Substitution in Families of Languages

Seymour Ginsburg
Edwin H. Spanier

16 September 1968

Prepared for
AIR FORCE CAMBRIDGE RESEARCH LABORATORIES
OFFICE OF AEROSPACE RESEARCH, USAF
BEDFORD, MASSACHUSETTS

and

AIR FORCE OFFICE OF SCIENTIFIC RESEARCH
OFFICE OF AEROSPACE RESEARCH, USAF
ARLINGTON, VIRGINIA

Contract Monitor: Thomas V. Griffiths

Data Sciences Laboratory

Supported by
SDC's Independent Research Program
Contract FA8628-67-C-0008, Programming (Algorithmic) Languages, Project No 5632.
Task No 564205, and Grant No AF AFOSR 1203 67

Work Unit No 56320501
Qualified requestors may obtain additional copies from the Defense Documentation Center. All others should apply to the Clearinghouse for Federal Scientific and Technical Information.
The work reported herein was supported by SDC and Contract F1962867C0008, Programming (Algorithmic) Languages; Project No. 5632, Task No. 563205; Work Unit No. 56320501; and Grant No. AF-AFOSR-1203-67

SCIENTIFIC REPORT NO. 22

Substitution in Families of Languages
by
Seymour Ginsburg*
and
Edwin H. Spanier**
16 September 1968

Monitored by: Contract Monitor
Thomas V. Griffiths
Data Sciences Laboratory
R. Swanson, SRIR (AFOSR)
Prepared for
AIRC FORCE CAMBRIDGE RESEARCH LABORATORIES
OFFICE OF AEROSPACE RESEARCH, USAF
BEDFORD MASSACHUSETTS 01730
and
AIR FORCE OFFICE OF SCIENTIFIC RESEARCH
OFFICE OF AEROSPACE RESEARCH, USAF
ARLINGTON, VIRGINIA

Distribution of this document is unlimited. It may be released to the Clearinghouse, Department of Commerce, for sale to the general public.

*System Development Corporation and University of Southern California

**University of California at Berkeley and consultant, System Development Corporation
ABSTRACT

The effect of substitution in families of languages, especially AFL, is considered. Among the main results shown are the following: The substitution of one AFL into another is an AFL. Under suitable hypotheses, the AFL generated by the family obtained from the substitution of one family into another, is the family obtained from the substitution of the corresponding AFL. A condition is given for the AFL generated by the substitution closure of a family to be the substitution closure of the AFL generated by the family.
INTRODUCTION

In an earlier paper [6], the authors were led to consider the family of languages obtained from the family of linear context-free languages by iterated substitution. This in turn suggested to us a study of the substitution of one arbitrary family of languages into another and is the subject of the present work.

Recently the notion of an AFL (abstract family of languages) was introduced [3] as an abstraction of many of the formal languages of concern to computer science. In particular, it was shown in [3] that there is an intimate connection between AFL and the families of languages accepted by families of one-way nondeterministic acceptors. Thus AFL play a special role among arbitrary families of languages, at least for device theory. The theorems contained herein are concerned with the relationship of AFL and substitution. These theorems, in turn, are based on lesser results which are concerned with the relation of substitution to other operations. These lesser results are formulated in terms of arbitrary families of languages because (1) many of them are interesting in their own right and may have other applications, and (2) they isolate the difficulty inherent in the proofs of the main results.

The paper is organized into four sections. The first one is devoted to general concepts of families of languages, to a proof that under a mild condition substitution satisfies a kind of associativity, and to a formulation of the AFL properties in terms of substitution. Section two is concerned with showing that the substitution of one AFL into another is an AFL. Section three contains a proof that under suitable hypotheses the AFL generated by the family obtained from the substitution of one family into another is the family obtained from the substitution of the corresponding AFL.

Section four considers iterated substitution and the substitution closure of one family with respect to another. A special case is the substitution closure of a family. It is shown that the various substitution closures give AFL when applied to AFL, and a condition is given for the AFL generated by the substitution closure of a family to be the substitution closure of the AFL generated by the family. A condition is also given for the substitution closure of a family, not necessarily an AFL, to be an AFL. This specializes to the case of the family of linear context-free languages and implies that the substitution closure of this family is a full AFL (a result obtained by other means in [C]).

Section 1. Families of languages

In this section we review some concepts about families of languages. We also examine certain methods of constructing new families from old, especially substituting one family into another. Finally, we consider the concept of an abstract family of languages and reformulate it in terms of substitution.
Definition. A family of languages is a pair (\( \Sigma, \mathcal{L} \)), or \( \mathcal{L} \) when \( \Sigma \) is understood, where

1. \( \Sigma \) is an infinite set of symbols,
2. for each \( L \) in \( \mathcal{L} \) there is a finite set \( \Sigma_L \subseteq \Sigma \) such that \( L \subseteq \Sigma_L^* \),
3. \( L \neq \emptyset \) for some \( L \) in \( \mathcal{L} \).

Notation. Given \( \Sigma \) in \( \mathcal{L} \), \( \Sigma_L \) will denote the smallest set \( \Sigma_L \) such that \( L \subseteq \Sigma_L^* \).

Henceforth, \( \Sigma \) will always denote a given infinite set of symbols and \( \Sigma \) with a subscript a finite subset of \( \Sigma \). All symbols given or constructed will be assumed in \( \Sigma \).

We now distinguish some elementary conditions for families of languages.

Definition. A family of languages \( \mathcal{L} \) is said to be

1. symmetric if it is invariant under all permutations of \( \Sigma \).
2. \( \varepsilon \)-free if each \( L \) in \( \mathcal{L} \) is \( \varepsilon \)-free (i.e., \( \varepsilon \) is not in \( L \)).
3. nontrivial if there is some \( L \) in \( \mathcal{L} \) containing a non-\( \varepsilon \) word.

We shall be interested in various operations on languages and families of languages. We first present two operations on pairs of families.

Notation. Given families of languages \( \mathcal{L}_1 \) and \( \mathcal{L}_2 \), let

1. \( \mathcal{L}_1 \cap \mathcal{L}_2 = \{ L_1 \cap L_2 / L_1 \text{ in } \mathcal{L}_1, L_2 \text{ in } \mathcal{L}_2 \} \).
2. \( \text{Sub } (\mathcal{L}_1, \mathcal{L}_2) \) be the family obtained by substituting languages of \( \mathcal{L}_1 \) into languages of \( \mathcal{L}_2 \), i.e., the family of all sets \( \tau(L_2) \), where \( L_2 \) is in \( \mathcal{L}_2 \) and \( \tau \) is a substitution such that \( \tau(a) \) is in \( \mathcal{L}_1 \) for each \( a \) in \( \Sigma_{\mathcal{L}_2}^* \).

1. \( \Sigma_1^* \) is the free semigroup with identity \( \varepsilon \) generated by \( \Sigma_1 \), i.e., the set of all finite strings \( a_1 \ldots a_n \), each \( a_i \) in \( \Sigma_1 \). Each element of \( \Sigma_1^* \) is called a word of \( \Sigma_1^* \).
2. Let \( L \subseteq \Sigma_1^* \) and for each \( a \) in \( \Sigma_2 \) let \( L_a \subseteq \Sigma_1^* \). Let \( \tau \) be the function defined on \( \Sigma_2^* \) by \( \tau(\varepsilon) = \{ \varepsilon \} \), \( \tau(a) = L_a \) for each \( a \) in \( \Sigma_2 \), and \( \tau(a_1 \ldots a_n) = \tau(a_1) \ldots \tau(a_n) \) for each \( a_i \) in \( \Sigma_2 \) and \( \varepsilon \). Then \( \tau \) is called a substitution. \( \tau \) is extended to \( \Sigma_3^* \) by defining \( \tau(X) = \bigcup_{x \in X} \tau(x) \) for all \( X \subseteq \Sigma_3^* \).
As is evident from the title, our interest here is in substitutions in families of languages.

We next present two operations on a family of languages.

**Notation.** For each family $\mathcal{L}$ let

1. $\text{Hom}(\mathcal{L}) = \{ h(L)/L \in \mathcal{L}, h \text{ a homomorphism}\text{(3) on } L\}$, and
2. $\text{Hom}_r(\mathcal{L}) = \{ h(L)/L \in \mathcal{L}, h \text{ a restricted homomorphism on } L\text{(4)}\}$.

Clearly $\text{Hom}(\mathcal{L})$ and $\text{Hom}_r(\mathcal{L})$ are monotonically increasing in $\mathcal{L}$, and both $\mathcal{L}_1 \wedge \mathcal{L}_2$ and $\text{Sub}(\mathcal{L}_1, \mathcal{L}_2)$ are monotonically increasing in each of the families $\mathcal{L}_1$ and $\mathcal{L}_2$. For each family $\mathcal{L}$, $\text{Hom}(\mathcal{L})$ and $\text{Hom}_r(\mathcal{L})$ are symmetric, $\text{Hom}_r(\mathcal{L}) = \text{Hom}(\mathcal{L})$, and $\text{Hom}_r(\text{Hom}_r(\mathcal{L})) = \text{Hom}_r(\mathcal{L})$. If $\mathcal{L}_1$ and $\mathcal{L}_2$ are symmetric, then so is $\mathcal{L}_1 \wedge \mathcal{L}_2$. If $\mathcal{L}_1$ is symmetric, then so is $\text{Sub}(\mathcal{L}_1, \mathcal{L}_2)$ for each family $\mathcal{L}_2$.

From the definition, it is trivial that $\mathcal{L}_1 \wedge \mathcal{L}_2 = \mathcal{L}_2 \wedge \mathcal{L}_1$ and $(\mathcal{L}_1 \wedge \mathcal{L}_2) \wedge \mathcal{L}_3 = \mathcal{L}_1 \wedge (\mathcal{L}_2 \wedge \mathcal{L}_3)$. However, $\text{Sub}(\mathcal{L}_1, \mathcal{L}_2)$ need not equal $\text{Sub}(\mathcal{L}_2, \mathcal{L}_1)$.

For example, if $\mathcal{L}_1 = \{a^2\}$ and $\mathcal{L}_2 = \{L_2\}$, where $L_2 = \{a^3/a \text{ in } \Sigma_1\}$, then $\text{Sub}(\mathcal{L}_1, \mathcal{L}_2) = \{((a^3/b, a \text{ in } \Sigma_1)) \neq \text{Sub}(\mathcal{L}_2, \mathcal{L}_1) = \{(a^3b^3/a, b \text{ in } \Sigma_1)\}$. However, substitution does have the following associative properties:

**Proposition 1.1.** Let $\mathcal{L}_1$, $\mathcal{L}_2$, and $\mathcal{L}_3$ be families of languages. Then

(a) $\text{Sub}(\mathcal{L}_1, \text{Sub}(\mathcal{L}_2, \mathcal{L}_3)) \subseteq \text{Sub} \left( \text{Sub}(\mathcal{L}_1, \mathcal{L}_2), \mathcal{L}_3 \right)$.

(b) $\text{Sub}(\mathcal{L}_1, \text{Sub}(\mathcal{L}_2, \mathcal{L}_3)) = \text{Sub} \left( \text{Sub}(\mathcal{L}_1, \mathcal{L}_2), \mathcal{L}_3 \right)$ if $\mathcal{L}_2$ is symmetric.

---

(3) A mapping $h$ of $\Sigma_1^*$ into $\Sigma_2^*$ is called a homomorphism if $h(xy) = h(x)h(y)$ for all $x$ and $y$ in $\Sigma_1^*$.

(4) A homomorphism $h$ on $L$ is restricted on $L$ if $h(w) = \epsilon$ for $w \in L$ implies $w = \epsilon$ and there is a positive integer $q$ such that $h(w) \neq \epsilon$ for each subword $w$ of length $\geq q$ of each word in $L$.

(5) The ordering, of course, is understood to be by family inclusion.
Proof. (a) Given $L_3$ in $f_3$, let $\tau_2$ be a substitution of $L_3$ by languages of $f_2$ and $\tau_1$ a substitution of $\tau_2(L_3)$ by languages of $f_1$. Then

$$\tau_1(\tau_2(L_3)) = \tau'(L_3),$$

where $\tau'$ is the substitution such that $\tau'(a) = \tau_1(\tau_2(a))$ for each $a$. Since $\tau_2(a)$ is in $f_2$ and $\tau_1$ is a substitution by languages of $f_1$, $\tau_1(\tau_2(a))$ is in $\text{Sub}(f_1, f_2)$. Therefore $\tau'$ is a substitution by languages of $\text{Sub}(f_1, f_2)$, so that $\text{Sub}(f_1, \text{Sub}(f_2, f_3)) \subseteq \text{Sub}(\text{Sub}(f_1, f_2), f_3)$.

(b) Suppose $f_2$ is symmetric. Let $L_3$ be in $f_3$ and $\tau'$ a substitution of $L_3$ by languages of $\text{Sub}(f_1, f_2)$. Then for each $a$ in $E_L$, $\tau'(a) = \tau_1(\tau_2(a))$, where $L_2, a$ is in $f_2$ and $\tau_1, a$ is a substitution on $E_L$ by languages of $f_1$.

Since $f_2$ is symmetric and $E$ is infinite, we may assume that $E_{L_2, a} \cap E_{L_2, a'} = \emptyset$ for $a \neq a'$. Let $E_a = \bigcup E_{L_2, a}$. Then there exists a substitution $\tau_1$ of $E_a$ by languages of $f_1$ such that $\tau_1(b) = \tau_1(a)(b)$ for each $a$ in $E_a$ and $b$ in $E_{L_2, a}$.

Hence $\tau'(L_3) = \tau_1(\tau_2(L_3))$, where $\tau_2$ is the substitution of $L_3$ by languages of $f_2$ defined by $\tau_2(a) = L_2, a$ for each $a$ in $E_{L_3}$. Thus $\tau'$ is in $\text{Sub}(f_1, \text{Sub}(f_2, f_3))$. Therefore $\text{Sub}(\text{Sub}(f_1, f_2), f_3) \subseteq \text{Sub}(f_1, \text{Sub}(f_2, f_3))$, whence equality by (a).

Remark. The hypothesis in (b) cannot always be removed and the result be true. For example, let $\Sigma = a_2 / 1x2$, $f_1 = \{[a_1] / 1x1\}$, $f_2 = \{[a_2] / 1x1\}$, and $f_3 = \{[a_1, a_2] / 1, 1x1\}$. Then $\text{Sub}(f_2, f_3) = \{[a_2] / 1x1\}$, $\text{Sub}(f_3, f_2, f_3) = \{[a_1] / 1x1\}$, $\text{Sub}(f_1, f_3) = f_1$, and $\text{Sub}(\text{Sub}(f_1, f_2), f_3) = f_3$. Clearly $\text{Sub}(\text{Sub}(f_1, f_2), f_3)$ is not contained in $\text{Sub}(f_1, \text{Sub}(f_2, f_3))$.

Recently, families of languages with six additional properties have been introduced [3] because of their intimate connection with families of languages.

A substitution $\tau$ is a substitution by languages of $f_2$ if $\tau(a)$ is in $f_2$ for each $a$. \(\text{(6)}\)
of interest in computer science. These families, called "abstract families of languages" are currently under extensive investigation and the present paper may be viewed as an addition to their literature.

**Definition.** As abstract family of languages (abbreviated AFL) is a family of languages closed under union, concatenation, +, e-free homomorphism, inverse homomorphism, and intersection with regular sets.

If $h$ is a homomorphism from $\Sigma_1^*$ into $\Sigma_2^*$ then $h^{-1}$, the inverse homomorphism, is the mapping from $2^{\Sigma_1^*}$ into $2^{\Sigma_2^*}$ defined by $h^{-1}(A) = \{w/h(w) \in A\}$ for each $A \subseteq \Sigma_2^*$.

For our purposes it is convenient to consider a reformulation of the closure properties of an AFL, expressed as follows:

**Notation.** Let $R$ be the family of regular sets (over $\Sigma$) and $R_o$ the family of e-free regular sets.

**Proposition 1.2.** A family $L$ of languages is an AFL if and only if:
1. $R_o \subseteq L$,
2. $\text{Sub}(R_o, L) \subseteq L$,
3. $\text{Sub}(L, R_o) \subseteq L$,
4. $L \cap R \subseteq L$, and
5. $\text{Hom}_L(L) \subseteq L$.

**Proof.** It is known [3] that each AFL satisfies all five of the conditions. Thus consider the converse. Assume $L$ satisfies (1) - (5). Thus $L$ has all the closure properties of an AFL except possibly for closure under inverse homomorphism.

Suppose $L$ contains a language containing $e$. It follows from (4) that $L$ contains $(e)$ and, from (1) and the fact that $L$ is closed under union, that $L$ contains $R$.

**Notation.** For each set $A \subseteq \Sigma^*$, $A^* = \cup_{i \geq 1} A_i$.

**Notation.** A homomorphism $h$ is called e-free if $h(w) = e$ implies $w = e$.

If $h$ is a homomorphism from $\Sigma_1^*$ into $\Sigma_2^*$ then $h^{-1}$, the inverse homomorphism, is the mapping from $2^{\Sigma_1^*}$ into $2^{\Sigma_2^*}$ defined by $h^{-1}(A) = \{w/h(w) \in A\}$ for each $A \subseteq \Sigma_2^*$.

The family of regular sets is the smallest family of languages containing all the finite languages and closed under union, concatenation, and $*$, where $A^* = A^* \cup \{e\}$ for each $A \subseteq \Sigma^*$. 
It then follows from Theorem 4 of [7] that $\mathcal{L}$ is closed under inverse homomorphism. Suppose $\mathcal{L}$ is $e$-free. By an argument similar to the one given in the proof of Theorem 4 of [7], it follows that $\mathcal{L}$ is closed under inverse homomorphism. In either case, therefore, $\mathcal{L}$ is closed under inverse homomorphism and so is an AFL.

Remark. In the presence of conditions (2) - (5), condition (1) is equivalent to the condition that $\mathcal{L}$ is nontrivial. That is, the only family $\mathcal{L}$ satisfying (2) - (5) but not (1) is the family $\mathcal{L} = \{\{e\}, \emptyset\}$.

Notation. Given a family $\mathcal{L}$ let AFL($\mathcal{L}$) be the smallest AFL containing $\mathcal{L}$.

It is known [3] that AFL($\mathcal{L}$) exists for each family $\mathcal{L}$.

Corollary. Given a family $\mathcal{L}$, let $\mathcal{L}^{(0)} = \emptyset \cup \mathcal{L}$ and for each $n \geq 0$ let $\mathcal{L}^{(n+1)} = \text{Sub}(\emptyset, \mathcal{L}^{(n)})$ if $n \equiv 0 \pmod{4}$, $\mathcal{L}^{(n+1)} = \text{Sub}(\mathcal{L}^{(n)}, \emptyset)$ if $n \equiv 1 \pmod{4}$, $\mathcal{L}^{(n+1)} = \mathcal{L}^{(n)} \setminus \mathcal{R}$ if $n \equiv 2 \pmod{4}$, and $\mathcal{L}^{(n+1)} = \text{Hom}_\mathcal{L}(\mathcal{L}^{(n)})$ if $n \equiv 3 \pmod{4}$. Then

$\mathcal{L} = \mathcal{L}^{(0)} \subseteq \mathcal{L}^{(1)} \subseteq \cdots \subseteq \mathcal{L}^{(n)} \subseteq \cdots$ and $\text{AFL}(\mathcal{L}) = \bigcup_{n \geq 0} \mathcal{L}^{(n)}$.

Proof. Let $\mathcal{L}'$ be any AFL containing $\mathcal{L}$. Then $\mathcal{L}^{(0)} \subseteq \mathcal{L}'$ and a simple induction on $n$ shows that $\mathcal{L}^{(n)} \subseteq \mathcal{L}'$ for all $n \geq 0$. Therefore $\bigcup_{n \geq 0} \mathcal{L}^{(n)} \subseteq \mathcal{L}'$. To complete the proof it suffices to verify that $\bigcup_{n \geq 0} \mathcal{L}^{(n)}$ is an AFL.

Clearly $\bigcup_{n \geq 0} \mathcal{L}^{(n)}$ satisfies condition (1) of Proposition 1.2.

Since

$\text{Sub}(\emptyset, \bigcup_{n \geq 0} \mathcal{L}^{(n)}) = \bigcup_{n \geq 0} \text{Sub}(\emptyset, \mathcal{L}^{(n)}) = \bigcup_{n \geq 0} \mathcal{L}^{(n+1)} = \bigcup_{n \geq 0} \mathcal{L}^{(n)}$

$\mathcal{L}^{(n)}$ satisfies condition (2). A similar calculation shows that it satisfies conditions (4) and (5). Finally, observe that each substitution $\tau$ by languages of $\bigcup_{n \geq 0} \mathcal{L}^{(n)}$ involves only finitely many languages. Therefore there is some $m$ such that $\tau$ is a substitution by languages of $\mathcal{L}^{(m)}$. Hence
16 September 1968

Sub( U f(n), R_o ) ≤ U Sub(f(n), R_o) = U Sub(f(n), R_o)

= U f(n+1) = U f(n),

which gives condition (3). Therefore U f(n) is an AFL.

A particularly important type of AFL is one which is closed under arbitrary homomorphism. That is,

Definition. An AFL f is said to be full [3] if it is closed under arbitrary homomorphism.

The following result provides a useful reformulation of full AFL.

Proposition 1.3. A family f of languages is a full AFL if and only if

(1) R_o ⊆ f, (2) Sub(R, f) ≠ f, (3) Sub(f, R_o) = f, and (4) f ∩ R ⊆ f.

Proof. Suppose f is a full AFL. By Proposition 1.2, f satisfies (1), (3), and (4). By Theorem 2.4 of [3], f satisfies (2).

Now assume f satisfies the above conditions. Then f satisfies condition (1) - (4) of Proposition 1.2. Since each homomorphism is a substitution by languages of R, (2) above implies f is closed under homomorphism. Thus f satisfies (1) - (5) of Proposition 1.2 and so is an AFL. Being closed under arbitrary homomorphism, f is a full AFL.

Notation. For each AFL f, let AFL_f(f) be the smallest full AFL containing f.

Corollary. Given a family f, let f^(o) = f U R_o and for each n ≥ 0, let f^(n+1) = Sub(R, f^(n)) if n = 0 (mod 3), f^(n+1) = Sub(f^(n), R_o) if n = 1 (mod 3), and f^(n+1) = f^(n) ∩ R if n = 2 (mod 3). Then f = f^(o) ⊆ ... ⊆ f^(n) ⊆ ... and AFL_f(f) = U f^(n).

The proof is analogous to that of the corollary to Proposition 1.2 and is omitted.
Section 2. Substitution of AFL

In this section we show that the substitution of an AFL into an AFL is an AFL. To do this we first prove two technical lemmas. The first asserts a kind of distributivity of regular set intersection with respect to substitution and the second a kind of distributivity of Hom with respect to substitution. (These lemmas are also used in section 4.)

Notation. Let $X_{o}$ be the family of $e$-free finite sets.

Note that $X_{o}$ is symmetric and $\text{Sub}(X_{o}, X_{o}) = X_{o}$. 

Lemma 2.1. For all families of languages $I_{1}$ and $I_{2}$,

$$\text{Sub}(I_{1}, I_{2}) \cap R \subseteq \text{Sub}(I_{1} \cap R, \text{Sub}(X_{o}, I_{2}) \cap R).$$

Proof. Let $I_{2}' = \text{Sub}(X_{o}, I_{2})$. Then $I_{2} \subseteq I_{2}'$ and $\text{Sub}(I_{1}, I_{2}) \cap R \subseteq \text{Sub}(I_{1}, I_{2}') \cap R$.

By (b) of Proposition 1.1, $\text{Sub}(X_{o}, I_{2}') = \text{Sub}(X_{o}, \text{Sub}(X_{o}, I_{2})) = \text{Sub}(\text{Sub}(X_{o}, X_{o}), I_{2})$. Thus

$$\text{Sub}(I_{1} \cap R, \text{Sub}(X_{o}, I_{2}') \cap R) = \text{Sub}(I_{1} \cap R, \text{Sub}(X_{o}, I_{2}) \cap R).$$

Hence it suffices to show the lemma for $I_{2}$ replaced by $I_{2}'$. Thus, without loss of generality, we may assume that $\text{Sub}(X_{o}, I_{2}) = I_{2}$. Using this assumption it suffices to show

$$\text{Sub}(I_{1}, I_{2}) \cap R \subseteq \text{Sub}(I_{1} \cap R, I_{2} \cap R).$$

Since $I_{2}$ is closed under substitution by $X_{o}$, it follows that $I_{2}$ is symmetric. Let $L_{2}$ be in $I_{2}$ and let $\tau_{1}$ be a substitution of $L_{2}$ by languages in $I_{1}$. Let $R$ be a regular set, with $R \subseteq \Sigma_{1}^{*}$. By extending $\Sigma_{1}$ if necessary, we may assume that $\tau_{1}(L_{2}) \subseteq \Sigma_{1}^{*}$. Since $I_{2}$ is symmetric, we may assume that $L_{2} \subseteq \Sigma_{2}^{*}$ with $\Sigma_{1} \cap \Sigma_{2} = \emptyset$. Let $\tau_{2}$ be the substitution on $L_{2}^{*}$ defined by $\tau_{2}(a) = \tau_{1}(a)$ for each $a$ in $\Sigma_{2}$. Let $h$ be the homomorphism on $(\Sigma_{1} \cup \Sigma_{2})^{*}$ defined by $h(a) = e$ for each $a$
in \( \Sigma_2 \) and \( h(b) = b \) for each \( b \) in \( \Sigma_1 \). Then \( \tau_1 = h \tau_2 \) and

\[
\tau_1(L_2) \cap R = h \tau_2(L_2) \cap R = h(\tau_2(L_2) \cap h^{-1}(R)).
\]

Since \( R \) is regular, \( h^{-1}(R) \) is regular \([5]\). Thus there exists an fsa \([11]\)

\[
A = (K, \Sigma_1 \cup \Sigma_2, \delta', P_0, F) \text{ such that } (12) \quad \tau(A) = h^{-1}(R).
\]

Let

\[
R' = (\tau(A) \cap \{e\}) \cup (a_1, P_0, F_1) \ldots (a_m, P_{m-1}, P_m) / \forall 1, \text{ each } a_i \text{ in } \Sigma_2,
\]

\[
\text{each } p_i \text{ in } K, \text{ and } p_m \text{ in } F).
\]

As is well known \([2]\), \( R' \) is regular.

For each \( (a, p, q) \) in \( \Sigma_2 \times K \times K \), let

\[
R(a, p, q) = \{ w \text{ in } \Sigma_1^* / \delta(p, aw) = q \}.
\]

Then \( R(a, p, q) \) is regular (since \( R(a, p, q) = T(B) \), where \( B \) is the fsa \((K, \Sigma_1, \delta, \delta(p, a), \{q\})\)). Let \( \tau_3 \) be the substitution defined on \((\Sigma_2 \times K \times K)^*\)

by \( \tau_3((a, p, q)) = R(a, p, q) \) for each \( (a, p, q) \) in \( \Sigma_2 \times K \times K \). Then \( \tau_3(R') \) is the subset of \( h^{-1}(R) \) of all words that do not begin with a symbol of \( \Sigma_1 \). Since \( \tau_2(L_2) \) contains no word beginning with a symbol of \( \Sigma_1 \),

\[
\tau_2(L_2) \cap h^{-1}(R) = \tau_2(L_2) \cap \tau_3(R').
\]

Let \( \tau' \) be the substitution on \( \Sigma_2^* \) defined by \( \tau'(a) = (a) \times K \times K \) for each \( a \) in \( \Sigma_2 \) and \( \tau'' \) the substitution on \((\Sigma_2 \times K \times K)^*\) by \( \tau''((a, p, q)) = \tau_2(a) \) for each \( (a, p, q) \). Then \( \tau_2 = \tau'' \tau' \), so that

\[
\tau_2(L_2) \cap h^{-1}(R) = \tau'' \tau_2(L_2) \cap \tau_3(R').
\]

\([11]\) As fsa (finite state acceptor) is a 5-tuple \( A = (K, \Sigma_1, \delta, P_0, F) \), where (i) \( K \) and \( \Sigma_1 \) are finite sets (of states and inputs, resp.), (ii) \( \delta \) is a function from \( K \times \Sigma_1 \) to \( K \) (the next state function), (iii) \( s_0 \) is an element of \( K \) (the start state), and (iv) \( F \subseteq K \) (the set of accepting states). The function \( \delta \) is extended inductively to \( K \times \Sigma_1 \) by letting \( \delta(p, \varepsilon) = p \) and \( \delta(p, a_1 \ldots a_n) = \delta(\delta(p, a_1 \ldots a_{n-1}), a_n) \) for each \( p \) in \( K \), each \( n \geq 1 \), and each \( a_1, \ldots, a_n \) in \( \Sigma_1 \).

\([12]\) For each fsa \( A \), \( \tau(A) = \{ w \text{ in } \Sigma_1^*/(p, w) \text{ in } F \} \). It is known \([8]\) that a

set \( R \subseteq \Sigma_1^* \) is regular if and only if \( \tau(A) = R \) for some fsa \( A \).
Let $\tau_3'$ be the substitution on $(\mathbb{E}_2 \times K \times K)^*$ defined by

$$
\tau_3'((a,p,q)) = \tau''((a,p,q)) \cap \tau_3((a,p,q))
$$

= $\sigma_1(a) \cap aR(a,p,q)$

for each $(a,p,q)$.

We now show that

(*) $\tau''(L_2) \cap \tau_3(R') = \tau_3'[\tau'(L_2) \cap R']$.

Since $\tau_3''((a,p,q)) = \tau''((a,p,q)) \cap \tau_3((a,p,q))$, we have $\tau_3''(R') \subseteq \tau_3(R')$ and $\tau_3'[\tau'(L_2) \subseteq \tau''(L_2) \cap \tau_3(R')]$. Thus

$$
\tau_3'[\tau'(L_2) \cap R'] \subseteq \tau_3''(L_2) \cap \tau_3''(R')
$$

$\subseteq \tau''(L_2) \cap \tau_3(R')$.

To see the reverse containment, assume $w$ is in $\tau''(L_2) \cap \tau_3(R')$.

Suppose $w = \varepsilon$. Then $\varepsilon$ is in $L_2$ and in $R'$, since $\tau''$ and $\tau_3$ are $\varepsilon$-free substitutions. (13) Therefore $\varepsilon$ is in $\tau'''(L_2) \cap R')$. Suppose $w \neq \varepsilon$. Then

$w = a_1x_1 \ldots a_q x_q$ for some $q \in \mathbb{N}$, some $a_1, \ldots, a_q$ in $L_2$, and some

$x_1, \ldots, x_q$ in $K$. Since $w$ is in $\tau''(L_2) = \tau_2(L_2)$, it follows that $a_1 \ldots a_q$ is in $L_2$ and $x_i$ is in $\sigma_1(a_i)$ for each $i$. Define $p_i$ in $K$ by induction on $i$ so that $p_{j+1} = \delta(p_j, a_{j+1}x_{j+1})$ for each $j$, $0 \leq j < q$. Let $w_2 = (a_1, p_2, p_1, \ldots, a_q, p_{q-1}, p_q)$. Since $w$ is in $\tau_3(R')$, $w_2$ is in $R'$. Since $\tau'(a) = (a) \times K \times K$ for each $a$, $w_2$ is in $\tau'(a_1, \ldots, a_q)$. Therefore $w_2$ is in $\tau'(L_2) \cap R'$. By definition of $\tau_3'$, $a_1 x_1$ is in $\tau_3'(a_1, p_1, p_1)$ for $1 \leq i \leq q$. Therefore, $w$ is in $\tau_3(w_2)$ so that $w$ is in $\tau_3''(L_2) \cap R')$. Hence $\tau''(L_2) \cap \tau_3(R') \subseteq \tau''(L_2) \cap \tau_3(R')$, implying (*).

(13) A substitution $\tau$ is $\varepsilon$-free if $\varepsilon$ in $\tau(a)$ implies $a = \varepsilon$. 
From (*), it follows that
\[ \tau_1(L_2) \cap R = h[\tau_2(L_2) \cap h^{-1}(R)] \]
\[ = h[\tau'(L_2) \cap \tau_3(R')] \]
\[ = h\tau_3'(L_2) \cap R'. \]

Now \( \tau'(L_2) \) is in \( \text{Sub}(\mathcal{F}_0, L_2) = L_2 \) and \( \tau'(L_2) \cap R' \) is in \( L_2 \cap R. \) The composite
\[ h \tau_3' \]
is a substitution such that
\[ h \tau_3'((a,p,q)) = h[a \tau_1(a) \cap aR(a,p,q)] = \tau_1(a) \cap R(a,p,q) \]
for each \( (a,p,q) \). Since \( \tau_1(a) \) is in \( L_1 \) and \( R(a,p,q) \) in \( R, h \tau_3' \)
is a substitution by sets of \( L_1 \cap R. \) Therefore \( \tau_1(L_2) \cap R \) is in
\[ \text{Sub}(L_1 \cap R, L_2 \cap R), \]
and the lemma is proved.

The result pertaining to the distributivity of \( \text{Hom}_r \) is

**Lemma 2.2.** For all families of languages \( L_1 \) and \( L_2 \),
\[ \text{Hom}_r[\text{Sub}(L_1, L_2)] \subseteq \text{Sub}[	ext{Hom}_r(L_1 \cap R), \text{Hom}_r(\text{Sub}(\mathcal{F}_0, L_2))]. \]

**Proof.** Let \( L_2 \) be a language in \( L_2 \), \( \tau \) a substitution of \( L_2 \) by languages of
\( L_1 \), \( \Sigma_1 = \Sigma_1(L_2), \Sigma_2 = \Sigma_2 \), and \( h \) a homomorphism from \( \Sigma_1 \) to \( \Sigma_2 \) which is
restricted on \( \tau(L_2). \) By definition of \( \Sigma_2 \), for each \( a \) in \( \Sigma_2 \), there is a word \( w \) in
\( L_2 \) containing an occurrence of \( a. \) Let \( R \) be the set containing \( \varepsilon \) and all words \( w \)
in \( \Sigma_1 \) such that \( h(w) \neq \varepsilon. \) Then \( R = (\varepsilon) \cup h^{-1}(\Sigma_2) \) is a regular set. Since, for
each \( a \) in \( \Sigma_2 \), each subword of a word of \( \tau(a) \) is also a subword of some word of
\( \tau(L_2), \) \( h \) is restricted on \( \tau(a) \cap R. \) Let \( \tau' \) be the substitution by languages
of \( \text{Hom}_r(L_1 \cap R) \) defined by \( \tau'(a) = h[\tau(a) \cap R] \) for each \( a \) in \( \Sigma_2 \).

For each \( a \) in \( \Sigma_2 \) let \( a' \) be a new symbol and let \( \Sigma_2' = (a'/a \text{ in } \Sigma_2). \) Let \( \tau' \)
be the substitution on \( \Sigma_2' \) defined by \( \tau'(a) = (a) \) if \( \tau(a) \) contains \( \varepsilon \) or \( h\tau(a) \)
is \( \varepsilon \)-free, and \( \tau'(a) = (a,a') \) if \( \tau(a) \) is \( \varepsilon \)-free and \( h\tau(a) \) contains \( \varepsilon. \) Then \( \tau' \)
is a substitution by languages of \( \mathcal{F}_0. \) Let \( h' \) be the homomorphism on \( (E_2 \cup E_2') \) defined
\[ h'(a) = a \text{ and } h'(a') = e \text{ for each } a \text{ in } \Sigma_2. \] Then \( h' \) is restricted on \( \tau'(L_2) \).

[For let \( \Sigma_q = \{a/\tau'(a) = a\} \) and \( \Sigma'_q = \{a/\tau'(a) = (a, a')\} \). By definition of \( \tau' \), for each \( a \) in \( \Sigma_q \) there is a non-\( e \) word \( w_a \) in \( \tau(a) \) such that \( h(w_a) = e \). Since \( h \) is restricted on \( \tau(L_2) \) it follows that (a) \( L_2 \cap \Sigma'_q = L_2 \cap \{e\} \) and (b) there exists \( q \geq 0 \) such that any subword of length \( q \) of a word of \( L_2 \) contains an occurrence of an element of \( \Sigma_q \).]

Now (a) implies that if \( h'(w') = e \) for \( w' \) in \( \tau'(L_2) \) then \( w' = e \), and (b) implies that if \( h'(u') = e \) for a subword \( u' \) of some word of \( \tau'(L_2) \) then \( |u'| \leq q \). Therefore \( h' \) is restricted on \( \tau'(L_2) \).

Therefore \( h'(L_2) \) is a language in \( \text{Hom}_r(\text{Sub}(\Sigma'_2, L_2)) \), so that \( \tau^* h'^*(L_2) \) is in \( \text{Sub}[\text{Hom}_r(L_1 \times R), \text{Hom}_r(\text{Sub}(\Sigma'_2, L_2))] \).

To complete the proof it suffices to show that \( \text{hr}(L_2) = \tau^* h'^*(L_2) \).

Therefore let \( w \) be a word in \( L_2 \). We shall show that \( \text{hr}(w) = \tau^* h'^*(w) \). If \( w = e \), then \( \text{hr}(w) = e = \tau^* h'^*(w) \). Assume \( w = a_1 \ldots a_n, n \geq 1 \), each \( a_i \) in \( \Sigma_2 \).

By definition,

\[ \text{hr}(a_1 \ldots a_n) = (h(w_1) \ldots h(w_n))/w_1 \in \tau(a_i)). \]

For each \( i \), let \( w_i \) be a word of \( \tau(a_i) \). Let \( J \) be the set of all \( j \) such that \( h(w_j) = e \) and \( \tau(a_j) \) is \( e \)-free, and let \( J' = \{1, \ldots, n\} - J \). Then \( h(w_j) \) is in \( \tau^*(a_j) \) for each \( j \) in \( J' \). For each \( i, 1 \leq i \leq n \), let \( b_i = a_i \) if \( i \) is in \( J \) and \( b_i = a_i \) if \( i \) is in \( J' \). Since \( h(w_j) = e \) and \( \tau(a_j) \) is \( e \)-free for each \( j \) in \( J \), \( b_1 \ldots b_n \) is in \( \tau^*(a_1 \ldots a_n) \). If \( i \) is in \( J \) then \( h'(b_i) = e = h(w_i) \) and \( h(w_i) \) is in \( \tau^* h'(b_i) \). If \( i \) is in \( J' \), then \( h'(b_i) = a_i \) and \( h(w_i) \) is in \( \tau^*(a_i) \). Thus \( h(w_1) \ldots h(w_n) \) is in \( \tau^*(h'(b_1 \ldots b_n)) \), so that \( \text{hr}(w) \subseteq \tau^* h'^*(w) \).

To see the reverse containment, note that

\[ \tau^* h'^*(a_1 \ldots a_n) = (u_1 \ldots u_n/u_1 \in \tau^*(h'(b_i)), b_i \in \tau^*(a_i)). \]
For each $i$, if $b_i = a_i$ then $h'(b_i) = e$, $u_i = e$, $\rho(a_i)$ is $e$-free, and $\rho(a_i)$ contains $e = u_i$; and if $b_i = a_i$, then $u_i$ is in $\tau''(a_i) = h[\tau(a_i) \cap R] \subseteq \rho(a_i)$.

In either case, $u_i$ is in $\rho(a_i)$. Thus $u_1 \ldots u_n$ is in $\rho(a_1 \ldots a_n)$ and $\tau''h''(w) \subseteq \rho(w)$.

We are now ready for the main result of the section.

**Theorem 2.1.** If $f_1$ and $f_2$ are AFL, then so is $\text{Sub}(f_1, f_2)$.

**Proof.** Since $\rho_o \subseteq f_1$ and $\rho_o \subseteq f_2$, it follows that $\rho_o = \text{Sub}(\rho_o, \rho_o) \subseteq \text{Sub}(f_1, f_2)$. Thus $\text{Sub}(f_1, f_2)$ satisfies (1) of Proposition 1.2. By Propositions 1.1 and 1.2,

$$\text{Sub}(\rho_o, \text{Sub}(f_1, f_2)) = \text{Sub}(\text{Sub}(\rho_o, f_1), f_2) \subseteq \text{Sub}(f_1, f_2)$$

and

$$\text{Sub}(\text{Sub}(f_1, f_2), \rho_o) = \text{Sub}(f_1, \text{Sub}(f_2, \rho_o)) \subseteq \text{Sub}(f_1, f_2).$$

Thus $\text{Sub}(f_1, f_2)$ satisfies conditions (2) and (3) of Proposition 1.2. By Lemma 2.1,

$$\text{Sub}(f_1, f_2) \cap R \subseteq \text{Sub}(f_1 \cap R, \text{Sub}(f_0, f_2) \cap R) \subseteq \text{Sub}(f_1, f_2).$$

By Lemma 2.2,

$$\text{Hom}_r[\text{Sub}(f_1, f_2)] \subseteq \text{Sub}[\text{Hom}_r(f_1 \cap R), \text{Hom}_r(\text{Sub}(f_0, f_2))] \subseteq \text{Sub}(f_1, f_2).$$

Thus $\text{Sub}(f_1, f_2)$ satisfies conditions (4) and (5) of Proposition 1.2.

Hence $\text{Sub}(f_1, f_2)$ is an AFL.

**Corollary 1.** If $f_1$ is a full AFL and $f_2$ is an AFL, then $\text{Sub}(f_1, f_2)$ is a full AFL.

**Proof.** By Theorem 2.1, $\text{Sub}(f_1, f_2)$ is an AFL. By Propositions 1.1 and 1.3, we have

$$\text{Sub}(R, \text{Sub}(f_1, f_2)) = \text{Sub}(\text{Sub}(R, f_1), f_2) \subseteq \text{Sub}(f_1, f_2).$$

Thus $\text{Sub}(f_1, f_2)$ satisfies (1) - (4) of Proposition 1.3 and so is a full AFL.
Corollary 2. If \( \mathcal{L} \) is an AFL, then \( \text{AFL}_x(\mathcal{L}) = \text{Sub}(\mathcal{R}, \mathcal{L}) \).

**Proof.** By Corollary 1, \( \text{Sub}(\mathcal{R}, \mathcal{L}) \) is a full AFL. If \( \mathcal{L}' \) is any full AFL containing \( \mathcal{L} \), then \( \text{Sub}(\mathcal{R}, \mathcal{L}) \subseteq \text{Sub}(\mathcal{R}, \mathcal{L}') \subseteq \mathcal{L}' \) by (2) of Proposition 1.3. Thus \( \text{Sub}(\mathcal{R}, \mathcal{L}) \) is the smallest full AFL containing \( \mathcal{L} \), i.e., \( \text{Sub}(\mathcal{R}, \mathcal{L}) = \text{AFL}_x(\mathcal{L}) \).

**Remarks** (1) Theorem 2.1 and Corollary 1 both hold if \( \mathcal{L}_2 \) is a family of languages such that \( \text{Sub}(\mathcal{R}, \mathcal{L}_2) \) is an AFL. For in this case

\[
\text{Sub}(\mathcal{L}_1, \mathcal{L}_2) \subseteq \text{Sub}(\mathcal{L}_1, \text{Sub}(\mathcal{R}, \mathcal{L}_2)) = \text{Sub}(\text{Sub}(\mathcal{L}_1, \mathcal{R}), \mathcal{L}_2) \subseteq \text{Sub}(\mathcal{L}_1, \mathcal{L}_2).
\]

Thus \( \text{Sub}(\mathcal{L}_1, \mathcal{L}_2) = \text{Sub}(\mathcal{L}_1, \text{Sub}(\mathcal{R}, \mathcal{L}_2)) \) is a (full) AFL if \( \mathcal{L}_1 \) is a (full) AFL.

In particular, the results are valid if \( \mathcal{L}_2 \) is a pre-AFL or e-free pre-AFL [4].

Similarly, Corollary 2 is valid if \( \mathcal{L} \) is any family of languages such that \( \text{Sub}(\mathcal{R}, \mathcal{L}) \) is an AFL, hence if \( \mathcal{L} \) is a pre-AFL or e-free pre-AFL.

(2) Theorem 2.1 suggests the following general problem (which is not studied here): "Identify" \( \text{Sub}(\mathcal{L}_1, \mathcal{L}_2) \) for well-known AFL, \( \mathcal{L}_1 \) and \( \mathcal{L}_2 \).

Section 3. AFL of Substitutions

This section is concerned with relations between substitution and the AFL generated by a family. The main result asserts that, with suitable hypotheses,

\[
\text{AFL}[\text{Sub}(\mathcal{L}_1, \mathcal{L}_2)] = \text{Sub}([\text{AFL}(\mathcal{L}_1), \text{AFL}(\mathcal{L}_2)]).
\]

In order to prove the main result, a sequence of lemmas is needed.

**Lemma 3.1.** Let \( \mathcal{L}_1 \) be an e-free (or arbitrary) family of languages. Then for every family of languages \( \mathcal{L}_2 \),

\[
\text{Sub}(\mathcal{L}_1 \wedge \mathcal{R}, \mathcal{L}_2 \wedge \mathcal{R}) \subseteq \text{Hom}_2[\text{Sub}(\mathcal{L}_1 \cup \mathcal{R}, \mathcal{L}_2), \mathcal{R}]
\]

(or \( \text{Sub}(\mathcal{L}_1 \wedge \mathcal{R}, \mathcal{L}_2 \wedge \mathcal{R}) \subseteq \text{Hom}[\text{Sub}(\mathcal{L}_1 \cup \mathcal{R}, \mathcal{L}_2), \mathcal{R}] \)).
Proof. We prove the lemma for the \( \varepsilon \)-free case, the argument for arbitrary \( \Sigma_1 \) differing trivially.

Let \( L_2 \) be in \( L_2 \), \( R_2 \) in \( R \), and consider \( L_2 \cap R_2 \) with \( L_2 \cup R_2 \subseteq \Sigma_2^* \). Let \( \tau \) be a substitution on \( \Sigma_2^* \) by languages of \( \Sigma_1 \cap R \). Then for each \( a \) in \( \Sigma_2 \), \( \tau(a) = L_a \cap R_a \), where \( L_a \) is in \( L_1 \) and \( R_a \) is in \( R \). For each \( a \) in \( \Sigma_2 \) let \( a' \) be a new symbol and let \( \Sigma'_2 = \{a'/a \text{ in } \Sigma_2 \} \). Let \( \tau' \) be the substitution on \( \Sigma'_2 \) defined by \( \tau'(a) = (a'a) \) for each \( a \). Then \( \tau' \) is a substitution by languages of \( \Sigma_2 \), so that \( \tau'(L_2) \) is in \( \text{Sub}(\Sigma_2, \Sigma'_2) \). Let \( \tau'' \) be the substitution on \( (\Sigma_2 \cup \Sigma'_2)^* \) defined by \( \tau''(a') = \{a'\} \) and \( \tau''(a) = L_a \) for each \( a \) in \( \Sigma_2 \). Then \( \tau'' \) is a substitution by languages of \( \Sigma_2 \), so that \( \tau''(L_2) \) is in \( \text{Sub}(\Sigma_2 \cup \Sigma'_2, \Sigma_2 \cup \Sigma'_2) \). Let \( R' = \bigcup \{a'R_a\} \) and let \( R'' = \tau''(R_2) \), where \( \tau'' \) is the substitution on \( \Sigma_2^* \) defined by \( \tau''(a) = a' \) for each \( a \). Then \( R' \) and \( R'' \) are regular sets, and \( \tau''(L_2) \cap R' \cap R'' \) is in \( \text{Sub}(\Sigma_2 \cup \Sigma'_2, \Sigma_2 \cup \Sigma'_2) \). Let \( h \) be the homomorphism such that \( h(a') = \varepsilon \) for each \( a' \) and \( h(b) = b \) if \( b \) is a symbol not in \( \Sigma'_2 \). Since \( L_1 \) is \( \varepsilon \)-free, \( h \) cannot erase two consecutive symbols of a word in \( \tau''(L_2) \). Furthermore, if \( w \) is in \( \tau''(L_2) \) and \( h(w) = \varepsilon \), then \( w = \varepsilon \). Therefore \( h \) is restricted on \( \tau''(L_2) \) and thus restricted on \( \tau''(L_2) \cap R' \cap R'' \).

To complete the proof it suffices to show that \( \tau(L_2 \cap R_2) = h[\tau''(L_2) \cap R' \cap R''] \). Let \( w \) be any word in \( \Sigma_2^* \). If \( w = \varepsilon \), then \( \tau(\varepsilon) = (\varepsilon) = h(\tau''(\varepsilon)) \). Suppose \( w = a_1 \ldots a_n \), \( n \geq 1 \), each \( a_i \) in \( \Sigma_2 \). Then

\[
\tau(w) = (v_1 \ldots v_n/v_i \text{ in } L_{a_i} \cap R_{a_i}) = h(a_1 v_1 \ldots a_n v_n/v_i \text{ in } L_{a_i} \cap R_{a_i})
\]

(14) Regular sets are closed under intersection [8] and under substitution by regular sets [1].
Therefore $\tau(w) = h[\tau''\tau'(w) \cap R']$ for every word $w$ in $\Sigma^*$. Hence

$$\tau(L_2 \cap R_2) = h[\tau''\tau'(L_2 \cap R_2) \cap R'].$$  

Clearly $\tau''\tau'(R_2) \subseteq R'$. Hence $\tau''\tau'(L_2 \cap R_2) \subseteq \tau''\tau'(L_2 \cap R')$. We prove the reverse inclusion, thereby obtaining equality.

Suppose $w$ is in $\tau''\tau'(L_2) \cap R'$. Assume $w = e$. Then $e$ is in $\tau''\tau'(L_2)$ and in $R'$, thus in $L_2$ and in $R_2$. Hence $e$ is in $L_2 \cap R_2$ and therefore in $\tau''\tau'(L_2 \cap R_2)$. Assume $w = a_1^{v_1} \ldots a_n^{v_n}$, $\forall k \geq 1$, is in $\tau''\tau'(L_2) \cap R'$. Then each $v_i$ is in $L_{a_i}$, and $a_1 \ldots a_n$ is in $L_2$ and in $R_2$. Therefore $w$ is in $\tau''\tau'(L_2 \cap R_2)$. Thus $\tau''\tau'(L_2 \cap R') \subseteq \tau''\tau'(L_2 \cap R_2)$.

Since $\tau''\tau'(L_2 \cap R_2) = \tau''\tau'(L_2) \cap R'$, we have

$$\tau(L_2 \cap R_2) = h[\tau''\tau'(L_2 \cap R_2) \cap R']$$  

$$= h[\tau''\tau'(L_2) \cap R' \cap R'].$$

**Lemma 3.2.** Let $L_2$ be a family of languages. Then

(a) for each $e$-free symmetric family $L_1$,

$$\text{Sub}[\text{Hom}_{X}(L_1), L_2] \subseteq \text{Hom}_{X}[\text{Sub}(L_1, L_2)].$$

(b) for each $e$-free family $L_1$,

$$\text{Sub}_{X}[L_1, \text{Hom}_{X}(L_2)] \subseteq \text{Hom}_{X}[\text{Sub}_{X}(L_1 U \Phi_0, \text{Sub}(\Phi_0, L_2)).$$

**Proof.** (a) Let $L_2$ be in $L_2$ and $\tau$ a substitution of $L_2^*$ by languages of $\text{Hom}_X(L_1)$. Then for each $a$ in $L_2$, $\tau(a) = h_a(L_a)$, where $L_a$ is in $L_1$ and $h_a$ is restricted on $L_a$. Since $L_1$ is symmetric, we may assume that $L_a \cap L_b = \emptyset$ for $a \neq b$. Then there is a homomorphism $h$ on $(L_2^*)_a$ such that $h(L_a) = h_a(x)$, where $x$ is in $L_a^*$. Hence $\tau$ is the composite $hr'$, where $r'$ is the substitution on $L_2^*$ defined by $r'(a) = L_a$ for each $a$. Since $r'$ is a substitution by languages of $L_1$, it
suffices to show that \( h \) is restricted on \( \tau'(L_2) \).

Since \( h_a \) is restricted on \( L_a \) for each \( a \) there exists \( q_a > 0 \) such that

\[ |w'| \leq q_a \] whenever \( w' \) is a subword of a word of \( L_a \) and \( h_a(w') = \epsilon \). Furthermore, since \( L_a \) is \( \epsilon \)-free, \( h_a(w) \neq \epsilon \) for each \( w \) in \( L_a \). Let \( q = \max(q_a/a) \) and consider \( h \) on \( \tau'(L_2) \). If \( w' \) is a subword of a word of \( \tau'(L_2) \), then \( w' \) is a subword of a word of the form \( w_1 \ldots w_n \) with \( w_i \) in \( L_{a_i} \) for each \( i \). Also, \( w' = w_1 w_{i+1} \ldots w_{j-1} w_j \), where \( i < j \), \( w_i' \) is a subword of \( w_i \), and \( w_j' \) is a subword of \( w_j \). Thus \( |w_i'| \leq q \) and \( |w_j'| \leq q \). Suppose \( h(w) = \epsilon \). Then \( w_1 \ldots w_{j-1} w_j = \epsilon \), so that \( |w'| \leq 2q \). If \( w' \) is a word of \( \tau'(L_2) \), that is, \( w' = w_1 \ldots w_n \) each \( w_i \) in \( L_{a_i} \), and if \( h(w) = \epsilon \); then \( w_i = \epsilon \) for each \( i \), so that \( w = \epsilon \). Therefore \( h \) is restricted on \( \tau'(L_2) \) and the proof of (a) is complete.

(b) Let \( L_2 \) be in \( \Sigma_2' \), \( h \) a homomorphism restricted on \( L_2 \) and \( \tau \) a substitution on \( \Sigma^*_h(L_2) \) by languages of \( \Sigma_1 \). Let \( c \) be a symbol not occurring in \( \Sigma^*_h(L_2) \) and let \( \tau' \) be the substitution on \( \Sigma^*_h(L_2) \) defined by \( \tau'(a) = (c \, a) \) for each \( a \) in \( \Sigma_2 \). Clearly \( \tau'(L_2) \) is in \( \text{Sub}(\Sigma_0, \Sigma_2) \). Let \( \tau'' \) be the substitution on \( ((c \cup \Sigma^*_h(L_2))^* \) defined by \( \tau''(c) = \{ c \} \) and \( \tau''(b) = \tau(b) \) for \( b \) in \( \Sigma_h(L) \). Then \( \tau''[\tau'(L_2)] \) is in \( \text{Sub}(\Sigma_1 \cup \Sigma_2, \text{Sub}(\Sigma_0, \Sigma_2)) \). Let \( h' \) be the homomorphism on \( \Sigma^*_h(L_2) \) defined by \( h'(c) = \epsilon \) and \( h'(b) = b \) for \( b \) in \( \Sigma^*_h(L_2) \) \( \{ c \} \). Obviously \( \tau h(w) = h'' \tau'(w) \) for each \( w \) in \( \Sigma^*_h(L_2) \).

To complete the proof it suffices to show that \( h' \) is restricted on \( \tau'' \tau'(L_2) \).

Suppose there exists \( w \) in \( L_2 \) and \( w' \) in \( \tau'' \tau'(w) \) such that \( h'(w') = \epsilon \). Then \( \epsilon \) is in \( \tau h(w) \). Since \( \tau \) is a substitution by \( \epsilon \)-free sets, \( h(w) = \epsilon \). Since \( h \) is restricted on \( L_2 \), \( w = \epsilon \). Hence \( \epsilon = \epsilon \). Suppose there exists \( w \neq \epsilon \) in \( L_2 \) and a subword \( w' \) of some word in \( \tau'' \tau'(w) \) such that \( h'(w') = \epsilon \). Then \( w' = c^k \) for some \( k > 0 \). Let \( w = a_1 \ldots a_n \), each \( a_i \) in \( \Sigma_2 \). Then there exist positive
integers $i$ and $j$, with $1 < j$, and words $u_1, u_{i+1}, \ldots, u_{j-1}, u_j$ such that $u_i$ and $u_j$ are subwords of words in $\mathcal{C} \mathcal{R} \mathcal{H}(a_i)$ and $\mathcal{C} \mathcal{R} \mathcal{H}(a_j)$ respectively, $u_m$ is a word in $\mathcal{C} \mathcal{R} \mathcal{H}(a_m)$ for each $m$, $1 < m < j$, and $w' = u_1 u_{i+1} \ldots u_{j-1} u_j$. Thus $k \leq j-1 + 2$.

Since $f_1$ is $\epsilon$-free, $h(a_m) = \epsilon$ for each $m$, $1 < m < j$. Since $h$ is restricted on $L_2'$ there exists $q > 0$ such that $|v| < q$ if $h(v) = \epsilon$ and $v$ is a subword of a word in $L_2$. Hence $|j-1| < q + 2$, so that $|w'| = k < q+4$. Therefore $h'$ is restricted on $\tau''(L_2)$.

Remark. The method of proof of Lemma 3.2 shows that both parts (a) and (b) hold if the $\epsilon$-free condition on $f_1$ is dropped and $\text{Hom}_x$ is replaced throughout by $\text{Hom}$.

Lemma 3.3. Let $\mathcal{L}_2$ be a family of languages such that $\text{Sub}(\mathcal{R}_0, \mathcal{L}_2) \subseteq \mathcal{L}_2$ and let $\mathcal{L}_3$ be a (full) AFL. Given an $\epsilon$-free (arbitrary) symmetric family $\mathcal{L}$ containing $\mathcal{R}_0$ such that $\text{Sub}(\mathcal{L}, \mathcal{L}_2) \subseteq \mathcal{L}_3$, then $\text{Sub}(\text{AFL}(\mathcal{L}), \mathcal{L}_2) \subseteq \mathcal{L}_3$ ($\text{Sub}(\text{AFL}_{\alpha}(\mathcal{L}), \mathcal{L}_2) \subseteq \mathcal{L}_3$).

Proof. Let $\Lambda$ be the collection of all $\epsilon$-free (or arbitrary) symmetric families $\mathcal{L}'$ containing $\mathcal{R}_0$ and such that $\text{Sub}(\mathcal{L}', \mathcal{L}_2) \subseteq \mathcal{L}_3$. Let $(\mathcal{L}'_n)_{n \geq 0}$ be defined with respect to $\mathcal{L}$ as in the corollary to Proposition 1.2 (the corollary to Proposition 1.3). Then $\mathcal{L}_1 = \mathcal{L}$.

We shall show that $\mathcal{L}'$ in $\Lambda$ implies $\text{Sub}(\mathcal{R}_0, \mathcal{L}')$ (or $\text{Sub}(\mathcal{R}, \mathcal{L}')$), $\text{Sub}(\mathcal{L}', \mathcal{R}_0)$, $\mathcal{L}' \land \mathcal{R}$, and $\text{Hom}_x(\mathcal{L}')$ are all in $\Lambda$. By induction on $n$ it will then follow that each $\mathcal{L}'_n$ is in $\Lambda$. Clearly each of the above sets is symmetric. Also,

$\text{Hom}_x(\mathcal{L}')$ is omitted if $\mathcal{L}$ is not $\epsilon$-free.
Sub(\text{Sub}(r_0, z'), z_2) = \text{Sub}(r_0, \text{Sub}(z', z_2)) \subseteq \text{Sub}(r_0, z_3) \subseteq z_3

(\text{or} \quad \text{Sub(\text{Sub}(r, z'), z_2) = Sub(r, \text{Sub}(z', z_2)) \subseteq Sub(r, z_3) \subseteq z_3})

and

\text{Sub}(\text{Sub}(z', r_0), z_2) = \text{Sub}(z', \text{Sub}(r_0, z_2)) \subseteq \text{Sub}(z', z_2) \subseteq z_3.

Clearly Sub(z' \wedge r, z_2) \subseteq \text{Sub}(z' \wedge r, z_2 \wedge r). Since r_0 \subseteq z', r_0 \subseteq z'. Thus, by Lemma 3.1,

\text{Sub}(z' \wedge r, z_2) \subseteq \text{Hom}_r[\text{Sub}(z', \text{Sub}(z_0, z_2)) \wedge r]

\subseteq \text{Hom}_r[\text{Sub}(z', z_2) \wedge r]

\subseteq \text{Hom}_r[z_3 \wedge r]

\subseteq z_3

(\text{or} \quad \text{Sub}(z' \wedge r, z_2) \subseteq \text{Hom}(z_3 \wedge r) \subseteq z_3).

By Lemma 3.2 (a),

\text{Sub}[\text{Hom}_r(z'), z_2] \subseteq \text{Hom}_r[\text{Sub}(z', z_2)]

\subseteq \text{Hom}_r(z_3)

\subseteq z_3.

It now follows that the sequence of families \{I(n)\}_{n \geq 0} defined for I as in the corollary to Proposition 1.2 (or the corollary to Proposition 1.3) is in \Delta. To complete the proof of the lemma it suffices to show that \text{U}(n) is in \Delta. Clearly \text{U}(n) is \$-free if I is \$-free, contains r_0, and is symmetric. Suppose \tau is a substitution of L_2 in I_2 by languages of \text{U}(n). Then \tau(L_2) involves only a finite number of languages of \text{U}(n). Since \text{I}(n) is increasing in n, there exists m such that \tau is a substitution by languages of \text{I}(m). Hence

\text{Sub}(\cup \text{I}(n), z_2) = \cup \text{Sub}(I(n), z_2) \subseteq z_3.

Therefore \text{U}(n) is in \Delta and the proof is complete.
Next we have the analogue to Lemma 3.3 for the second variable in
\( \text{Sub}(\mathcal{L}_1, \mathcal{L}_2) \).

**Lemma 3.4.** Let \( \mathcal{L}_1 \) be a family of languages containing \( R_o \) and such that
\( \text{Sub}(\mathcal{L}_1, R_o) \subseteq \mathcal{L}_1 \). Let \( \mathcal{L}_3 \) be an AFL, with \( \mathcal{L}_1 \subseteq \mathcal{L}_3 \), and let \( \mathcal{L} \) be a family of
languages such that \( \text{Sub}(\mathcal{L}_1, \mathcal{L}) \subseteq \mathcal{L}_3 \). If \( \mathcal{L}_1 \) is \( \epsilon \)-free (or \( \mathcal{L}_3 \) is a full AFL),
then \( \text{Sub}(\mathcal{L}_1, \text{AFL}(\mathcal{L})) \subseteq \mathcal{L}_3 \) (or \( \text{Sub}(\mathcal{L}_1, \text{AFL}_f(\mathcal{L})) \subseteq \mathcal{L}_3 \)).

**Proof.** Let \( \Delta \) be the collection of all families \( \mathcal{L}' \) such that \( \text{Sub}(\mathcal{L}_1, \mathcal{L}') \subseteq \mathcal{L}_3 \).

Let \( \{\mathcal{L}'(n)\} \) be the sequence of families defined for \( \mathcal{L} \) in the corollary to
Proposition 1.2 (or the corollary to Proposition 1.3). Since \( \mathcal{L}'(0) = \mathcal{L} \cup R_o \) and
\( \text{Sub}(\mathcal{L}_1, R_o) \subseteq \mathcal{L}_1 \subseteq \mathcal{L}_3 \),
\( \text{Sub}(\mathcal{L}_1, \mathcal{L}'(0)) = \text{Sub}(\mathcal{L}_1, \mathcal{L}) \cup \text{Sub}(\mathcal{L}_1, R_o) \subseteq \mathcal{L}_3 \).
Therefore \( \mathcal{L}'(0) \) is in \( \Delta \). We shall show that \( \mathcal{L}' \) in \( \Delta \) implies \( \text{Sub}(R_o, \mathcal{L}') \)
(or \( \text{Sub}(R, \mathcal{L}') \), \( \text{Sub}(\mathcal{L}', R_o) \), \( \mathcal{L}' \cap R \), and \( \text{Hom}(\mathcal{L}') \)) are all in \( \Delta \).

Clearly
\[
\text{Sub}(\mathcal{L}_1, \text{Sub}(R_o, \mathcal{L}')) = \text{Sub}(\text{Sub}(\mathcal{L}_1, R_o), \mathcal{L}')
\leq \text{Sub}(\mathcal{L}_1, \mathcal{L}') \subseteq \mathcal{L}_3
\]
(or \( \text{Sub}(\mathcal{L}_1, \text{Sub}(R, \mathcal{L}')) = \text{Sub}(\text{Sub}(\mathcal{L}_1, R), \mathcal{L}')
\leq \text{Sub}(\mathcal{L}_1, \mathcal{L}') \subseteq \mathcal{L}_3 \).

and \( \text{Sub}(\mathcal{L}_1, \text{Sub}(\mathcal{L}', R_o)) \subseteq \text{Sub}(\text{Sub}(\mathcal{L}_1, \mathcal{L}'), R_o) \), by (a) of Proposition 1.1,
\( \leq \text{Sub}(\mathcal{L}_3, R_o) \)
\( \leq \mathcal{L}_3 \).

(15) Hom(\mathcal{L}') is omitted if \( \mathcal{L}_3 \) is a full AFL.
Now \( \text{Sub}(\mathcal{L}_1, \mathcal{L}' \wedge \mathcal{R}) \subseteq \text{Sub}(\mathcal{L}_1 \wedge \mathcal{R}, \mathcal{L}' \wedge \mathcal{R}) \)

\[ \subseteq \text{Hom}_\mathcal{R}(\text{Sub}(\mathcal{L}_1 \cup \mathcal{F}_0, \text{Sub}(\mathcal{F}_0, \mathcal{L}')) \wedge \mathcal{R}), \text{by Lemma 3.1}, \]

\[ = \text{Hom}_\mathcal{R}(\text{Sub}(\mathcal{L}_1, \text{Sub}(\mathcal{F}_0, \mathcal{L}')) \wedge \mathcal{R}) \]

\[ = \text{Hom}_\mathcal{R}(\text{Sub}(\text{Sub}(\mathcal{L}_1, \mathcal{F}_0), \mathcal{L}')) \wedge \mathcal{R}) \]

\[ = \text{Hom}_\mathcal{R}(\text{Sub}(\mathcal{L}_1, \mathcal{L}') \wedge \mathcal{R}) \]

\[ = \text{Hom}_\mathcal{R}(\mathcal{L}_3 \wedge \mathcal{R}) = \mathcal{L}_3 \]

(or \( \text{Sub}(\mathcal{L}_1, \mathcal{L}' \wedge \mathcal{R}) \subseteq \text{Hom}(\mathcal{L}_3 \wedge \mathcal{R}) = \mathcal{L}_3 \)).

By Lemma 3.2(b),

\[ \text{Sub}(\mathcal{L}_1, \text{Hom}_\mathcal{R}(\mathcal{L}')) \subseteq \text{Hom}_\mathcal{R}(\text{Sub}(\mathcal{L}_1 \cup \mathcal{F}_0, \text{Sub}(\mathcal{F}_0, \mathcal{L}')) \wedge \mathcal{R}) \]

\[ \subseteq \text{Hom}_\mathcal{R}(\text{Sub}(\text{Sub}(\mathcal{L}_1, \mathcal{F}_0), \mathcal{L}')) \wedge \mathcal{R}) \]

\[ \subseteq \text{Hom}_\mathcal{R}(\text{Sub}(\mathcal{L}_1, \mathcal{L}' \wedge \mathcal{R}) \wedge \mathcal{R}) \]

\[ \subseteq \mathcal{L}_3 \]

From the above, it follows that \( \mathcal{L}(n) \) is in \( \Delta \) for each \( n \geq 0 \). Then

\[ \text{Sub}(\mathcal{L}_1, \text{AFL}(\mathcal{L})) = \text{Sub}(\mathcal{L}_1 \cup \mathcal{L}(n)) \]

\[ = \bigcup_{n \geq 0} \text{Sub}(\mathcal{L}_1, \mathcal{L}(n)) \]

\[ \subseteq \mathcal{L}_3 \]

(or, similarly, \( \text{Sub}(\mathcal{L}_1, \text{AFL}(\mathcal{L})) = \mathcal{L}_3 \)) and the proof is complete.

**Lemma 3.5.** Let \( \mathcal{L}_1 \) be an \( \epsilon \)-free (arbitrary) family of languages containing \( \mathcal{F}_0 \) such that \( \text{Sub}(\mathcal{L}_1, \mathcal{F}_0) \subseteq \mathcal{L}_1 \), and let \( \mathcal{L}_2 \) be a nontrivial family of languages.

Then \( \mathcal{L}_1 \subseteq \text{AFL}(\mathcal{L}_2) \) (or \( \mathcal{L}_1 \subseteq \text{AFL}(\mathcal{L}_2) \)).

**Proof.** Let \( \mathcal{L}_2 \) be a language in \( \mathcal{L}_2 \) containing a word of length \( k \mathcal{L} \) and let \( \mathcal{L}_1 \) be a language in \( \mathcal{L}_1 \). (\( \mathcal{L}_2 \) exists since \( \mathcal{L}_2 \) is nontrivial.) Let \( \mathcal{C} \) be a new symbol. Since \( \mathcal{F}_0 \subseteq \mathcal{L}_1 \) \( (\mathcal{C}) \) is in \( \mathcal{L}_1 \). Since \( \text{Sub}(\mathcal{L}_1, \mathcal{F}_0) \subseteq \mathcal{L}_1 \), \( \mathcal{L}_1 \mathcal{C} \), hence \( \mathcal{L}_1 \mathcal{C} \cup \mathcal{C} \), is in \( \mathcal{L}_1 \). Let \( \tau \) be the substitution on \( \mathcal{L}_2 \) defined by \( \tau(a) = \mathcal{L}_1 \mathcal{C} \cup \mathcal{C} \)
for each $a$ in $E_{L_1}$. Then $\tau(L_2)$ is in $\text{Sub}(L_1, L_2)$. Thus $\tau(L_2) \cap E_{L_1}^+ c^k$ and $\tau(L_2) \cap E_{L_1}^* c^k$ are in $\text{AFL}[\text{Sub}(L_1, L_2)]$. Then $L_1 c^k = \tau(L_2) \cap E_{L_1}^+ c^k (L_1 c^k = \tau(L_2) \cap E_{L_1}^* c^k$ or $L_1 c^k = \tau(L_2) \cap E_{L_1}^* c^k$). Thus $L_1 c^k$ is in $\text{AFL}[\text{Sub}(L_1, L_2)]$.

$(\text{AFL}_f[\text{Sub}(L_1, L_2)])$. Let $h$ be the homomorphism on $(E_{L_1} \cup (c))^*$ defined by $h(c) = c$ and $h(b) = b$ for $b$ in $E_{L_1}$. Then $h$ is restricted (arbitrary homomorphism) on $L_1 c^k$ and $L_1 = h(L_1 c^k)$ is in $\text{Hom}_f(\text{AFL}(\text{Sub}(L_1, L_2))) \leq \text{AFL}(\text{Sub}(L_1, L_2))$ (or $L_1$ is in $\text{Hom}[\text{AFL}_f(\text{Sub}(L_1, L_2)) = \text{AFL}_f(\text{Sub}(L_1, L_2))]$).

We are now ready for the main result of the section.

**Theorem 3.1.** Let $L_1$ be an $\varepsilon$-free (or arbitrary) symmetric family of languages containing $R_0$ and let $L_2$ be a nontrivial family of languages. If either $\text{Sub}(R_0, L_2) \subseteq L_2$ or $\text{Sub}(L_1, R_0) \subseteq L_1$, then

$$\text{AFL}[\text{Sub}(L_1, L_2)] = \text{Sub}[\text{AFL}(L_1), \text{AFL}(L_2)]$$

(or $\text{AFL}_f[\text{Sub}(L_1, L_2)] = \text{Sub}[\text{AFL}_f(L_1), \text{AFL}_f(L_2)]$.)

**Proof.** We only consider the case when $L_1$ is $\varepsilon$-free, since the other case can be treated similarly.

By Theorem 2.1, $\text{Sub}[\text{AFL}(L_1), \text{AFL}(L_2)]$ is an AFL. Since this AFL contains $\text{Sub}(L_1, L_2)$, it follows that

$$\text{AFL}[\text{Sub}(L_1, L_2)] \subseteq \text{Sub}[\text{AFL}(L_1), \text{AFL}(L_2)].$$

Consider the reverse inclusion. First assume that $\text{Sub}(R_0, L_2) \subseteq L_2$.

Since $\text{Sub}(L_1, L_2) \subseteq \text{AFL}[\text{Sub}(L_1, L_2)]$, by Lemma 3.3 we obtain

$$(*) \quad \text{Sub}[\text{AFL}(L_1), L_2] \subseteq \text{AFL}[\text{Sub}(L_1, L_2)].$$

Since $L_2$ is a nontrivial family,

$$\text{AFL}(L_1) = \text{AFL}[\text{Sub}(\text{AFL}(L_1), L_2)],$$

by Lemma 3.5,

$$(**) \quad \subseteq \text{AFL}[\text{Sub}(L_1, L_2)],$$

by $(*)$. 

By (**), (*), and Lemma 3.4,
\[ \text{Sub}(\text{AFL}(\mathcal{L}_1), \text{AFL}(\mathcal{L}_2)) \subseteq \text{AFL}[\text{Sub}(\mathcal{L}_1, \mathcal{L}_2)]. \]

Next assume that \( \text{Sub}(\mathcal{L}_1, \mathcal{A}) \subseteq \mathcal{L}_1 \). Since \( \mathcal{L}_2 \) is a nontrivial family, \( \mathcal{L}_1 \subseteq \text{AFL}[\text{Sub}(\mathcal{L}_1, \mathcal{L}_2)] \) by Lemma 3.5. Since \( \text{Sub}(\mathcal{L}_1, \mathcal{L}_2) \subseteq \text{AFL}[\text{Sub}(\mathcal{L}_1, \mathcal{L}_2)], \)
\[ \text{Sub}[\mathcal{L}_1, \text{AFL}(\mathcal{L}_2)] \subseteq \text{AFL}[\text{Sub}(\mathcal{L}_1, \mathcal{L}_2)] \]
by Lemma 3.4. Then, by Lemma 3.3,
\[ \text{Sub}(\text{AFL}(\mathcal{L}_1), \text{AFL}(\mathcal{L}_2)) \subseteq \text{AFL}[\text{Sub}(\mathcal{L}_1, \mathcal{L}_2)]. \]

Remarks. (1) The proof of Theorem 3.1 shows that the inclusion
\[ \text{AFL}[\text{Sub}(\mathcal{L}_1, \mathcal{L}_2)] \subseteq \text{Sub}(\text{AFL}(\mathcal{L}_1), \text{AFL}(\mathcal{L}_2)) \]
(or \( \text{AFL}[\text{Sub}(\mathcal{L}_1, \mathcal{L}_2)] \subseteq \text{Sub}(\text{AFL}(\mathcal{L}_1), \text{AFL}(\mathcal{L}_2)) \))
is valid with no hypotheses on \( \mathcal{L}_1 \) or \( \mathcal{L}_2 \).

(2) The reverse inclusion in remark (1), thus Theorem 3.1, is not valid without some hypotheses. For example, if \( \mathcal{L}_1 \) is the trivial family consisting of just \( \{e\} \), then \( \text{Sub}(\mathcal{L}_1, \mathcal{L}_2) = \mathcal{L}_1 \) and \( \text{AFL}[\text{Sub}(\mathcal{L}_1, \mathcal{L}_2)] = \mathcal{R} = \text{AFL}[\text{Sub}(\mathcal{L}_1, \mathcal{L}_2)] \)
ned not contain \( \text{Sub}(\text{AFL}(\mathcal{L}_1), \text{AFL}(\mathcal{L}_2)) = \text{AFL}(\mathcal{L}_2) = \text{Sub}(\text{AFL}(\mathcal{L}_1), \text{AFL}(\mathcal{L}_2)) \).

Similarly, if \( \mathcal{L}_2 \) is the trivial family, then \( \text{AFL}[\text{Sub}(\mathcal{L}_1, \mathcal{L}_2)] = \mathcal{R} = \text{AFL}[\text{Sub}(\mathcal{L}_1, \mathcal{L}_2)] \)
ned not contain \( \text{Sub}(\text{AFL}(\mathcal{L}_1), \mathcal{R}) \) which contains \( \text{AFL}(\mathcal{L}_1) \). If \( \mathcal{L}_1 = a^+ \) and \( \mathcal{L}_2 \) is the AFL generated by \( \{a^nb^n | n \geq 1\} \), then \( \text{Sub}(\mathcal{R}, \mathcal{L}_2) = \mathcal{L}_2 \), \( \text{AFL}[\text{Sub}(\mathcal{L}_1, \mathcal{L}_2)] \)
\( \subseteq \text{AFL}(\mathcal{L}_2) = \mathcal{R} \), and \( \text{Sub}(\text{AFL}(\mathcal{L}_1), \text{AFL}(\mathcal{L}_2)) = \text{Sub}(\mathcal{R}, \mathcal{L}_2) = \mathcal{L}_2 \).

Section 4. Iterated substitution

We are interested in families of languages closed with respect to substitution into another family, or closed with respect to substitution by languages.
of another family. If a family lacks the desired closure property, then we apply the substitutions in question and enlarge the original family by adding the languages obtained by these substitutions. Iterating this procedure leads to a family containing the original one and having the desired closure property.

This section contains the definitions and some results about iterated substitution. In particular, we show that iterated substitution applied to an AFL yields an AFL. We present a condition that guarantees the substitution closure of a family to be an AFL even if the original family is not an AFL.

Definition. Let $L_1$ and $L_2$ be families of languages. Then $L_2$ is closed under $L_1$ substitution if $\text{Sub}(L_1, L_2) \subseteq L_2$. The closure of $L_2$ under $L_1$ substitution is the smallest family containing $L_2$ and closed under $L_1$ substitution.

Since the intersection of all families containing $L_2$ and closed under $L_1$ substitution (the family of all sets $L \subseteq \Sigma^* \subseteq L^*$ is one such family) is also such a family, the closure exists. In Lemma 4.1, we shall show how to calculate it.

Notation. Given $L_1$ and $L_2$, let $\text{Sub}^0(L_1, L_2) = L_2$ and, by induction, $\text{Sub}^{k+1}(L_1, L_2) = \text{Sub}(L_1, \bigcup_{i=0}^{k} \text{Sub}^i(L_1, L_2))$ for each $k \geq 0$. Let $\text{Sub}^\infty(L_1, L_2) = \bigcup_{i=0}^{\infty} \text{Sub}^i(L_1, L_2)$.

Note that for every $L_1$, $L_2$, and $k$,

$$\text{Sub}(L_1, \text{Sub}^k(L_1, L_2)) \subseteq \bigcup_{i=0}^{k} \text{Sub}(L_1, \text{Sub}^i(L_1, L_2))$$

$$= \text{Sub}(L_1, \bigcup_{i=0}^{k} \text{Sub}^i(L_1, L_2))$$

$$\subseteq \text{Sub}^{k+1}(L_1, L_2).$$
Also, for each k ≥ 1,
\[ \text{Sub}^k(\mathcal{L}_1, \mathcal{L}_2) = \text{Sub}[\mathcal{L}_1, \bigcup_{i=0}^{k-1} \text{Sub}^i(\mathcal{L}_1, \mathcal{L}_2)] \]
\[ \subseteq \text{Sub}[\mathcal{L}_1, \bigcup_{i=0}^{k} \text{Sub}^i(\mathcal{L}_1, \mathcal{L}_2)] \]
\[ = \text{Sub}^{k+1}(\mathcal{L}_1, \mathcal{L}_2). \]

It is not necessarily true that \( \mathcal{L}_2 = \text{Sub}^0(\mathcal{L}_1, \mathcal{L}_2) \subseteq \text{Sub}^1(\mathcal{L}_1, \mathcal{L}_2) \). [For example, let \( \mathcal{L}_1 = \{(ab)\} \) and \( \mathcal{L}_2 = \{(a)\} \). Thus \( \text{Sub}^1(\mathcal{L}_1, \mathcal{L}_2) = \mathcal{L}_1 \), which does not contain \( \mathcal{L}_2 = \text{Sub}^0(\mathcal{L}_1, \mathcal{L}_2) \).]

Suppose \( \mathcal{L}_1 \) contains \( \{a\} \) for each \( a \) in \( \mathcal{I} \). Then \( \mathcal{L}_2 \subseteq \text{Sub}(\mathcal{L}_1, \mathcal{L}_2) \) for every family \( \mathcal{L}_2 \). Thus, for every \( k \geq 0 \),
\[ \text{Sub}^{k+1}(\mathcal{L}_1, \mathcal{L}_2) = \text{Sub}[\mathcal{L}_1, \bigcup_{i=0}^{k} \text{Sub}^i(\mathcal{L}_1, \mathcal{L}_2)] \]
\[ = \text{Sub}[\mathcal{L}_1, \text{Sub}^{k+1}(\mathcal{L}_1, \mathcal{L}_2)]. \]

**Lemma 4.1.** The family \( \text{Sub}^\mathcal{I}(\mathcal{L}_1, \mathcal{L}_2) \) is the closure of \( \mathcal{L}_2 \) under \( \mathcal{L}_1 \) substitution.

**Proof.** Let \( \mathcal{L} \) be the closure of \( \mathcal{L}_2 \) with respect to \( \mathcal{L}_1 \) substitution. Since
\[ \text{Sub}[\mathcal{L}_1, \text{Sub}^\mathcal{I}(\mathcal{L}_1, \mathcal{L}_2)] = \text{Sub}[\mathcal{L}_1, \bigcup_{i=0}^{\infty} \text{Sub}^i(\mathcal{L}_1, \mathcal{L}_2)] \]
\[ = \bigcup_{i=0}^{\infty} \text{Sub}[\mathcal{L}_1, \text{Sub}^i(\mathcal{L}_1, \mathcal{L}_2)] \]
\[ = \bigcup_{i=0}^{\infty} \text{Sub}^{i+1}(\mathcal{L}_1, \mathcal{L}_2) \]
\[ = \text{Sub}^\mathcal{I}(\mathcal{L}_1, \mathcal{L}_2). \]

Thus \( \text{Sub}^\mathcal{I}(\mathcal{L}_1, \mathcal{L}_2) \) is closed under \( \mathcal{L}_1 \) substitution, so that \( \mathcal{L} \subseteq \text{Sub}^\mathcal{I}(\mathcal{L}_1, \mathcal{L}_2) \).

To see the reverse containment, it suffices to show that \( \text{Sub}^k(\mathcal{L}_1, \mathcal{L}_2) \subseteq \mathcal{L} \) for each \( k \geq 0 \). By definition, \( \text{Sub}^0(\mathcal{L}_1, \mathcal{L}_2) = \mathcal{L}_2 \subseteq \mathcal{L} \). Continuing by induction, suppose \( \text{Sub}^i(\mathcal{L}_1, \mathcal{L}_2) \subseteq \mathcal{L} \) for \( 0 < i < k \), \( k \geq 1 \). Then
Sub^k(f_1, f_2) = Sub(f_1, \bigcup_{i=0}^{k-1} Sub^i(f_1, f_2)) \subseteq Sub(f_1, f) \subseteq f.

Thus the induction is extended and the proof is complete.

**Definition.** Let \( f_1 \) and \( f_2 \) be families of languages. A family \( f_1 \) is **closed** under substitution in \( f_2 \) if \( Sub(f_1, f_2) \subseteq f_1 \). The **closure** of \( f_1 \) under substitution in \( f_2 \) is the smallest family containing \( f_1 \) and closed under substitution in \( f_2 \).

Again, it is easily seen that the closure exists.

**Notation.** Given \( f_1 \) and \( f_2 \), let \( Sub_0(f_1, f_2) = f_1 \) and, by induction,

\[
Sub_{k+1}(f_1, f_2) = Sub(\bigcup_{i=0}^{k} Sub_i(f_1, f_2), f_2)
\]

for each \( k \geq 0 \). Let \( Sub_\infty(f_1, f_2) = \bigcup_{i=0}^{\infty} Sub_i(f_1, f_2) \).

Again, \( Sub_k(f_1, f_2) \subseteq Sub_{k+1}(f_1, f_2) \) for \( k \geq 1 \). If \( f_2 \) contains \( \{a\} \) for some \( a \) in \( \Sigma \), then \( Sub_0(f_1, f_2) = f_1 \subseteq Sub_1(f_1, f_2) \).

**Lemma 4.2.** The family \( Sub_\infty(f_1, f_2) \) is the closure of \( f_1 \) under substitution in \( f_2 \).

**Proof.** Let \( f \) be the closure of \( f_1 \) under substitution in \( f_2 \). As in the proof of Lemma 4.1, it is easily seen that \( Sub_k(f_1, f_2) \subseteq f \) for each \( k \geq 0 \). Hence \( Sub_\infty(f_1, f_2) \subseteq f \). To see the reverse containment, we have

\[
Sub(Sub_\infty(f_1, f_2), f_2) = Sub(\bigcup_{i=0}^{\infty} Sub_k(f_1, f_2), f_2).
\]

Let \( L_2 \) be in \( f_2 \) and \( \tau \) a substitution of \( L_2 \) by languages of \( \bigcup_{i=0}^{\infty} Sub_k(f_1, f_2) \).

Since \( L_2 \) is finite, there exists \( m \geq 0 \) such that \( \tau(a) \) is in \( \bigcup_{i=0}^{m} Sub_k(f_1, f_2) \) for each \( a \) in \( \Sigma_2 \). Therefore \( \tau(L_2) \) is in \( Sub(Sub_\infty(f_1, f_2), f_2) \).

Thus \( Sub_{m+1}(f_1, f_2) = Sub(m+1(f_1, f_2) \subseteq Sub_\infty(f_1, f_2) \).
Sub[Sub(\mathcal{L}_1, \mathcal{L}_2), \mathcal{L}_2] = Sub(\mathcal{L}_1, \mathcal{L}_2),

so that \mathcal{L} \subseteq Sub(\mathcal{L}_1, \mathcal{L}_2).

**Definition.** A family \mathcal{L} is substitution closed if Sub(\mathcal{L}, \mathcal{L}) = \mathcal{L}. The substitution closure of \mathcal{L} is the smallest substitution closed family \mathcal{L}_\infty containing \mathcal{L}.

Clearly \mathcal{L}_\infty exists.

**Theorem 4.1.** (a) For each family \mathcal{L}, \mathcal{L}_\infty = Sub(\mathcal{L}, \mathcal{L}).

(b) If \mathcal{L} is a symmetric family such that \mathcal{L} = Sub(\mathcal{L}, \mathcal{L}), then \mathcal{L}_\infty = Sub(\mathcal{L}, \mathcal{L}).

**Proof.** (a) Since \mathcal{L}_\infty is substitution closed and contains \mathcal{L}, it is closed under substitution in \mathcal{L}. By Lemma 4.2, Sub(\mathcal{L}, \mathcal{L}) = \mathcal{L}.

To complete the proof of (a), it suffices to show that Sub(\mathcal{L}, \mathcal{L}) is substitution closed. [For this will imply that \mathcal{L}_\infty \subseteq Sub(\mathcal{L}, \mathcal{L}).] Since Sub(\mathcal{L}, \mathcal{L}) is the closure of \mathcal{L} under substitution in \mathcal{L},

\[ Sub[Sub(\mathcal{L}, \mathcal{L}), Sub(\mathcal{L}, \mathcal{L})] = Sub[Sub(\mathcal{L}, \mathcal{L}), \mathcal{L}] = Sub(\mathcal{L}, \mathcal{L}). \]

Continuing by induction, assume \mathcal{L} = Sub(\mathcal{L}, \mathcal{L}) for all 0 \leq j < n. Then for \mathcal{L}' = \bigcup_{j=0}^{n-1} Sub(\mathcal{L}, \mathcal{L}),

\[ Sub[Sub(\mathcal{L}, \mathcal{L}), \mathcal{L}'] = \bigcup_{j=0}^{n-1} Sub[Sub(\mathcal{L}, \mathcal{L}), Sub(\mathcal{L}, \mathcal{L})] \subseteq Sub(\mathcal{L}, \mathcal{L}). \]

Furthermore, Sub(\mathcal{L}, \mathcal{L}) = Sub(\mathcal{L}', \mathcal{L}). Thus

\[ Sub[Sub(\mathcal{L}, \mathcal{L}), Sub(\mathcal{L}, \mathcal{L})] = Sub(Sub(\mathcal{L}, \mathcal{L}), Sub(\mathcal{L}', \mathcal{L})) \subseteq Sub(Sub(Sub(\mathcal{L}, \mathcal{L}), \mathcal{L}'), \mathcal{L}), \]

by (a) of Proposition 1.1,

\[ = Sub(Sub(\mathcal{L}, \mathcal{L}), \mathcal{L}) \subseteq Sub(\mathcal{L}, \mathcal{L}). \]
Hence the induction is extended. Therefore

\[ \text{Sub}[\text{Sub}_k(x,x), \text{Sub}_k(x,x)] = \text{Sub}_k(x,x) \]

for all \( k > 0 \), so that

\[ \text{Sub} \left[ \bigcup_{k=0}^\infty \text{Sub}[\text{Sub}_k(x,x), \text{Sub}_k(x,x)] \right] = \text{Sub}_\infty(x,x). \]

(b) Since \( \mathcal{L}_\infty \) is substitution closed and contains \( \mathcal{L} \), it is closed under \( \mathcal{L} \) substitution. By Lemma 4.1, \( \text{Sub}^{\infty}(x,x) = \mathcal{L}_\infty \).

To complete the proof of (b), it suffices to show that \( \text{Sub}^{\infty}(x,x) \) is substitution closed. By hypothesis, \( \text{Sub}^0(x,x) = \mathcal{L} \subseteq \text{Sub}^{1}(x,x) \). Thus \( \text{Sub}^{k}(x,x) \subseteq \text{Sub}^{k+1}(x,x) \) for all \( k > 0 \). Hence \( \text{Sub}^{k+1}(x,x) = \text{Sub}[\mathcal{L}, \text{Sub}^{k}(x,x)] \)

for all \( k > 0 \). Obviously \( \text{Sub}^{k+1}(x,x) \) is symmetric for all \( k > 0 \). By Lemma 4.1,

\[ \text{Sub}[x, \text{Sub}^{\infty}(x,x)] = \text{Sub}^{\infty}(x,x) \]

Continuing by induction, assume that

\[ \text{Sub}\left[\text{Sub}_k(x,x), \text{Sub}^{\infty}(x,x)\right] = \text{Sub}^{\infty}(x,x) \]

for \( 0 \leq k < n, n \geq 1 \). Then

\[ \text{Sub}\left[\text{Sub}^n(x,x), \text{Sub}^{\infty}(x,x)\right] = \text{Sub}\left[\text{Sub}(x, \text{Sub}^{n-1}(x,x)), \text{Sub}^{\infty}(x,x)\right] \]

by Proposition 1.1,

\[ \subseteq \text{Sub}[x, \text{Sub}^{\infty}(x,x)] \]

Therefore the induction is extended, so that

\[ \text{Sub}\left[\text{Sub}_k(x,x), \text{Sub}^{\infty}(x,x)\right] = \text{Sub}^{\infty}(x,x). \]

Now let \( L \) be in \( \text{Sub}^{\infty}(x,x) \) and \( \tau \) a substitution on \( L \) by languages in
Sub^oo(L,L) = \bigcup_{k=0}^{\infty} \text{Sub}^k(L,L). \text{ There exists } m \geq 0 \text{ such that } \tau(a) \text{ is in } \bigcup_{k=0}^{m} \text{Sub}^k(L,L) = \\
Sub^m(L,L) \text{ for each } a \text{ in } \Sigma_L. \text{ Thus } \tau(L) \text{ is in Sub}[\text{Sub}^m(L,L), \text{Sub}^\infty(L,L)]. \text{ Hence } \\
\text{Sub}[\text{Sub}^\infty(L,L), \text{Sub}^\infty(L,L)] = \bigcup_{k=0}^{\infty} \text{Sub}[\text{Sub}^k(L,L), \text{Sub}^\infty(L,L)] \\
= \text{Sub}^\infty(L,L),

completing the proof.

We do not know to what extent the hypotheses in the theorems of Section 4 can be weakened. In particular, we do not know if the hypotheses of Theorem 4.1 can be weakened. In general, (b) of Theorem 4.1 is not valid without some hypotheses on L, i.e., Sub^\infty(L,L) is not always substitution closed. For example, let L = \{(ab)/a,b \text{ in } L\}. \text{ Then Sub}^\infty(L,L) \text{ consists of all words of length } 2^n \text{ for all } n \geq 1. \text{ However, the substitution closure of } L \text{ contains } \{a\} \text{ for each } a \text{ in } \Sigma.

The rest of this section is concerned with relations between the various substitution closures and AFL.

Theorem 4.2. If L_1 and L_2 are AFL, then so are Sub^\infty(L_1, L_2) and Sub^\infty(L_1, L_2). If, in addition, L_1 is full, then so are Sub^\infty(L_1, L_2) and Sub^\infty(L_1, L_2).

Proof. Since L_1 contains (a) for each a in \Sigma, Sub^k(L_1, L_2) = Sub^k(L_1, L_2). Similarly, Sub^k(L_1, L_2) = Sub^k(L_1, L_2).

By Theorem 2.1 and induction, Sub^k(L_1, L_2) and Sub^k(L_1, L_2) are AFL for all k (and are full if L_1 is full, by Corollary 1 of Theorem 2.1). Therefore Sub^\infty(L_1, L_2) and Sub^\infty(L_1, L_2) are AFL (and full if L_1 is full).

From Theorems 4.1 and 4.2 we get

Corollary. For each (full) AFL L, L_\infty = Sub^\infty(L,L) = Sub^\infty(L,L) and is a(full) AFL.

Remarks. (1) For AFL L_1 and L_2, Sub^\infty(L_1, L_2) and Sub^\infty(L_1, L_2) are not necessarily
the same. For let \( L_1 = R_o \) and \( L_2 \) be the family of context-free languages.
Thus \( \text{Sub}_o(L_1, L_2) = L_2 \), \( \text{Sub}_o(L_1, L_2) = L_1 \), and \( L_1 \neq L_2 \).

(2) For well-known AFL there arise the general problems of "identifying"
\( \text{Sub}_o(L_1, L_2) \) and \( \text{Sub}_o(L_1, L_2) \).

Theorem 4.3. Let \( \mathcal{L} \) be an \( \epsilon \)-free (or arbitrary) symmetric family of languages
containing \( R_o \) and such that either \( \text{Sub}(R_o, \mathcal{L}) \subseteq \mathcal{L} \) or \( \text{Sub}(\mathcal{L}, R_o) \subseteq \mathcal{L} \). Then
\( \text{AFL}(\mathcal{L}) = [\text{AFL}(\mathcal{L})]_o \) (or \( \text{AFL}_x(\mathcal{L}) = [\text{AFL}_x(\mathcal{L})]_o \)).

Proof. We shall prove the theorem for the \( \epsilon \)-free case, an analogous argument
holding for the arbitrary \( \mathcal{L} \) case.

Note that since \( R_o \subseteq \mathcal{L} \) and \( R_o \subseteq \text{AFL}(\mathcal{L}) \), \( \mathcal{L} \subseteq \text{Sub}(\mathcal{L}, \mathcal{L}) \) and \( \text{AFL}(\mathcal{L}) \subseteq \text{Sub}[\text{AFL}(\mathcal{L}), \text{AFL}(\mathcal{L})] \).

Hence, for each \( n > 0 \),

\[ \text{Sub}_n(\mathcal{L}, \mathcal{L}) = \text{Sub}[\text{Sub}_{n-1}(\mathcal{L}, \mathcal{L}), \mathcal{L}] \]

and

\[ \text{Sub}_n[\text{AFL}(\mathcal{L}), \text{AFL}(\mathcal{L})] = \text{Sub}[\text{Sub}_{n-1}[\text{AFL}(\mathcal{L}), \text{AFL}(\mathcal{L})], \text{AFL}(\mathcal{L})]. \]

We first show that for each \( n > 0 \), \( \text{AFL}[\text{Sub}_{n-1}(\mathcal{L}, \mathcal{L})] = \text{Sub}[\text{AFL}(\mathcal{L}), \text{AFL}(\mathcal{L})] \).

For \( n = 0 \), \( \text{AFL}[\text{Sub}_{n}(\mathcal{L}, \mathcal{L})] = \text{AFL}(\mathcal{L}) = \text{Sub}[\text{AFL}(\mathcal{L}), \text{AFL}(\mathcal{L})] \). Continuing by
induction, suppose \( n > 0 \) and that \( \text{AFL}[\text{Sub}_{n-1}(\mathcal{L}, \mathcal{L})] = \text{Sub}_{n-1}[\text{AFL}(\mathcal{L}), \text{AFL}(\mathcal{L})] \).

Then

\[ \text{AFL}[\text{Sub}_{n}(\mathcal{L}, \mathcal{L})] = \text{AFL}[\text{Sub}_{n-1}(\mathcal{L}, \mathcal{L}), \mathcal{L}], \]

by \((*)\),

\[ = \text{Sub}[\text{AFL}[\text{Sub}_{n-1}(\mathcal{L}, \mathcal{L})], \text{AFL}(\mathcal{L})], \]

by Theorem 3.1,

\[ = \text{Sub}[\text{Sub}_{n-1}[\text{AFL}(\mathcal{L}), \text{AFL}(\mathcal{L})], \text{AFL}(\mathcal{L})], \]

by induction,

\[ = \text{Sub}_{n}[\text{AFL}(\mathcal{L}), \text{AFL}(\mathcal{L})], \]

by \((**)\),

extending the induction.

(17) Since \( \mathcal{L} \) is symmetric, \( \text{Sub}_{n-1}(\mathcal{L}, \mathcal{L}) \) is symmetric. If \( \text{Sub}(R_o, \mathcal{L}) \subseteq \mathcal{L} \), then
the hypotheses of Theorem 3.1 are obviously satisfied. Suppose \( \text{Sub}(\mathcal{L}, R_o) \subseteq \mathcal{L} \).
A simple induction, using Proposition 1.1, shows that \( \text{Sub}[\text{Sub}(\mathcal{L}, \mathcal{L}), R_o] \subseteq \text{Sub}(\mathcal{L}, \mathcal{L}) \) for each \( k \geq 0 \). Thus, in this case also, the hypotheses of Theorem 3.1
are satisfied.
To complete the proof, we see that \( \text{AFL}(I) = \text{AFL}(\text{Sub}(I, I)) = \text{AFL} \left( \bigcup_n \text{Sub}_n(I, I) \right) = \bigcup_n \text{AFL}(I) \) = \( \text{AFL}(I) \).

**Corollary.** If \( I \) is a substitution closed AFL, then the full AFL generated by \( I \) is also substitution closed.

**Proof.** By Theorem 6.3, \( \text{AFL}_f(I) = \text{AFL}_f(I) \). Since \( I \) is substitution closed, \( I = I \). Hence \( \text{AFL}_f(I) = \text{AFL}_f(I) \), so that \( \text{AFL}_f(I) \) is substitution closed.

Our final theorem provides a criterion for \( I \) to be an AFL even if \( I \) is not.

**Theorem 6.4.** (a) Let \( I \) be a family of languages containing \( L \) for every finite \( \Sigma^+ \subseteq \Sigma \) and such that \( \text{Sub}(\Sigma^+, I) \subseteq I \). Then \( I \) is an AFL if and only if \( \text{Hom}(I, R) \subseteq I \).

(b) Let \( I \) be a family of languages containing \( \Sigma_0 \) for every finite \( \Sigma_0 \subseteq \Sigma \) and such that \( \text{Sub}(\Sigma_0, I) \subseteq I \). Then \( I \) is a full AFL if and only if \( \text{Hom}(I, R) \subseteq I \), or equivalently, if and only if \( \text{Hom}(\Sigma^+ \cap R) \subseteq I \).

**Proof.** (a) Suppose \( I \) is an AFL. Then \( \text{Hom}(I, R) \subseteq I \). Since \( I \subseteq I \), \( \text{Hom}(I, R) \subseteq I \).

To prove the sufficiency, assume \( \text{Hom}(\Sigma^+ \cap R) \subseteq I \). Let \( L \) be in \( R \).

Then \( \Sigma^+ \cap R \) is in \( I \), so that \( L = \Sigma^+ \cap R \subseteq \text{Hom}(\Sigma^+ \cap R) \subseteq I \). Thus \( R \subseteq I \). Since \( I \) is substitution closed,

\[
\text{Sub}(R, I) \subseteq \text{Sub}(I, I) \subseteq I
\]

and

\[
\text{Sub}(R, I) \subseteq \text{Sub}(I, I) \subseteq I
\]

Thus, to show that \( I \) is an AFL, it suffices to verify that \( I \subseteq I \) and \( \text{Hom}(I) \subseteq I \). Since \( (I \cap R) \cup \text{Hom}(I) \subseteq \text{Hom}(I \cap R) \), it
suffices to show that $\text{Hom}_r(\mathcal{L}_\infty \land R) \subseteq \mathcal{L}_\infty$.

We first show that $\text{Hom}_r[\text{Sub}_n(\mathcal{L}, \mathcal{L}) \land R] \subseteq \mathcal{L}_\infty$ for each $n > 0$. For $n = 0$,

$\text{Hom}_r[\text{Sub}_0(\mathcal{L}, \mathcal{L}) \land R] = \text{Hom}_r(\mathcal{L} \land R) \subseteq \mathcal{L}_\infty$, by hypothesis. Assume $n > 0$ and

$\text{Hom}_r[\text{Sub}_0(\mathcal{L}, \mathcal{L}) \land R] \subseteq \mathcal{L}_\infty$ for all $J$, $0 \leq J < n$. Let $\mathcal{L}' = \bigcup_{j=0}^{n-1} \text{Sub}_j(\mathcal{L}, \mathcal{L})$. Then

$\text{Hom}_r(\mathcal{L}' \land R) \subseteq \mathcal{L}_\infty$ and, by definition, $\text{Sub}_n(\mathcal{L}, \mathcal{L}) = \text{Sub}(\mathcal{L}', \mathcal{L})$. Therefore

$\text{Hom}_r[\text{Sub}_n(\mathcal{L}, \mathcal{L}) \land R] = \text{Hom}_r[\text{Sub}(\mathcal{L}', \mathcal{L}) \land R]$

$\subseteq \text{Hom}_r[\text{Sub}(\mathcal{L}' \land R), \text{Sub}(\mathcal{F}_0, \mathcal{L} \land R)]$, by Lemma 2.1,

$\subseteq \text{Hom}_r[\text{Sub}(\mathcal{L}' \land R), \mathcal{L} \land R]$, since $\text{Sub}(\mathcal{F}_0, \mathcal{L}) \subseteq \mathcal{L}$,

$\subseteq \text{Sub}(\text{Hom}_r(\mathcal{L}' \land R), \text{Hom}_r(\text{Sub}(\mathcal{F}_0, \mathcal{L} \land R)))$, by Lemma 2.2,

$\subseteq \text{Sub}(\mathcal{L}_\infty, \text{Hom}_r(\text{Sub}(\mathcal{F}_0, \mathcal{L} \land R)))$,

since $R \land R = R$ and $\text{Hom}_r(\mathcal{L}' \land R) \subseteq \mathcal{L}_\infty$.

Now

$\text{Hom}_r[\text{Sub}(\mathcal{F}_0, \mathcal{L} \land R)] = \text{Hom}_r[\text{Sub}(\mathcal{F}_0 \land R, \mathcal{L} \land R)]$, since $\mathcal{F}_0 \land R = \mathcal{F}_0$,

$\subseteq \text{Hom}_r(\mathcal{F}_0 \land R, \mathcal{L} \land R)$, by Lemma 3.1,

$\subseteq \text{Hom}_r[\text{Sub}(\mathcal{F}_0, \mathcal{L}) \land R]$, since $\text{Sub}(\mathcal{F}_0, \mathcal{L}) \subseteq \mathcal{L}$

and $\text{Hom}_r(\mathcal{F}_0 \land R) = \text{Hom}_r(\mathcal{L} \land R)$,

$\subseteq \mathcal{L}_\infty$.

Therefore

$\text{Hom}_r[\text{Sub}_n(\mathcal{L}, \mathcal{L}) \land R] \subseteq \text{Sub}(\mathcal{L}_\infty, \mathcal{L}_\infty)$

$\subseteq \mathcal{L}_\infty$,

so that the induction is extended.

To complete the proof, we have
\[ \text{Hom}_T(\mathcal{L}^\wedge \mathcal{R}) = \text{Hom}_T(\bigcup_{A \in \mathcal{R}} \text{Sub}(\mathcal{L}, \mathcal{L}^\wedge \mathcal{R})) \text{, by Theorem 4.1,} \]
\[ = \text{Hom}_T(\bigcup_{A \in \mathcal{R}} \text{Sub}(\mathcal{L}) \wedge \mathcal{R}) \]
\[ \subseteq \mathcal{L}^\infty. \]

(b) It suffices to show that "only if." Thus assume either \( \text{Hom}(\mathcal{L} \wedge \mathcal{R}) = \mathcal{L}^\infty \text{ or } \mathcal{L} \wedge \mathcal{R} = \mathcal{L}^\infty. \) Clearly \( \mathcal{R} = \mathcal{L}^\infty. \) Thus \( \mathcal{L} \) is closed under arbitrary homomorphism since it is closed under substitution. Hence \( \text{Hom}(\mathcal{L}^\wedge \mathcal{R}) = \mathcal{L}^\infty \) if \( \mathcal{L}^\wedge \mathcal{R} = \mathcal{L}^\infty. \) (Obviously, \( \mathcal{L} \wedge \mathcal{R} \subseteq \mathcal{L}^\infty \) if \( \text{Hom}(\mathcal{L} \wedge \mathcal{R}) \subseteq \mathcal{L}^\infty \)) By (a), \( \mathcal{L}^\infty \) is an AFL.

Since \( \mathcal{L}^\infty \) is closed under arbitrary homomorphism, it is a full AFL.

The above theorem gives another proof of the following result [6].

**Corollary.** The family of derivation-bounded languages is a full AFL.

**Proof.** It is known [6] that the family of derivation-bounded languages is the substitution closure of \( \mathcal{L}_{\text{lin}}^\infty \) the family of linear context-free languages. Since \( \mathcal{L}_{\text{lin}}^\infty \) contains \( \mathcal{R} \) and is closed under intersection with regular sets, and \( \text{Sub}(\mathcal{L}^\infty, \mathcal{L}_{\text{lin}}) \subseteq \mathcal{L}_{\text{lin}}^\infty \) the substitution closure of \( \mathcal{L}_{\text{lin}} \) is a full AFL by Theorem 4.4.


The effect of substitution in families of languages, especially AFL, is considered. Among the main results shown are the following: The substitution of one AFL into another is an AFL. Under suitable hypotheses, the AFL generated by the family obtained from the substitution of one family into another, is the family obtained from the substitution of the corresponding AFL. A condition is given for the AFL generated by the substitution closure of a family to be the substitution closure of the AFL generated by the family.
<table>
<thead>
<tr>
<th>KEY WORDS</th>
<th>LINK A</th>
<th>LINK B</th>
<th>LINK C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Families of languages</td>
<td>ROLE</td>
<td>WT</td>
<td>ROLE</td>
</tr>
<tr>
<td>Abstract family of languages (AFL)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AFL</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Substitution closure</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>