables from $\tilde{T}_{l_j,i_1}$.

In exactly the same way $\overline{\overline{X}}_2$ is derived from $\overline{\overline{X}}_1$ and so forth. Thus $X_t \in L(G')$. And the inclusion is proved.

$\supseteq$: Assume that $\overline{\overline{S}} \overset{*}{\underset{G'}{\Rightarrow}} X$, where $X \in V_T^*$, then obviously $\overline{\overline{S}} \overset{*}{\underset{G'}{\Rightarrow}} \overline{\overline{X}} \overset{*}{\underset{G'}{\Rightarrow}} X$. By definition of T' the only table, which is able to change a double-barred word (without producing terminals from which only strings with \$-symbols are derivable) is t_o.

In t_o we choose for each double-barred symbol $\overline{\overline{a}}_k$ a substitution $\mathcal{T}_j$ by rewriting it to $\overline{S}_{k,j}$. As usual if a \$-symbol is introduced in a string, we are never able to derive a terminal word. Therefore the only possibility to rewrite $\overline{S}_{k,j}$ to anything different from $\overline{S}_{k,j}$ is to use the table t_j, in doing this we are forced to choose the same substitution $\mathcal{T}_j$ for each double-barred symbol, if we want to avoid producing \$-symbols.

The table t_j can only change a $\overline{S}_{k,j}$ by changing it to $S_{k,j}$, and now we are forced (to avoid producing \$) to rewrite this via the tables in $\tilde{T}_{k,j}$ until a single-barred word is produced, this word with bars deleted belongs therefore to $\mathcal{T}_j(a_k)$.

At this point we are forced to use t_j, which double-bars the single-barred string. We are forced to go on in this way until all $\overline{S}_{k,j}$'s are rewritten through $S_{k,j}$ to a single-barred word in $\mathcal{T}_j(a_k)$, and then this word is immediately double-barred.

We have now produced a new double-barred word, which with the bars deleted is an element of $\mathcal{T}_j$(the double-barred word, which we started to rewrite via t_o, with the bars deleted). Now we of course have the possibility to start all over again via t_o, and the only other possibility is if the double-barred word is an element of $\overline{\overline{V}}_T^*$, then we can choose the table with terminal productions, which eliminates the double-bars and from here we are only able to derive strings in $\{\$\}^*$. Thus we have proved that there exists $t \geq o$ such that $X \in \mathcal{T}_{i_t} \ldots\ldots\ldots\ldots \mathcal{T}_{i_1}(S) \cap V_T^* \subseteq L(G)$. And the theorem is proved.

Corollary 4.8.

ETOL is a hyper$^{(1)}$- AFL and a hyper - AFL.

proof:

because of theorem 4.7 we only need to prove that ETOL is a quasoid. The proof of this is straight-forward, but we will omit this, since it is already well-known, G. Rosenberg [11], that ETOL is a full AFL, and therefore of cource a quasoid.

Corollary 4.9.

If K is a family of languages such that : $F \subseteq K \subseteq ETOL$ then K_{iter} =ETOL.

proof:

according to lemma 2.5 F_{iter} = ETOL, thus $ETOL = F_{iter} \subseteq K_{iter} \subseteq ETOL_{iter}$ = ETOL.

We have thus proved that:

$$ETOL = F_{iter} = R_{iter} = CF_{iter} = EOL_{iter} = RMOL_{iter} = ETOL_{iter}$$

since we know that

$F \subseteq R \subseteq CF \subseteq EOL \subseteq RMOL$.

Corollary 4.10.

ETOL is the smallest hyper-AFL.

proof:

Each hyper-AFLK is a full AFL and therefore $K \supseteq R$ and thus K =

$K \supseteq R_{iter} = ETOL$

From corollary 4.6 and corollary 4.8 follows:

Corollary 4.11.

ETOL is closed under substitution.

From this corollary and from the following theorem of Greibach [7] we will be able to prove the existance of an infinite hierarchy of full AFL's containing CF and contained in CS.

Theorem 4.12.

If L is a full AFL, then $L \; M \; L = L \Leftrightarrow (L \; M \; L) \; M \; (L \; M \; L) = L \; M \; L$; i.e. L is substitution-closed iff $L \; M \; L$ is too.

Corollary 4.13.

$\forall \; n \geq 1$: $R(MOL)^n$ is a full AFL and $CF \subsetneqq EOL \subsetneqq R(MOL)^n \subsetneqq R(MOL)^{n+1} \subsetneqq ETOL \subsetneqq CS$.

proof:

from the fact that $\{a\} \in OL$ follows that for each $n \geq 1$. $R(MOL)^n \subseteq R(MOL)^{n+1}$. As mentioned before $CF \subsetneqq R(MOL)^n$ for each $n \geq 1$. From corollary 4.11 follows by induction that $R(MOL)^{n+1} = R(MOL)^n \; MOL \subseteq (ETOL) \; M \; (ETOL) = ETOL$. It is well known that $ETOL \subsetneqq CS$. Therefore we just need to prove that the inclusions $R(MOL)^n \subseteq R(MOL)^{n+1} \subseteq ETOL$ are proper.

From corollary 3.2 follows by induction that $R(MOL)^n$ is a full AFL for each $n \geq 1$.

Now define $\quad A_1 := RMOL$

and for $n \geq 1$: $\quad A_{n+1} := A_n \; M \; A_n$.

Since A_1 is a full AFL, then according to lemma 3.4 $A_1 \subsetneqq A_2$. But from this it follows from theorem 4.12 by induction that $A_n \subsetneqq A_{n+1}$ for each $n \geq 1$, since Ginsburg and Spanier in [5] proved that if L_1 and L_2 are full AFL's then so is $L_1 \; M \; L_2$, and therefore A_n is a full AFL

for each $n \geq 1$ and thus $A_n \subseteq A_{n+1}$.
But since RMOL is a full AFL and since $\{a\} \in R$: RMOL = (RMOL)MR
From this we conclude (RMOL)MOL = ((RMOL)MR)MOL = (RMOL) M
(RMOL), where the last equality holds because of the fact, which we have used before without mentioning it, that as far as we are dealing with symmetric families of languages (i. e. families closed under isomorphisms) the substitution-operator is associative. By induction we can now prove that: $\forall_n \geq 1$: RMOL $(MOL)^n$= RMOL$(M\ RMOL)^n$. Thus according to the definition of A_n then for each $n \geq 1$ there exists a $k \geq n$ such that $R(MOL)^k = A_n$.
Finally if $R(MOL)^{n_1} = R(MOL)^{n_1+1}$ then obviously for each $j \geq 1$: $R(MOL)^{n_1} = R(MOL)^{n_1+j}$.
Therefore since $A_1 \subsetneqq A_2 \subsetneqq \cdots\cdots \subsetneqq A_n \subsetneqq A_{n+1} \subsetneqq \cdots\cdots$ and since each A_n equals $R(MOL)^k$ for some $k \geq n$, there cannot exist any $n_1 \geq 1$ such that $R(MOL)^{n_1} = R(MOL)^{n_1+1}$. Since we already know that $R(MOL)^n \subseteq R(MOL)^{n+1}$ for each n the corollary is proved, since the argument also proves that $R(MOL)^{n+1}$ is not closed under substitution, which we know ETOL is.

Remark:

We have now proved the existance of a proper infinite hierarchy of full AFL's containing CF and contained in CS.

Since RMTOL is the smallest full AFL containing TOL and ETOL is a full AFL then of course RMTOL = ETOL. But since TOL $\subseteq$ ETOL and (ETOL) M(ETOL) = ETOL then for each $n \geq 1$ $R(MTOL)^n$ = ETOL, this operation doesn't give rise to an infinite hierarchy in the case of TOL-systems.

Theorem 4. 14.

$R^{(1)}_{iter}$ is not closed under substitution.

proof:

As already mentioned $L_1 = \{ a^{2^n} \mid n \geq 0\}$ and $L_2 = \{ ab^{2^n} \mid n \geq o\}$ are

OL-languages and therefore EOL = $F^{(1)}_{iter} \subseteq R^{(1)}_{iter}$ -languages. Define the substitution $\mathcal{T}$ by $\mathcal{T}(a) = L_2$, thus $\mathcal{T}(L_1) \in R^{(1)}_{iter} \; M \; R^{(1)}_{iter}$.

But $\mathcal{T}(L_1)$ cannot belong to $R^{(1)}_{iter}$, since as mentioned in the proof of lemma 3.4. $\mathcal{T}(L_1) \notin EOL = F^{(1)}_{iter}$, and infinite regular sets fulfil a pumping-lemma thus $\mathcal{T}(L_1) \notin R^{(1)}_{iter} \setminus F^{(1)}_{iter}$. We conclude $\mathcal{T}(L_1) \notin R^{(1)}_{iter}$, and the theorem is proved.

Corollary 4.15.

$R^{(1)}_{iter} \subsetneqq R_{iter} = ETOL$.

proof:

It is obvious that $R^{(1)}_{iter} \subseteq R_{iter} = ETOL$, and thus the corollary follows from theorem 4.14 and corollary 4.11.

Final remarks:

From theorem 4.14 we conclude that we can construct an infinite hierarchy of full AFL's from $R^{(1)}_{iter}$ in the same way as we did in corollary 4.13 from RMOL.

From theorem 4.14 and corollary 4.6 we conclude that $R^{(1)}_{iter}$ is not a hyper$^{(1)}$- AFL, so if K is a hyper$^{(1)}$- AFL then $K \supsetneqq R^{(1)}_{iter}$.

Which gives the following interesting question: is there a smallest hyper$^{(1)}$- AFL and if there is which?

There exists a full AFL K such that $K \subsetneqq K_{iter}$ but $K_{iter} = (K_{iter})_{iter} = (K_{iter})^{(1)}_{iter}$ and $K^{(1)}_{iter} \subsetneqq K_{iter}$ namely K=R. But this doesn't answer the interesting question: Does there exist a hyper$^{(1)}$- AFL which is not a hyper - AFL.

References:

1. Alfred V Aho:
Indexed Grammar - An extension of Context-Free Grammars - Journal of the ACM. 15. 1968.

2. K. Čulik II :
On some Families of languages related to Developmental Systems. CS - 73 - 12.

3. K. Čulik II and J. Opatrný:
Macro OL-systems. CS - 73 - 06.

4. Seymour Ginsburg and Sheila Greibach:
Abstract Families of Languages, in Studies in Abstract Families of Languages, Amer. Math. Soc. Memoir 87 (1969).

5. S. Ginsburg and Edwin H. Spanier:
Substitution in Families of Languages. Information Science vol 2. (1970).

6. S. Greibach:
Full AFL's and Nested Iterated Substitutions. Information and Control 16 (1970).

7. S. Greibach:
Chains of Full AFL's. Math. Syst. Theory. vol 4, no. 3.

8. Gabor T. Herman:
Closure Properties of some Families of Languages associated with Biological Systems.

9. John E. Hopcroft and Jeffrey D. Ullmann:
Formal Languages and their Relations to Automata. Addison-Wesley.

10. Jan van Leeuwen: F-iteration Languages. (Extended abstract).

11. G. Rozenberg: Extensions of Tables OL-systems of Languages. R. Report 53 - 1972, Department of Comp. Science.SUNY at Buffalo.

12. Arto Salomaa:
Macros, Iterated Substitution and Lindenmayer - AFL's
DAIMI PB - 18.