UNIVERSITY OF WATERLOO COMPUTER SCIENCE DEPARTMENT
UNIVERSITY OF WATERLOO COMPUTER SCIENCE DEPARTMENT
UNIVERSITY OF WATERLOO COMPUTER SCIENCE DEPARTMENT
UNIVERSITY OF WATERLOO COMPUTER SCIENCE DEPARTMENT

On Generalized Locally Testable Languages

J.A. Brzozowski
Faith E. Fich

CS-82-05

February, 1982

On Generalized Locally Testable Languages

J.A. Brzozowski

Computer Science Department
University of Waterloo
Waterloo, Ontario, Canada

Faith E. Fich

Computer Science Division
University of California
Berkeley, California, U.S.A.

ABSTRACT

This paper discusses a family of star-free languages, over a two letter alphabet, which generalizes both the locally testable languages and the $\mathcal{G}$-trivial languages. Characterizations are given in terms of congruences, monoids, and semiautomata.

This research was supported by the Natural Science and Engineering Research Council of Canada under grant no. A1617 and under a Postgraduate Scholarship and also by National Science Foundation grant MCS79-15763.

1. Motivation

The regular languages over a finite alphabet Σ are those sets which can be built up from finite and cofinite subsets of Σ^* using Boolean operations, concatenation, and the star operator. Star-free languages are those regular languages which can be obtained without the use of the star operator. A family or star-free languages that has been studied extensively is the family of locally testable languages [BS, E, M].

Locally testable languages can be defined by certain congruences of finite index. If $x \in \Sigma^*$ then $|x|$ denotes the length of x. For $r \geq 0$, let xf_r, the *front of length* r of x, denote the prefix of x of length r or x if $|x| < r$. Similarly, let xt_r, the *tail of length* r of x, denote the suffix of x of length r or x if $|x| < r$. Also define $xm_r = \{ v \mid x = uvw \text{ and } |v| = r \}$ to be the *set of all subsegments of* x *of length* r. Then define the following congruences $\sim_r$ on Σ^+:

$$x \sim_r y \text{ if and only if } xf_{r-1} = yf_{r-1},\ xt_{r-1} = yt_{r-1}, \text{ and } xm_r = ym_r\ . \quad (1)$$

A language L is locally testable if and only if it is a $\sim_r$ language (i.e. it is a union of congruence classes of $\sim_r$), for some $r \geq 1$.

If in (1) we remove the condition $xm_r = ym_r$, i.e. if we test only the fronts and tails for equality, we obtain the family of generalized definite languages. Further, if only the tails (fronts) are tested, we obtain the family of definite (reverse definite) languages. The intersection of the family of definite languages with the family of reverse definite languages is the family of finite/cofinite languages. This is summarized in Figure 1(a).

The finite/cofinite, definite, reverse definite, generalized definite, and locally testable languages have natural characterizations in terms of their syntactic semigroups. More precisely, a language $L \subseteq \Sigma^+$ is in a particular family of Figure 1(a) if and only if its syntactic semigroup S is finite and satisfies the corresponding property of Figure 1(b), for every idempotent $e \in S$.

The condition "$eSe = e$ for all idempotents $e \in S$" can be generalized as follows. It turns out to be more convenient to deal with monoids rather than semigroups. If M is a monoid and $f \in M$, define $P_f = \{g \mid f \in MgM\}$ and define M_f to be the submonoid of M generated by P_f. Now the top four conditions of Figure 1(c) define some very well-known families of monoids, namely the J-trivial, L-trivial, and R-trivial monoids of classical semigroup theory, and the recently studied G-trivial monoids [BF, E, F, FB, S]. We will call a language J-trivial if and only if its syntactic monoid is J-trivial, etc. The purpose of this paper is to study the family of subset of Σ^* whose syntactic monoids M satisfy the condition that eM_ee is idempotent and commutative for every idempotent $e \in M$. We call these generalized locally testable languages; they generalize both the locally testable languages and the G-trivial languages.

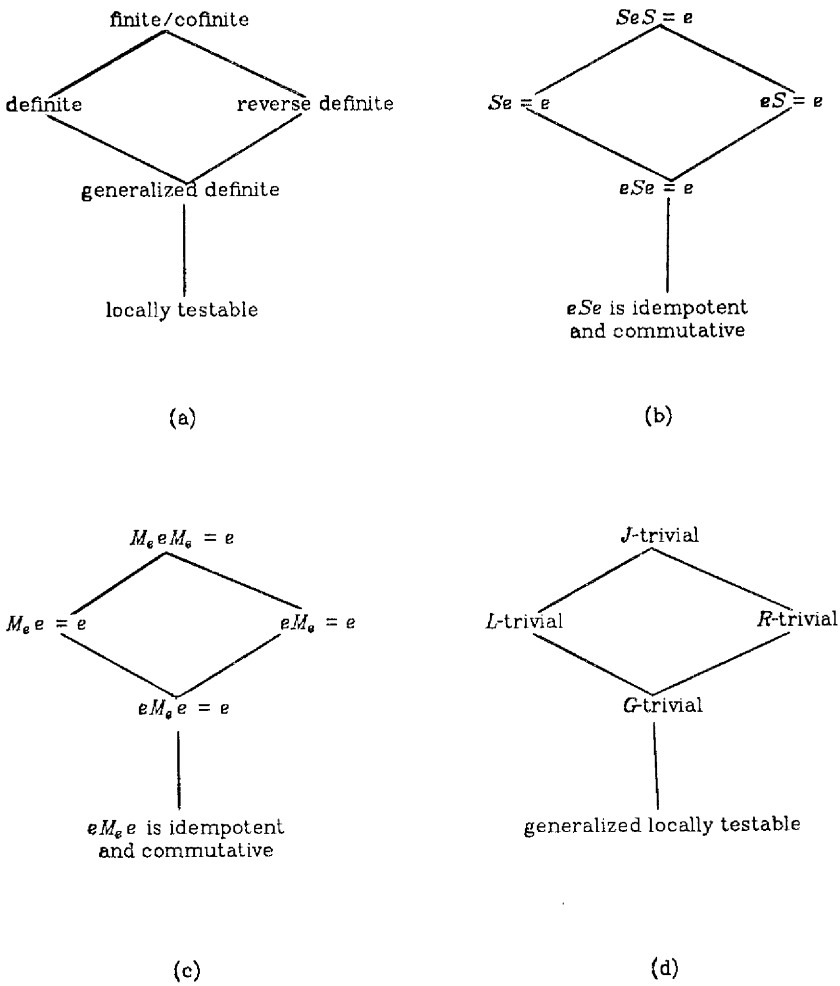
finite/cofinite
definite
reverse definite
generalized definite
locally testable
(a)
SeS = e
Se = e
eS = e
eSe = e
eSe is idempotent and commutative
(b)
$M_e e M_e = e$
$M_e e = e$
$e M_e = e$
$e M_e e = e$
$e M_e e$ is idempotent and commutative
(c)
J-trivial
L-trivial
R-trivial
G-trivial
generalized locally testable
(d)

Figure 1

2. The Basic Congruences

In this section we define two families of congruences that turn out to characterize generalized locally testable languages over a two letter alphabet Σ. The cardinality of Σ will be denoted by $\#\Sigma$. For $x \in \Sigma^*$, $x\alpha$ will denote the set of letters appearing in x. Note that $x\alpha = xm_1$.

Any word $w \in \Sigma^+$ can be written as $w_1 w_2 \cdots w_l$ where $l \geq 1$, $w_i = a_i^{n_i}$, $a_i \in \Sigma$, $n_i \geq 1$ for $1 \leq i \leq l$, and $a_i \neq a_{i+1}$ for $1 \leq i \leq l-1$. This representation will be referred to as the *run form* of w. The *run length*, $\|w\|$, of w is simply the value l, giving the number of factors in the run form of w. By convention, we define the run form of 1 to be 1 and $\|1\| = 0$.

Now let $r \geq 1$ and suppose $w \in \Sigma^*$ has run form $w_1 \cdots w_l$. The *front of run length* r of w, $w\hat{f}_r$, is defined by:

$$w\hat{f}_r = \begin{cases} w & \text{if } \|w\| \leq r \\ w_1 \cdots w_r & \text{if } \|w\| > r \end{cases}.$$

Analogously, the *tail of run length* r of w is:

$$w\hat{t}_r = \begin{cases} w & \text{if } \|w\| \leq r \\ w_{l-r+1} \cdots w_l & \text{if } \|w\| > r \end{cases}.$$

By convention, $w\hat{f}_0 = w\hat{t}_0 = 1$ for all $w \in \Sigma^*$.

If $x_1, \ldots, x_k \in \Sigma^*$ and $w = x_1 \cdots x_k$, then $(x_1, \ldots, x_k)$ is a *run partition* of w provided $\|x_1\| + \cdots + \|x_k\| = \|w\|$. Equivalently, this says that $x_i t_1 \neq x_{i+1} f_1$ for $1 \leq i < k$. If (x, y) is a run partition of w

then x is said to be a *run prefix* of w and y is said to be a *run suffix* of w. Note that x is a run prefix of w if and only if $x = w\hat{f}_r$ for some $r \geq 0$ and x is a run suffix of w if and only if $x = w\hat{t}_r$ for some $r \geq 0$.

Finally, we define

$$w\hat{m}_r = \{ y \mid \|y\| = r \text{ and } w = uyv \text{ for some } u,v \in \Sigma^* \}$$

to be the *set of all subsegments of w of run length r*, where $r \geq 0$. Note that $w\hat{m}_0 = \{1\}$ for all $w \in \Sigma^*$. It is clear that if x is a subsegment of w then $x\hat{m}_r \subseteq w\hat{m}_r$. Another consequence of this definition is the following result.

Proposition 1 . If $\|x\| \geq r + 1$ then $(uxv)\hat{m}_r = (ux)\hat{m}_r \cup (xv)\hat{m}_r$ for all $x,u,v \in \Sigma^*$.

Counting letters up to a threshold is an important concept in what follows. Its use is formalized in this definition.

Definition 2 . Let $h \geq 1$ and suppose $w,w' \in \Sigma^*$ have run forms $w_1 \cdots w_l$ and $w'_1 \cdots w'_{l'}$ respectively. Then $w \; \Theta_h \; w'$ if and only if $\|w\| = \|w'\|$ and, for $i = 1, \ldots, l$, $w_i\alpha = w'_i\alpha$ and either $w_i = w'_i$ or $|w_i|, |w'_i| \geq h$.

This is just another way of saying that Θ_h is the smallest congruence such that $a^h \; \Theta_h \; a^{h+1}$ for all $a \in \Sigma$. Two sets $S,S' \subseteq \Sigma^*$ are congruent with respect to Θ_h if for each $w \in S$ there exists a $w' \in S'$ such that $w \; \Theta_h \; w'$ and vice versa. Note that $w \; \Theta_{h+1} \; w'$ implies $w \; \Theta_h \; w'$ for all $h \geq 1$.

Definition 3 . Let $h, r \geq 1$. Then $w \hat{\sim}_{r,h} w'$ if and only if $w\hat{f}_r \; \Theta_h \; w'\hat{f}_r$, $w\hat{t}_r \; \Theta_h \; w'\hat{t}_r$, and $w\hat{m}_r \; \Theta_h \; w'\hat{m}_r$.

The following fact is easily verified.

Proposition 4 . If $w, w' \in \Sigma^*$ then $w \hat{\sim}_{r+1,h} w'$ implies $w \hat{\sim}_{r,h} w'$, and $w \hat{\sim}_{r,h+1} w'$ implies $w \hat{\sim}_{r,h} w'$.

If $\|w\| \leq r$ then $w\hat{f}_r = w = w\hat{t}_r$. If $\|w\| = r$ and $w\hat{m}_r \; \Theta_h \; w'\hat{m}_r$ then $\|w'\| = r$. Together, these two implications yield the next result.

Proposition 5 . For all $w, w' \in \Sigma^*$ and $r, h \geq 1$, $\|w\| \leq r$ and $w \hat{\sim}_{r,h} w'$ imply $w \; \Theta_h \; w'$. Also $w \; \Theta_h \; w'$ implies $w \hat{\sim}_{r,h} w'$.

Thus there is a very close relationship between Θ_h and $\hat{\sim}_{r,h}$ especially for words of short run length.

Theorem 6 . $\hat{\sim}_{r,h}$ is a congruence of finite index.

Proof: Let $r, h \geq 1$ and let $w, w' \in \Sigma^*$ be such that $w \hat{\sim}_{r,h} w'$. We claim that $aw \hat{\sim}_{r,h} aw'$ for all $a \in \Sigma^*$.

If $\|w\| \leq r$ or $\|w'\| \leq r$ then, since Θ_h is a congruence, the claim follows from Proposition 5.

So suppose $\|w\|, \|w'\| \geq r + 1$. Let $w = w_1 \cdots w_l$ and $w' = w'_1 \cdots w'_{l'}$ be the run forms of w and w' respectively. We have $l, l' \geq r + 1$, $w\hat{f}_r = w_1 \cdots w_r$, $w\hat{t}_r = w_{l-r+1} \cdots w_l$, $w'\hat{f}_r = w'_1 \cdots w'_r$, and $w'\hat{t}_r = w'_{l'-r+1} \cdots w'_{l'}$. Clearly $(aw)\hat{t}_r = w\hat{t}_r \; \Theta_h \; w'\hat{t}_r = (aw')\hat{t}_r$.

Also if $\{a\} \neq w_1\alpha$ then $(aw)\hat{f}_r = aw_1 \cdots w_{r-1} \ \Theta_h \ aw'_1 \cdots w'_{r-1} = (aw')\hat{f}_r$. If $\{a\} = w_1\alpha$ we have $(aw)\hat{f}_r = aw_1 \cdots w_r \ \Theta_h \ aw'_1 \cdots w'_r = (aw')\hat{f}_r$. It follows that $(aw)\hat{f}_r \ \Theta_h \ (aw')\hat{f}_r$.

Now consider $x \in (aw)\hat{m}_r$. If $x \in w\hat{m}_r$ then there exists $x' \in w'\hat{m}_r$ such that $x' \ \Theta_h \ x$. Hence assume $x \not\in w\hat{m}_r$. First suppose $\{a\} \neq w_1\alpha$. If $r = 1$ then $x = a \in (aw')\hat{m}_r$. Otherwise $x = aw_1 \cdots w_{r-2}u$ where u is a nonempty prefix of w_{r-1}. Because $w_{r-1} \ \Theta_h \ w'_{r-1}$ there must exist a nonempty prefix u' of w'_{r-1} such that $x \ \Theta_h \ aw'_1 \cdots w'_{r-2}u' \in (aw')\hat{m}_r$. On the other hand, suppose $\{a\} = w_1\alpha$. Then $x = aw_1 \cdots w_{r-1}u$, where u is a nonempty prefix of w_r. Since $w_r \ \Theta_h \ w'_r$ there exists a prefix u' of w'_r such that $x \ \Theta_h \ aw'_1 \cdots w'_{r-1}u' \in (aw')\hat{m}_r$. Hence for all $x \in (aw)\hat{m}_r$ there exists $x' \in (aw')\hat{m}_r$ such that $x \ \Theta_h \ x'$. Similarly, for all $x' \in (aw')\hat{m}_r$ there exists $x \in (aw)\hat{m}_r$ such that $x \ \Theta_h \ x'$. Thus $(aw)\hat{m}_r \ \Theta_h \ (aw')\hat{m}_r$.

The fact that $wa \ \hat{\sim}_{r,h} \ w'a$ follows by symmetry. Hence $\hat{\sim}_{r,h}$ is a congruence.

Finally, there are only a finite number of different Θ_h classes of segments of run length r and there are only a finite number of different Θ_h classes of fronts and tails, so that $\hat{\sim}_{r,h}$ is of finite index. []

Two additional facts are straightforward consequences of the definition of $\hat{\sim}_{r,h}$ and Proposition 5.

Proposition 7 . If $a \in \Sigma$ and $r, h \geq 1$, then $a^h \hat{\approx}_{r,h} a^{h+1}$.

Proposition 8 . If $x \in \Sigma^*$, $r, h \geq 1$, $\|x\| \geq 2$, and $s = (\|x\| - 1)r$, then $x^{r+1} \hat{\approx}_{s,h} x^{r+2}$.

Consider $x = a_1 \cdots a_n a_1$ where $a_1, \ldots, a_n$ are distinct letters in Σ and let $h \geq 2$. Since $a_1 x^r a_1 \in x^{r+2}\hat{m}_{s+1}$ but $a_1 x^r a_1 \not\Theta_h y$ for all $y \in x^{r+1}\hat{m}_{s+1}$, it is not true that $x^{r+1} \hat{\approx}_{s+1,h} x^{r+2}$. Therefore Proposition 8 cannot be improved, except in special cases.

Lemma 9 . Let $x, u, v \in \Sigma^*$, $r, h \geq 1$, and $\|x\| \geq r + 1$. Then $xu \ \Theta_h \ vx$ implies $xu \hat{\approx}_{r,h} xu^2$.

Proof: Since $\|x\| \geq r + 1$, $(xu)\hat{f}_r = x\hat{f}_r = (xu^2)\hat{f}_r$. Because $xu \ \Theta_h \ vx$ we also have $(xu)\hat{t}_r \ \Theta_h \ (vx)\hat{t}_r = x\hat{t}_r$. Next notice that $xu^2 \ \Theta_h \ vxu \ \Theta_h \ v^2x$ which implies $(xu^2)\hat{t}_r \ \Theta_h \ (v^2x)\hat{t}_r = x\hat{t}_r$. Hence $(xu)\hat{t}_r \ \Theta_h \ (xu^2)\hat{t}_r$. From Definition 3 and Proposition 5, $(xu^2)\hat{m}_r \ \Theta_h \ (vxu)\hat{m}_r$. Finally, by Proposition 1, $(vxu)\hat{m}_r = (vx)\hat{m}_r \cup (xu)\hat{m}_r \ \Theta_h \ (xu)\hat{m}_r$. Thus $(xu^2)\hat{m}_r \ \Theta_h \ (xu)\hat{m}_r$. []

Lemma 10 . Let $x, u, v \in \Sigma^*$, $r, h \geq 1$, and $\|x\| \geq r + 1$. Then $xuxvx \hat{\approx}_{r,h} xvxux$.

Proof: Clearly $(xuxvx)\hat{f}_r = x\hat{f}_r = (xvxux)\hat{f}_r$ and $(xuxvx)\hat{t}_r = x\hat{t}_r = (xvxux)\hat{t}_r$. Also, by Proposition 1, $(xuxvx)\hat{m}_r = (xux)\hat{m}_r \cup (xvx)\hat{m}_r = (xvxux)\hat{m}_r$. []

These last two lemmas motivate the following definition.

Definition 11 . Let $r,h \geq 1$ and let $\hat{\approx}_{r,h}$ be the smallest congruence on Σ^* satisfying:

(a) if $u \ \Theta_h \ v$ then $u \hat{\approx}_{r,h} v$,

(b) if $\|x\| \geq r+1$ and $xu \ \Theta_h \ vx$ then $xu \hat{\approx}_{r,h} xu^2$,

and (c) if $\|x\| \geq r+1$ then $xuxvx \hat{\approx}_{r,h} xvxux$.

Proposition 12 . For all $w, w' \in \Sigma^*$ and $r, h \geq 1$, $w \hat{\approx}_{r,h} w'$ implies $w \hat{\sim}_{r,h} w'$.

Proof: This follows from Proposition 5, Lemma 9, and Lemma 10. Any time we use one of the substitutions of Definition 11, we preserve the congruence $\hat{\sim}_{r,h}$. □

A (somewhat modified) converse of Proposition 12 also holds but the proof is considerably more involved as we will show.

3. A Structural Decomposition of Semiautomata

The semiautomata of generalized locally testable languages are considered in this section. Specifically, we show that every such semiautomaton can be covered by a cascade connection of a semiautomaton of an L-trivial language with an idempotent and commutative semiautomaton. It is convenient to phrase the proof in terms of congruences. The results of this section will also yield a converse of Proposition 12.

A *(finite) automaton* is a 4-tuple (Σ, Q, q_0, F), where Σ is a finite, non-empty alphabet, Q is a finite, non-empty set of states, $q_0 \in Q$ is the initial state, and $F \subseteq Q$ is the set of final states. The letters of Σ are viewed as functions from Q into Q. Concatenation of letters corresponds to functional composition [E]. An automaton is ***reduced*** if, for all $q, q' \in Q$, there exists $x \in \Sigma^*$ such that $qx \in F$ if and only if $q'x \not\in F$. An *(initialized) semiautomaton* is the triple (Σ, Q, q_0).

Definition 13 . Define the following equivalence relation on Σ^* for each $r, h \geq 1$. For $w, w' \in \Sigma^*$, $w \; \lambda_{r,h} \; w'$ if and only if

(a) $w \; \Theta_h \; w'$,

or (b) $\|w\| \geq r + 1$, $\|w'\| \geq r + 1$, and $w\hat{t}_r \; \Theta_h \; w'\hat{t}_r$.

Note that $w \; \lambda_{r,h} \; w'$ always implies $w\hat{t}_r \; \Theta_h \; w'\hat{t}_r$. However the converse is false; e.g. let $r = h = 2$, $w = ba$, and $w' = aba$. One easily verifies that $\lambda_{r,h}$ is a congruence relation of finite index on Σ^*. Let

$[x]_{r,h}$ denote the congruence class of $\lambda_{r,h}$ containing x.

We now define the *free* $\lambda_{r,h}$ *semiautomaton* that corresponds in the natural way to the congruence above. The semiautomaton is $(\Sigma, \{[x]_{r,h} \mid x \in \Sigma^*\}, [1]_{r,h})$, where $[x]_{r,h}a = [xa]_{r,h}$.

For the case of a two-letter alphabet, the following result holds. Let L be a regular language; then L is L-trivial if and only if there exist $r,h \geq 1$ such that L is a $\lambda_{r,h}$ language [B]. For the reader familiar with locally testable languages [BS], we point out that the role played by free definite semiautomata in that theory is played by free $\lambda_{r,h}$ semiautomata here.

It is also convenient to represent the free $\lambda_{r,h}$ semiautomaton by a directed graph $G_{r,h}$ defined as follows. The vertices of $G_{r,h}$ are the congruence classes $[x]_{r,h}$. There is an edge from $[x]_{r,h}$ to $[y]_{r,h}$ if and only if there exists $a \in \Sigma$ such that $xa \; \lambda_{r,h} \; y$; the edge is labelled by the pair $([x]_{r,h}, a)$. Clearly each edge in $G_{r,h}$ is uniquely identified by its label. Let $\Gamma_{r,h} = \{ ([x]_{r,h}, a) \mid x \in \Sigma^*, a \in \Sigma \}$ be the set of all the labels. This set of labels forms a new alphabet and paths in $G_{r,h}$ correspond to words in $\Gamma_{r,h}^+$. However, not all such words correspond to paths. Let $\Pi_{r,h}$ be the set of words corresponding to paths.

Define a mapping $\varphi : \Sigma^+ \to \Pi_{r,h} \subseteq \Gamma_{r,h}^+$ as follows:

$$a\varphi = ([1]_{r,h}, a) \text{ for } a \in \Sigma$$

$$(wa)\varphi = (w\varphi)([w]_{r,h}, a) \qquad \text{for} \qquad w \in \Sigma^+,$$

$a \in \Sigma$.

To decongest the notation we will denote $w\varphi$ by W, and letters in $\Gamma_{r,h}$ will be denoted by A, B, C, etc.

Proposition 14 . The mapping $\varphi : \Sigma^+ \to \Pi_{r,h}$ is one-to-one. The image of Σ^+ under φ is the subset of $\Pi_{r,h}$ corresponding to paths beginning at $[1]_{r,h}$.

Proof: Clearly each word in Σ^n corresponds to a unique path in $G_{r,h}$ of length n starting at $[1]_{r,h}$ and vice versa. []

Note that if $W = ([x_1]_{r,h}, a_1) \cdots ([x_n]_{r,h}, a_n)$ is a path in $\Pi_{r,h}$ then $W\varphi^{-1} = a_1 \cdots a_n$. Whenever possible $W\varphi^{-1}$ will be denoted by w. For convenience, we let $w = 1$ when $W = 1$.

Next we define the congruence $\sim$ on Σ^* to be the smallest congruence satisfying

$$x^2 \sim x \text{ and } xy \sim yx$$

for all $x,y \in \Sigma^*$. As above, we define a semiautomaton corresponding to this congruence, namely the *free idempotent and commutative semiautomaton* over Σ, $(\Sigma, \{[x]_\sim \mid x \in \Sigma^*\}, [1]_\sim)$, where $[x]_\sim a = [xa]_\sim$. This is equivalent to a semiautomaton (Σ, Q, q_0) which is free except for the conditions $qx = qx^2$ and $qxy = qyx$ for all $q \in Q$, $x,y \in \Sigma^*$ [BS]. One can verify [BS] that this is also equivalent to the semiautomaton $(\Sigma, \{Q \mid Q \subseteq \Sigma\}, \phi)$, where $Qa = Q \cup \{a\}$ for all $Q \subseteq \Sigma$.

Finally, we define the cascade connection of the free $\lambda_{r,h}$ semiautomaton over Σ with the free idempotent and commutative semiautomaton over $\Gamma_{r,h}$. This semiautomaton will be called the ***cascade semiautomaton*** and is (Σ, P, p_0), where $P = \{([x]_{r,h}, X\alpha) \mid x \in \Sigma^*\}$, $p_0 = ([1]_{r,h}, \phi)$, and $([x]_{r,h}, X\alpha)a = ([xa]_{r,h}, X\alpha \cup \{([x]_{r,h}, a)\})$. An informal representation of these ideas is shown in Figure 2.

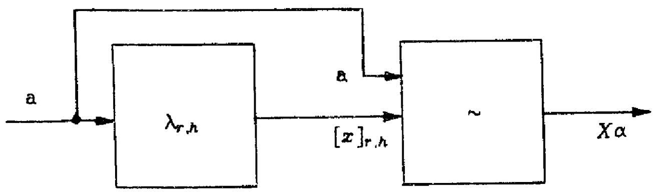

Figure 2. Cascade Connection.

Suppose x has been previously applied to the cascade semiautomaton and the present input letter is a. The front semiautomaton is in state $[x]_{r,h}$ and will move to state $[xa]_{r,h}$. The present input to the tail machine is the pair $([x]_{r,h}, a)$. The tail machine is in state $X\alpha$ where $X = x\varphi$, and will move to the state $X\alpha \cup \{([x]_{r,h}, a)\}$. Observe that in the cascade semiautomaton

$$p_0 x = p_0 y \text{ if and only if } [x]_{r,h} = [y]_{r,h} \text{ and } X\alpha = Y\alpha. \quad (2)$$

We now prove that any language accepted by (Σ, P, p_0) is a $\hat{\approx}_{r+2,h+1}$ language. It is sufficient to show that $x \hat{\approx}_{r+2,h+1} y$ implies

$[x]_{r,h} = [y]_{r,h}$ and $X\alpha = Y\alpha$. The fact that $x \hat{\approx}_{r+2,h+1} y$ implies $[x]_{r,h} = [y]_{r,h}$ follows from Definition 3, Propositions 4 and 5, and Definition 13.

Proposition 15 . Let $w, w' \in \Sigma^+$ and $r, h \geq 1$. Then

$$w \hat{\approx}_{r+2,h+1} w' \text{ implies } W\alpha = W'\alpha,$$

i.e. W and W' traverse the same set of edges in $G_{r,h}$.

Proof: Suppose $W = UAV$, $U, V \in \Gamma^*_{r,h}$, and $A \in \Gamma_{r,h}$.

If $U = 1$ then $A = ([1]_{r,h}, a)$ where $a = wf_1$. Since $w\hat{f}_{r+2}\, \Theta_{h+1}\, w'\hat{f}_{r+2}$, we have $w'f_1 = a$ and $A \in W'\alpha$.

So assume $U \neq 1$. Then $\|u\| > 0$. (Remember $u = U\varphi^{-1}$.) Let the run form of u be $u_1 \cdots u_m$ where $u_m = b^i$ for some $b \in \Sigma$ and $i > 0$. We consider the cases $b \neq a$ and $b = a$ separately.

1) $b \neq a$.

If $\|ua\| \leq r+2$ then ua is a prefix of $w\hat{f}_{r+2}$. But $w\hat{f}_{r+2}\, \Theta_{h+1}\, w'\hat{f}_{r+2}$ so there is a prefix $u'a$ of w' such that $u\, \Theta_h\, u'$. Thus $u\, \lambda_{r,h}\, u'$ and $A = ([u']_{r,h}, a) \in W'\alpha$.

Otherwise $\|ua\| > r+2$ and $u_{m-r} \cdots u_m a \in w\hat{m}_{r+2}$. Since $w\hat{m}_{r+2}\, \Theta_{h+1}\, w'\hat{m}_{r+2}$, it follows that $w' = x'y'az'$ where $u_{m-r} \cdots u_m a$ Θ_{h+1} $y'a$ and $y't_1 \neq a$. Now $\|y'\| > r$; therefore $(x'y')\hat{t}_r = y'\hat{t}_r$ Θ_{h+1} $u_{m-r+1} \cdots u_m = u\hat{t}_r$. Hence $u\, \lambda_{r,h}\, x'y'$ and $A = ([x'y']_{r,h}, a) \in W'\alpha$.

2) $b = a$.

If $V \neq 1$ let $j \geq 0$ and $z \in \Sigma^*$ be such that $v = a^j z$ and $zf_1 \neq a$. If $V = 1$ let $j = 0$ and $z = 1$. Then $w = u_1 \cdots u_{m-1} a^i a a^j z$.

If $\|ua\| \leq r + 2$ then ua is a prefix of $w\hat{f}_{r+2}$. Since $w\hat{f}_{r+2} \; \Theta_{h+1} \; w'\hat{f}_{r+2}$, there is a run partition (u', a^k, z') of w' such that $u_1 \cdots u_{m-1} a^{i+1+j} \; \Theta_{h+1} \; u'a^k$.

On the other hand, if $\|ua\| > r + 2$ then $u_{m-r-1} u_{m-r} \cdots u_{m-1} a^{i+1+j} \in w\hat{m}_{r+2} \; \Theta_{h+1} \; w'\hat{m}_{r+2}$. Thus we can write $w' = x'u'a^k z'$ where $u_{m-r-1} \cdots u_{m-1} a^{i+1+j} \; \Theta_{h+1} \; u'a^k$ and $u't_1 \neq a$.

In both cases either $k = i + 1 + j$ or $k, i + 1 + j \geq h + 1$. If $i < h$ let $i' = i$ and if $i \geq h$ let $i' = h$. Then $a^i \; \Theta_h \; a^{i'}$ and $a^k = a^{i'} a a^{j'}$ for some $j' \geq 0$. Hence $u \; \lambda_{r,h} \; u'a^{i'}$ and $A = ([u'a^{i'}]_{r,h}, a) \in W'\alpha$. []

We have now proved the following:

Proposition 16 . Any language accepted by the cascade connection of a free $\lambda_{r,h}$ semiautomaton with a free idempotent and commutative semiautomaton is a $\hat{\approx}_{r+2,h+1}$ language.

Using Proposition 15, we see that if $w \; \hat{\approx}_{r+2,h+1} \; w'$ then W and W' are coterminal paths in $G_{r,h}$ that contain the same set of edges. Therefore we can apply the following theorem on graphs. For further details see [E, page 224].

Theorem 17 . Let $\sim$ be the smallest congruence relation on $\Pi_{r,h}$ satisfying

$$XX \sim X \text{ and } XY \sim YX$$

for any two loops X and Y about the same vertex. Then for any two coterminal paths W and W', the conditions $W \sim W'$ and $W\alpha = W'\alpha$ are equivalent.

We now have the conclusion that for all $w, w' \in \Sigma^+$

$$w \mathrel{\hat{\approx}}_{r+2,h+1} w' \text{ implies } W \sim W'.$$

We will complete the proof of a converse of Proposition 12 with the aid of the next result.

Proposition 18 . Let $r \geq 2$, let $h \geq 1$, and suppose W and W' are coterminal paths in $G_{r,h}$ beginning at $[1]_{r,h}$. Then $W \sim W'$ implies $w \mathrel{\hat{\approx}}_{r-1,h} w'$.

Proof: It suffices to verify the claim in two cases. Here U and U' are any two loops about the same vertex. Note that $\|u\|, \|u'\| \geq 1$.

(1) $W = YUZ$ and $W' = YU^2Z$.

Since U is a loop, Y and YU are coterminal paths in $G_{r,h}$. Thus $y \, \lambda_{r,h} \, yu$.

Consider $\|u\| = 1$. Then $u = a^i$ where $a \in \Sigma$ and $i > 0$. Now $y\hat{t}_r \; \Theta_h \; (yu)\hat{t}_r$ implies $yt_1 = (yu)t_1 = a$. Therefore y can be written as $y = xa^j$ where $j > 0$ and $xt_1 \neq a$. Since $yu = xa^j a^i$ and

$y\hat{t}_r \ \Theta_h \ (yu)\hat{t}_r$, we have $a^j \ \Theta_h \ a^{i+j}$. Hence $w = yuz = xa^j uz \ \ \Theta_h \ \ xa^{j+i}uz = yu^2z = w'$. By Definition 11 (a), $w \ \hat{\approx}_{r-1,h} \ w'$.

If $\|u\| \geq 2$ then clearly $\|y\| < \|yu\|$. Since $y \ \lambda_{r,h} \ yu$, we must have $\|y\| > r$ and $y\hat{t}_r \ \Theta_h \ (yu)\hat{t}_r$. Let $x = y\hat{t}_r$ and let $y = y_1 x$. Then $(xu)\hat{t}_r = (yu)\hat{t}_r \ \Theta_h \ y\hat{t}_r = x$ and thus $xu \ \Theta_h \ vx$ for some $v \in \Sigma^*$. By Definition 11 (b), $xu \ \hat{\approx}_{r-1,h} \ xu^2$. Therefore $w = yuz = y_1 xuz \ \ \hat{\approx}_{r-1,h} \ \ y_1 xu^2 z = yu^2 z = w'$ as required.

(2) $W = YUU'Z$ and $W' = YU'UZ$.

In $G_{r,h}$ the paths Y, YU, YU', YUU' and $YU'U$ are all coterminal; therefore $y \ \ \lambda_{r,h} \ \ yu \ \ \lambda_{r,h} \ \ yu' \ \lambda_{r,h} \ \ yuu' \ \ \lambda_{r,h} \ \ yu'u$.

If $\|u\| = 1$ then $u = a^i$ where $a \in \Sigma$ and $i > 0$. Applying the argument of (1) to both y and yu' gives us $y \ \Theta_h \ yu$ and $yu' \ \Theta_h \ yu'u$. Thus $w = yuu'z \ \ \Theta_h \ \ yu'z \ \ \Theta_h \ \ yu'uz = w'$. By Definition 11(a), $w \ \hat{\approx}_{r-1,h} \ w'$. The case $\|u'\| = 1$ is similar.

Finally consider $\|u\|, \|u'\| \geq 2$. As in (1), $\|y\| > r$, we let $x = y\hat{t}_r$ and $y = y_1 x$, and it follows that $(xu)\hat{t}_r \ \Theta_h \ x$ and $xu \ \Theta_h \ vx$ for some $v \in \Sigma^*$. Then Definition 11(b) implies $xu \ \hat{\approx}_{r-1,h} \ xu^2$. By induction, we get $xu^n \ \Theta_h \ v^{n-1}xu$. Since $\hat{\approx}_{r-1,h}$ is a congruence we have $xu \ \hat{\approx}_{r-1,h} \ xu^n$ for all $n \geq 1$.

Choose n so that $\|u^n\| > \|x\|$; this can always be done since $\|u\| \geq 2$. Then $u^n\hat{t}_r = (xu^n)\hat{t}_r \ \Theta_h \ (v^{n-1}xu)\hat{t}_r \ \Theta_h \ x$ and $u^n \ \Theta_h \ sx$

for some $s \in \Sigma^*$. By Definition 11(a) it follows that $xu^n \hat{\approx}_{r-1,h} xsx$. Therefore $xu \hat{\approx}_{r-1,h} xsx$. Similarly $xu' \hat{\approx}_{r-1,h} xs'x$ for some $s' \in \Sigma^*$. Then $xuu' \hat{\approx}_{r-1,h} xsxu' \hat{\approx}_{r-1,h} xsxs'x$. By Definition 11(c), $xsxs'x \hat{\approx}_{r-1,h} xs'xsx$. Hence $w = yuu'z = y_1xuu'z \hat{\approx}_{r-1,h} y_1(xsxs'x)z \hat{\approx}_{r-1,h} y_1(xs'xsx)z \hat{\approx}_{r-1,h} y_1xu'uz = yu'uz = w'$. []

Proposition 19 . Let $r \geq 2$, $h \geq 1$, and $w, w' \in \Sigma^*$. Then $w \hat{\sim}_{r+2,h+1} w'$ implies $w \hat{\approx}_{r-1,h} w'$.

Proof: If $w = 1$ or $w' = 1$ then Proposition 5 implies $w = w'$ and hence $w \hat{\approx}_{r-1,h} w'$. Otherwise, by Proposition 15, $W\alpha = W'\alpha$ in $G_{r,h}$. Theorem 17 implies $W \sim W'$, i.e. W can be obtained from W' by using only the transformations of the type $X^2 \sim X$ and $XY \sim YX$ on loops in $G_{r,h}$. From Proposition 18 we obtain $w \hat{\approx}_{r-1,h} w'$. []

The results of this section can be summarized by the following theorem.

Theorem 20 . Let $L \subseteq \Sigma^*$. The following are equivalent.

(a) L is an $\hat{\sim}_{r,h}$ language for some $r, h \geq 1$.

(b) L is an $\hat{\approx}_{r,h}$ language for some $r, h \geq 1$.

(c) The reduced automaton for L is covered by the cascade connection of a free $\lambda_{r,h}$ automaton with a free idempotent and commutative semiautomaton.

Proof: By Proposition 12 and Proposition 19, we have the equivalence of (a) and (b). Using Proposition 16, we verify that if a semiautomaton is covered by the cascade semiautomaton, then any language accepted by that semiautomaton is a $\hat{\sim}_{r+2,h+1}$ language. Thus (a) implies (c).

Finally, we will show that (c) implies (b). More specifically we will show that the semiautomaton (Σ, Q, q_0) of the reduced automaton of a $\hat{\approx}_{r-1,h}$ language is covered by the cascade connection (Σ, P, p_0) of a free $\lambda_{r,h}$ automaton with a free idempotent and commutative semiautomaton. It suffices to verify that $p_0x = p_0y$ implies $q_0x = q_0y$. If $p_0x = p_0y$ then it follows from (2) that X and Y are coterminal paths in $G_{r,h}$ with $X\alpha = Y\alpha$. By Theorem 17, we have $X \sim Y$. Now Proposition 18, implies $x \hat{\approx}_{r-1,h} y$. Since (Σ, Q, q_0) is the semiautomaton of the reduced automaton of a $\hat{\approx}_{r-1,h}$ language, we must have $q_0x = q_0y$. Hence (Σ, P, p_0) covers (Σ, Q, q_0). □

4. The Monoid Characterization

We are now in a position to relate the congruence characterizations with the monoid characterizations mentioned in Section 1. Two preliminary results are needed first.

Proposition 21 . Let M be a finite monoid. If $eM_e e$ is idempotent for all idempotents $e \in M$, then M is aperiodic.

Proof: Let $f \in M$. Since M is finite, there exists m such that $f^m = f^{2m}$. Note that $e = f^m$ is an idempotent and $f \in M_e$. Now $f^{m+1} = f^{2m+1} = efe$ *in* $eM_e e$. Since $eM_e e$ is idempotent, $f^{m+1} = efe = (efe)^2 = f^{2m+2} = f^{m+2}$. Thus M is aperiodic. ▯

Lemma 22 . Suppose $xu \;\Theta_h\; vx$, $u \neq 1$, (x,u) is a run partition, and y is a run prefix of x. Then there exist a run partition (z_1,y',z_2) of x and a run prefix u' of u, such that $y \;\Theta_h\; y'$, $u \;\Theta_h\; u'z_2$, and $xu' \;\Theta_h\; vz_1y$.

Proof: Let $x = x_1 \cdots x_r$ be the run form of x and let $s = \|y\|$ so that $y = x_1 \cdots x_s$. Furthermore, let $k = \max\{i \mid x_i \cdots x_{i+s-1} \;\Theta_h\; y\}$, let $y' = x_k \cdots x_{k+s-1}$, let $z_1 = x_1 \cdots x_{k-1}$, and let $z_2 = x_{k+s} \cdots x_r$. Then (z_1,y',z_2) is a run partition of x, $y \;\Theta_h\; y'$, and $z_1y'z_2u = xu \;\Theta_h\; vx = vz_1y'z_2$.

If $\|z_2\| \geq \|u\|$ then z_2 has a run partition (z_3,u') such that $u \;\Theta_h\; u'$ and $x \;\Theta_h\; vz_1y'z_3$. (It may be that $z_3 = 1$.) Note that $\|z_2\| > \|z_3\|$, since $u \neq 1$. If $j = \|vz_1\| + 1$ then $x_j \cdots x_{j+s-1} \;\Theta_h\; y' \;\Theta_h\; y$. But $j =$

$\|vz_1\| + 1 = \|vz_1y'z_3\| - \|y'\| - \|z_3\| + 1 = \|x\| - \|y'\| - \|z_3\| + 1 >$
$\|x\| - \|y'\| - \|z_2\| + 1 = \|z_1y'z_2\| - \|y'\| - \|z_2\| + 1 = \|z_1\| + 1 =$
k. This contradicts the definition of k. Therefore $\|z_2\| < \|u\|$.

It follows that u has a run partition (u', z'_2) such that $z_2 \, \Theta_h \, z'_2$. Then $u = u'z'_2 \;\Theta_h\; u'z_2$. Also, since $xu'z'_2 = xu \;\Theta_h\; vx = vz_1y'z_2$, the final relationship, $xu' \;\Theta_h\; vz_1y$, is true. []

The key to relating language properties with monoid properties is the concept of the syntactic monoid of a language. The *syntactic congruence* $\equiv_L$ of a language $L \subset \Sigma^*$ is defined as follows. For all $u, v, x, y \in \Sigma^*$

$$x \equiv_L y \text{ if and only if } (\, uxv \in L \text{ if and only if } uyv \in L \,).$$

The quotient monoid $M = \Sigma^* / \equiv_L$ is called the *syntactic monoid* of L. The *syntactic morphism* of L is the natural morphism mapping $x \in \Sigma^*$ onto the congruence class of $\equiv_L$ containing x. For conveniene, x is used to denote the congruence class of $\equiv_L$ containing x.

At this point, it is necessary to restrict the size of the alphabet Σ to two.

Theorem 23 . Suppose $\#\Sigma = 2$. Let $L \subseteq \Sigma^*$ and let M be its syntactic monoid. If M is finite and $eM_e e$ is idempotent and commutative for all $e^2 = e \in M$ then L is a $\hat{S}_{r,h}$ language for some $r, h \geq 1$.

Proof: Suppose M is finite and $eM_e e$ is idempotent and commutative for all $e^2 = e \in M$. By Proposition 21, M is aperiodic. Thus there exists an $h \geq 1$ such that $f^h = f^{h+1}$ for all $f \in M$.

Let $r = 2(\#M) - 1$. We have to show that $w \approx_{r,h} w'$ implies $\underline{w} = \underline{w'}$ in M. It is sufficient to verify that for all $u, v, x \in \Sigma^*$

$$\text{if } u \,\Theta_h\, v \text{ then } \underline{u} = \underline{v},$$

$$\text{if } \|x\| \geq r + 1 \text{ and } xu \,\Theta_h\, vx \text{ then } \underline{xu} = \underline{xvu},$$

$$\text{and if } \|x\| \geq r + 1 \text{ then } \underline{xuxvx} = \underline{xvxux}.$$

For all $w \in \Sigma^*$, $\underline{w} \in M$ and thus $\underline{w^h} = (\underline{w})^h = (\underline{w})^{h+1} = \underline{w^{h+1}}$. In particular this implies that $\underline{u} = \underline{v}$ for all u, v such that $u \,\Theta_h\, v$.

Assume $\|x\| \geq r + 1$ and $x = x_1 \cdots x_s$ is the run form of x. Let $x'_0 = 1$. and let $x'_i = x_{2i-1}x_{2i}$ for $1 \leq i \leq \#M$. Then the $\#M + 1$ elements of M:

$$\underline{x'_0}, \underline{x'_0x'_1}, \underline{x'_0x'_1x'_2}, \ldots, \underline{x'_0x'_1x'_2 \cdots x'_{\#M}}$$

cannot all be distinct. Hence there exist i, j such that $0 \leq i < j \leq \#M$ and

$$\underline{x'_0 \cdots x'_i} = \underline{x'_0 \cdots x'_i x'_{i+1} \cdots x'_j} = (\underline{x'_0 \cdots x'_i})\,(\underline{x'_{i+1} \cdots x'_j}).$$

Let $y_1 = x'_0 \cdots x'_i$, let $y_2 = x'_{i+1} \cdots x'_j$, and let $e = \underline{y_2^h}$. If $s = 2j + 1$ then let $y_3 = 1$; otherwise let $y_3 = x_{2j+1} \cdots x_s$. Then e is an idempotent, (y_1, y_2, y_3) is a run partition of x, and $\underline{y_1} = \underline{y_1y_2} = \underline{y_1y_2^h} = \underline{y_1}e$.

Now $\underline{z} \in M_{\underline{y}}$ if and only if $z \in (y\alpha)^*$. Since $\|y_2\| \geq 2$ and $\#\Sigma = 2$, it follows that $\Sigma = y_2\alpha = y_2^h\alpha$. Therefore $\underline{z} \in M_e$ for all $z \in \Sigma^*$.

In particular, $\underline{y_3uy_1}, \underline{y_3vy_1} \in M_e$. Since eM_ee is commutative, $\underline{xuxvx} = \underline{y_1}e\underline{y_3uy_1}e\underline{y_3vy_1}e\underline{y_3} = \underline{y_1}(e\underline{y_3uy_1}e)(e\underline{y_3vy_1}e)\underline{y_3} =$

$\underline{y_1}(e\underline{y_3vy_1}e)(e\underline{y_3uy_1}e)\underline{y_3} = \underline{y_1ey_3vy_1ey_3uy_1ey_3} = \underline{xvxux}$.

Finally suppose $xu \; \Theta_h \; vx$. If $u = 1$ there is nothing to prove. Therefore assume that $u \neq 1$. Let $u = u_1 \cdots u_k$ be the run form of u.

If $\|u\| = 1$ then $u = a^i$ for some $a \in \Sigma$ and $i > 0$. Now $x_1 \cdots x_{s-1}x_s a^i = xu \; \Theta_h \; vx = vx_1 \cdots x_{s-1}x_s$. Since $\|x_s\| = 1$ and $s \geq r + 1 = 2(\#M) > 1$, it follows that $x_s \; \Theta_h \; x_s a^i = x_s u$. Hence $x_s u \; \Theta_h \; x_s uu$ and $\underline{xu} = \underline{x_1 \cdots x_{s-1}}\,\underline{x_s u} = \underline{x_1 \cdots x_{s-1}}\,\underline{x_s uu} = \underline{xuu}$.

It remains to consider the case $\|u\| \geq 2$. If (x,u) is a run partition (i.e. $xt_1 \neq uf_1$) then by Lemma 22 there exists a run partition (z_1, y', z_2) of x and a run prefix u' of u such that $y_1 \; \Theta_h \; y'$, $u \; \Theta_h \; u'z_2$, and $xu' \; \Theta_h \; vz_1y_1$. Thus

$\underline{xu} = \underline{xu'z_2} = \underline{vz_1y_1z_2} = \underline{vz_1y_1}e\underline{z_2} = \underline{xu'}e\underline{z_2} = \underline{z_1y_1z_2u'}e\underline{z_2} = \underline{z_1y_1}e\underline{z_2u'}e\underline{z_2}$

and

$$\underline{xuu} = \underline{vxu} = \underline{vxu'z_2} = \underline{vxu'}e\underline{z_2} = \underline{xuu'}e\underline{z_2} = (\underline{z_1y_1}e\underline{z_2u'}e\underline{z_2})\,\underline{u'}e\underline{z_2}.$$

Because $\underline{z_2u'} \in M_e$ and eM_ee is idempotent, we have $e\underline{z_2u'}e\underline{z_2u'}e = e\underline{z_2u'}e$. Thus $\underline{xu} = \underline{xuu}$.

Otherwise $xt_1 = uf_1$. Let $z = x_1 \cdots x_{s-1}$ and let $w = x_s u_1 \cdots u_{k-1}$. Clearly y_1 is a run prefix of z. Since $k, s > 1$, $u_k = (xu)\hat{t}_1$ and $x_s = (vx)\hat{t}_1$. Then $xu \; \Theta_h \; vx$ implies $zw = xu_1 \cdots u_{k-1} \; \Theta_h \; vx_1 \cdots x_{s-1} = vz$ and $u_k \; \Theta_h \; x_s$. Also notice that $\|w\| \geq 2$. (This is because either $k > 2$ or $k = 2$ and $x_s\alpha = u_k\alpha \neq u_1\alpha$.) The argument in the previous paragraph is now applicable with z and

w replacing x and u, respectively. Therefore $\underline{xw} = \underline{xwu}$ and $\underline{xu} = \underline{xwu_k} = \underline{xwwu_k} = \underline{x}\,\underline{u_1 \cdots u_{k-1}}\,\underline{x_s u_1 \cdots u_{k-1}}\,\underline{u_k} = \underline{x}\,\underline{u_1 \cdots u_{k-1}}\,\underline{u_k}\,\underline{u_1 \cdots u_{k-1} u_k} = \underline{xuu}$. []

The converse is also true and holds for arbitrary alphabets Σ.

Theorem 24 . Let $L \subseteq \Sigma^*$ and let M be its syntactic monoid. If L is a $\hat{\approx}_{r,h}$ language for some $r,h \geq 1$ then M is finite and $eM_e e$ is idempotent and commutative for all $e^2 = e \in M$.

Proof: Suppose L is a $\hat{\approx}_{r,h}$ language. From Theorem 6 and Proposition 19 it follows that $\hat{\approx}_{r,h}$ is a congruence of finite index. Hence the syntactic monoid M is finite.

Let e be an idempotent element of M and let $f, g \in M_e$. Since M is a syntactic monoid, there exist $w \in \Sigma^*$ and $u, v \in (w\alpha)^*$ such that $e = \underline{w}$, $f = \underline{u}$, and $g = \underline{v}$.

If $\|w\| = 0$ then $w = u = v = 1$ and $e = f = g = 1$ so that $efe = (efe)^2$ and $(efe)(ege) = (ege)(efe)$.

If $\|w\| = 1$ then $w = a^i$, $u = a^j$, and $v = a^k$ for some $a \in \Sigma$, $i \geq 1$ and $j, k \geq 0$. Let $x = w^h$ so that $\underline{x} = (\underline{w})^h = e^h = e$. It follows that $xux = a^{2hi+j}\ \Theta_h\ a^{4hi+2j} = (xux)^2$ which implies $xux \hat{\approx}_{r,h} (xux)^2$. Thus $efe = (efe)^2$. Also note that $xuxxvx = xvxxux$ and hence $(efe)(ege) = (ege)(efe)$.

Finally, suppose $\|w\| > 1$. Let $x = w^r$ so that $\underline{x} = (\underline{w})^r = e^r = e$

and $\|x\| > r$. Since $x(ux) = (xu)x$, it follows from Definition 11 that $xux \mathrel{\hat{\approx}_{r,h}} xuxux$. Also $xuxvx \mathrel{\hat{\approx}_{r,h}} xvxux$. Thus $efe = \underline{xux} = \underline{xuxux} = efefe = (efe)(efe)$ and $(efe)(ege) = efege = \underline{xuxvx} = \underline{xvxux} = egefe = (ege)(efe)$. □

5. Conclusions

The congruence $\hat{\sim}_{r,h}$ of Definition 3 can be viewed as a "testing" congruence in the sense that, given x and y, it is easy to determine whether $x \hat{\sim}_{r,h} y$ by testing $\hat{f}_r$, $\hat{t}_r$, and $\hat{m}_r$. On the other hand, the congruence $\hat{\approx}_{r,h}$ is a "substitution" congruence in the sense that any word can be obtained from a congruent word by a series of suitable substitutions. However, it is not at all clear how to test whether $x \hat{\approx}_{r,h} y$, for given $x, y \in \Sigma^*$.

For the case $\#\Sigma > 2$, the problem of characterizing the languages whose syntactic monoids M satisfy the condition that $eM_e e$ is idempotent and commutative for all idempotents $e \in M$, is still open in the sense that no testing congruence is known. However, we have succeeded in generalizing $\hat{\approx}_{r,h}$ to obtain a substitution congruence that corresponds to the monoid property.

As a final remark, we point out that run length is a generalization of length. It is a suitable generalization for $\#\Sigma \leq 2$, but not otherwise. This problem of generalizing length appears to be of fundamental importance not only in finding a testing congruence as mentioned above, but also in the general study of star-free languages.

References

[B] Brzozowski, J.A., *A Generalization of Finiteness*, Semigroup Forum 13 (1977), pages 239-251.

[BF] Brzozowski, J.A., and Fich, F.E., *Languages of R-Trivial Monoids*, J. Computer and System Sciences, 20 (1980), pages 32-49.

[BS] Brzozowski, J.A., and Simon, I., *Characterizations of Locally Testable Events*, Discrete Mathematics 14 (1973), pages 243-271.

[E] Eilenberg, S., *Automata, Languages, and Machines*, Vol. B, Academic Press, New York, 1976.

[F] Fich, Faith E., *Languages of R-Trivial and Related Monoids*, M. Math. thesis, University of Waterloo, Canada, 1979.

[FB] Fich, Faith E., and Brzozowski, J.A., *A Characterization of a Dot-depth Two Analogue of Generalized Definite Languages*, Proc. 6th ICALP, Lecture Notes in Computer Science, Vol. 71, Springer-Verlag, Berlin, 1979, pages 230-244.

[M] McNaughton, R., *Algebraic Decision Procedures for Local Testability*, Math. Systems Theory 8 (1974), pages 60-76.

[S] Simon, I., *Hierarchies of Events with Dot-depth One*, Ph.D. thesis, University of Waterloo, Canada, 1972.