EACH REGULAR CODE IS INCLUDED
IN A MAXIMAL REGULAR CODE

by

A. Ehrenfeucht** and G. Rozenberg***

CU-CS-236-82 September, 1982

**University of Colorado, Department of Computer Science, Boulder, Colorado

*Institue of Applied Mathematics and Computer Science, University of Leiden, Leiden, The Netherlands and University of Colorado, Department of Computer Science, Boulder, Colorado
ANY OPINIONS, FINDINGS, AND CONCLUSIONS OR RECOMMENDATIONS EXPRESSED IN THIS PUBLICATION ARE THOSE OF THE AUTHOR AND DO NOT NECESSARILY REFLECT THE VIEWS OF THE NATIONAL SCIENCE FOUNDATION.
EACH REGULAR CODE IS INCLUDED
IN A MAXIMAL REGULAR CODE

by

A. Ehrenfeucht*

and

G. Rozenberg**

*Department of Computer Science, University of Colorado, Boulder, Colorado 80309

**Institute of Applied Mathematics and Computer Science, University of Leiden, Leiden, The Netherlands.

All correspondence to second author.
ABSTRACT

It is proved that each regular code is included in a maximal regular code. A corollary of this result settles an open question from \([R]\).
INTRODUCTION

A language \( C \subseteq \Sigma^+ \) is called a code if \( C^* \) is a free submonoid of \( \Sigma^+ \) with base \( C \). The theory of codes initiated by M. Schützenberger ([Sch]) forms an interesting fragment of formal language theory. A code \( C \subseteq \Sigma^+ \) is called maximal if, for any \( z \in \Sigma^*-C \), \( C \cup \{z\} \) is not a code. All codes are subsets of maximal codes and the investigation of maximal codes forms an active research area within the theory of codes (see, e.g., [BPS], [P1], [R], and [SM]). In particular one is often interested in the problem of the following kind: given a code \( C \) of type \( X \) (e.g. finite or regular) is it possible to find a maximal code \( D \) of type \( X \) such that \( C \subseteq D \)?

It was shown in [R] that for finite codes this question gets a negative answer. Since then the following question remained open: is every finite code included in a maximal regular code? Obviously any finite (resp. regular) prefix code is included in a finite (resp. regular) maximal prefix code. Recently it was shown in [P2] that every finite bifrefix code is included in a maximal bifrefix regular code.

In this paper we provide a positive answer to the above question. As a matter of fact we prove a more general result (Theorem 5): each regular code is included in a regular maximal code. We would like to emphasize the following: the new result presented in this paper is Theorem 5; most of the other results is in one form or the other (and perhaps in a different terminology) retrievable from the literature. However we have decided to make this paper rather self-contained and to provide all the needed results with their (sometimes different from the literature) proofs carried out in a "uniform manner".

We assume the reader to be familiar with basic formal language theory - in particular with rudimentary theory of regular languages (see, e.g., [S]).
PRELIMINARIES

We use mostly standard language theoretic notation and terminology.

For a set \( A \), \( \#A \) denotes the cardinality of \( A \).

For sets \( A, B \), \( A - B \) denotes the set theoretic difference of \( A \) and \( B \).

For a word \( z \), \( |z| \) denotes its length and \( \text{first}(x) \) denotes the first letter of \( x \); if \( x = x_1 \ldots x_n \) then \( y \) is called a subword of \( x \) (also referred to as a segment or a factor of \( x \)). The set of all subwords of \( x \) is denoted by \( \text{sub}(x) \) and for a language \( K \), \( \text{sub}(K) = \bigcup_{x \in K} \text{sub}(x) \).

A nonempty word \( x \) is called bordered if \( x = y z y \) for a nonempty word \( y \); otherwise \( x \) is called unbordered.

A language \( C \subseteq \Sigma^+ \) is called a code if every word \( y \in C^+ \) satisfies the following condition:

if \( y = u_1 \ldots u_n \) and \( y = x_1 \ldots x_m \) for \( n, m \geq 1 \) and \( u_1, \ldots, u_n, x_1, \ldots, x_m \in C \) then \( n = m \) and \( u_i = x_i \) for \( 1 \leq i \leq n \). (In other words, \( y \) has a unique representation in \( C \); subwords \( u_1, \ldots, u_n \) of this representation are referred to as \( C \)-blocks of \( y \)).

A code \( C \subseteq \Sigma^+ \) is called maximal if, for each \( x \in \Sigma^+ - C \), \( C \cup \{x\} \) is not a code.

In the sequel of this paper we consider an arbitrary but fixed alphabet \( \Sigma \) where \( \sigma = \#\Sigma > 1 \); all languages we will consider are over \( \Sigma \).

For a language \( K \) and a positive integer \( n \), \( L_n(K) = \{ w \in K : |w| = n \} \) and \( a_n(K) = \# L_n(K) \).

We will define now and recall a number of notions concerning languages - they will be central to our paper.

Let \( K \subseteq \Sigma^+ \).

(1) \( K \) is dense if \( x \in \text{sub}(K^*) \) for each \( x \in \Sigma^* \).

(2) \( K \) is fast if there exists a positive integer \( n \) such that for each \( w \in \text{sub}(K^*) \)
there exist $x, y \in \Sigma^*$ such that $|xy| \leq n$ and $x \cdot y \in K^*$.

(3) $K$ is rich if there exists a positive integer $e$ such that $\sigma_m(K^*) \geq \frac{\sigma^m}{e}$ for infinitely many positive integers $m$. 
RESULTS

In this section we investigate the problem how various properties of a code (such as: fast, dense, rich, regular and maximal) influence each other. Once this relationship is explored we can settle the problem of completing a regular code to a regular maximal code.

Our first result is known (see [SM]). However for the sake of completeness we provide its proof (which is different from the proof in [SM]).

**Theorem 1.** Each maximal code is dense.

*Proof.*

First we prove the following result.

**Claim 1.** Let $C$ be a code that is not dense. There exists an unbordered word $w_C$ such that $w_C \notin \text{sub} (C^*)$.

*Proof of Claim 1.*

Since $C$ is not dense, there exists a word $z \notin \text{sub} (C^*)$. Let $b \in \Sigma$ be such that $b \neq \text{first}(z)$ and let $w_C = z b^{|z|}$. Clearly $w_C$ is unbordered. Moreover $w_C \notin \text{sub} (C^*)$, because $z \notin \text{sub} (C^*)$.

Thus Claim 1 holds. ♦

Now we prove Theorem 1 as follows.

Let $C$ be a maximal code.

Assume to the contrary that $C$ is not dense. Then let $w_C$ be an unbordered word satisfying the statement of Claim 1.

Consider $D = C \cup \{w_C\}$. Let $y$ be an arbitrary word in $D^*$. Since $w_C$ is unbordered, $y$ has a unique representation of the form $y = x_0 w_C x_1 w_C \cdots w_C x_n$, where $n \geq 0$ (that is if $y = u_0 w_C u_1 w_C \cdots w_C u_m$...
where

\( m \geq 0 \) then \( m = n \) and \( u_i = z_i \) for \( 1 \leq i \leq n \). Since \( C \) is a code and \( w_C \not\in \text{sub} \ (C^*) \), \( y \) has a unique representation in \( D \). Thus \( D \) is a code.

Since \( C \subseteq D \) and \( w_C \not\in \text{sub} \ (C^*) \) we get a contradiction (to the fact that \( C \) is maximal).

Consequently \( C \) must be dense and Theorem 1 holds. \( \blacksquare \)

**Theorem 2.** Each rich code is maximal.

**Proof.**

Let \( C \) be a rich code and let \( \varepsilon \) be a positive integer constant satisfying the definition of richness for \( C \).

Assume to the contrary that \( C \) is not maximal. Let \( z \) be a word such that \( B = C \cup \{z\} \) is a code; let \( |z| = t \).

Let \( k \) be a positive integer. Let \( n_1, \ldots, n_k \) be a sequence of positive integers such that

\[
\sum_{i=1}^{k} n_i < \sum_{i=1}^{k+1} n_i \quad \text{and} \quad a_{n_i}(C^*) \geq \frac{\sigma_i}{\varepsilon}
\]

(1)

(Since \( C \) is rich and \( \varepsilon \) satisfies the definition of richness of \( C \), such a sequence exists).

Consider \( \tau = n_1 + n_2 + \cdots + n_k + kt \). Clearly

\[
a_{\tau}(B^*) \leq \sigma^r
\]

(2)

On the other hand let us consider an arbitrary permutation \( i_1, \ldots, i_k \) of the set \( \{1, \ldots, k\} \). Let \( y_{i_1} \in L_{n_1}(C^*), \ldots, y_{i_k} \in L_{n_k}(C^*) \) and let \( \gamma(i_1, \ldots, i_k) = y_{i_1} z y_{i_2} z \cdots y_{i_k} z \). Since \( B \) is a code, if \( (j_1, \ldots, j_k) \) is a permutation of \( \{1, \ldots, k\} \) different from \( (i_1, \ldots, i_k) \), then \( \gamma(i_1, \ldots, i_k) \neq \gamma(j_1, \ldots, j_k) \). Consequently from (1) it follows that
\[
\frac{\sigma^1}{e} \frac{\sigma^2}{e} \cdots \frac{\sigma^n}{e} k! \leq \alpha_r(B^*) \tag{3}
\]

From (2) and (3) it follows that
\[
k! \leq e^k \sigma^k = (e \sigma^k)^k \tag{4}
\]

Since \(e \sigma^k\) is a constant (independent of \(k\)), there exists a positive integer \(k_0\) such that, for all \(s > k_0\), \(s! > (e \sigma^k)^s\). Consequently (4) yields a contradiction (\(k\) was chosen to be an arbitrary positive integer).

Thus \(C\) must be maximal and Theorem 2 holds. □

**Theorem 3.** Each regular code is fast.

**Proof.**

Obvious. □

**Theorem 4.** Each dense and fast code is rich.

**Proof.**

Let \(C\) be a code that is dense and fast. Then there exists a finite set \(F\) of ordered pairs of words from \(\Sigma^*\) such that for each \(w \in \Sigma^*\) there exists \((x, y) \in F\) such that \(x \ w \ y \in C^*\). Let \(q = \max\{|xy| : (x, y) \in F\}\), \(f = \#F\) and \(d = f \sigma^q\).

**Claim 2.** For each positive integer \(n\) there exists a positive integer \(m \leq n + q\) such that \(\alpha_m(C^*) \geq \frac{\sigma^n}{d}\).

**Proof of Claim 2.**

Let for each \(w \in \Sigma^*\), \(pair(w)\) be a fixed element \((x, y)\) of \(F\) such that \(x \ w \ y \in C^*\).

Let \(n\) be a positive integer. Let \(E(n, x, y) = \{w \in L_n(\Sigma^*) : pair(w) = (x, y)\}\). Clearly for some \((x_0, y_0) \in F\), \(\#E(n, x_0, y_0) \geq \frac{\sigma^n}{f}\). Let \(p = |x_0 \ y_0|\). Then
\[ \alpha_{n+p}(C^*) \geq \#E(n, x_0, y_0) \geq \frac{\sigma^n}{f}. \]

Hence
\[ \alpha_{n+p}(C^*) \geq \frac{\sigma^n}{f} \geq \frac{\sigma^{n+p}}{f \sigma^p} \geq \frac{\sigma^{n+p}}{d}. \]

Thus if we choose \( m = n + p \) we get \( m \leq n + q \) and Claim 2 holds. \( \blacksquare \)

Now Theorem 4 follows directly from Claim 2. \( \blacksquare \)

Remark. Theorems 2 and 4 together are more general than Theorem 7.4 (due to Schützenberger) from \([E]\). However, it is pointed out by D. Perrin in \([P3]\) that a proof of the general case can be retrieved from the proof of Theorem 9.3 in \([E]\). \( \blacksquare \)

Theorem 5. Let \( C \) be a regular code. There exists a code \( D \) which is dense, fast, regular and such that \( C \subseteq D \).

Proof.

Let \( C \) be a regular code.

We consider separately two cases.

(i) \( C \) is dense.

Then the theorem follows from Theorem 3 (take \( D = C \)).

(ii) \( C \) is not dense.

Then, by Claim 1, there exists an unbordered word \( w_C \) such that \( w_C \notin \text{sub}(C^*) \).

Let \( A = \{w_C x_1 w_C x_2 \cdots w_C x_n w_C : n \geq 1, \geq x_i \notin C^* \text{ and } w_C \notin \text{sub}(x_i)\} \)
and let \( D = C \cup \{w_C\} \cup A \).

Claim 3. \( D \) is a code.

Proof of Claim 3.

Let \( y \in D^+ \). Since \( w_C \) is unbordered, \( y \) has a unique representation of the form \( y = x_1 w_C x_2 w_C \cdots x_n w_C \) (that is we can uniquely distinguish all occurrences of \( w_C \) in \( y \)).
This representation provides the basis for the division of $y$ into $D$-blocks which is obtained as follows:

1. A subword $w_C x_j w_C x_{j+1} \ldots w_C x_{j+l} w_C$ constitutes a $D$-block (corresponding to $A$) if $2 \leq j \leq n-1$, $j+l \leq n-1$, $x_j, \ldots, x_{j+l} \notin C^*$ and $x_{j-1}, x_{j+l+1} \in C^*$; such a $D$-block is referred to as a $A$-block.

2. All occurrences of $w_C$ not involved in $A$-blocks are also $D$-blocks.

3. All $x_i$'s which are not involved in $A$-blocks must be in $C^*$ and so they are uniquely divisible in $D$-blocks (really $C$-blocks).

The definition of $A$ and the fact that $w_C \notin \text{sub } (C^*)$ and $w_C$ is unbordered guarantee that such a division is unique.

Hence $D$ is a code and Claim 3 holds. 


Proof of Claim 4.

Let $u \in \Sigma^*$.

Consider $y = w_C u w_C$. Reasoning as in the proof of Claim 3 we get a (unique) representation of $y$ in $D^*$.

Thus $D$ is dense and Claim 4 holds. 

Claim 5. $D$ is regular.

Proof.

Obvious. 


Proof.

This follows from Claim 5 and Theorem 3. 

Now Theorem 5 follows from Claims 3 through 5. 

Our results yield two interesting corollaries. The first one solves an open problem from the theory of codes (see, e.g., [R] and [P2]). As a matter of fact it provides a more general result: Restivo has asked ([R]) whether an arbitrary finite code can be completed to a maximal regular code - we show that even an arbitrary regular code can be completed to a maximal regular code.

**Corollary 1.** Let \( C \) be a code. If \( C \) is regular, then there exists a code \( D \) such that \( C \subseteq D \), \( D \) is maximal and \( D \) is regular.

**Proof.**

Let \( C \) be a regular code.

By Theorem 5 there exists a regular code \( D \) such that \( C \subseteq D \), \( D \) is fast and dense.

Thus, by Theorem 4, \( D \) is rich and so, by Theorem 2, \( D \) is maximal.

Hence Corollary 1 holds. 

Secondly, we notice that Theorems 1 through 4 provide an alternative proof of the theorem by Schutzenberger (see [E] p. 94).

**Corollary 2.** Let \( C \) be a regular code. Then \( C \) is maximal if and only if \( C \) is dense.

**Proof.**

It follows directly from Theorems 1 through 4.
DISCUSSION

We have established a number of relationships between dense, fast, rich, maximal and regular codes. Using these relationships we were able to demonstrate that each regular code is included in a maximal regular code.

In particular we have demonstrated that each rich code is maximal and each maximal code is dense. Hence each rich code is dense. We provide now a "direct" proof of this result - we believe it sheds a different light on this relationship.

Corollary 3. Each rich code is dense.

Proof.

Let \( C \) be a rich code.

Assume that \( C \) is not dense. Hence there exists a word \( z \notin \text{sub}(C^*) \); let \( |z| = t \). Let \( n \) be an arbitrary positive integer; \( n \) can be represented in the form \( n = k_1 t + k_2 \) for some \( k_1 \geq 0 \) and \( k_2 < t \). An arbitrary word from \( L_n(C^*) \) can be (starting from the left end) divided into \( k_1 \) consecutive subwords of length \( t \) leaving a suffix of length \( k_2 \). Thus

\[
\alpha_n(C^*) < (\sigma^t - 1) k_1 \sigma^{k_2}.
\]

Consequently

\[
\frac{\alpha_n(C^*)}{\sigma^n} \leq \frac{(\sigma^t - 1) k_1 \sigma^{k_2}}{\sigma^{k_1} \sigma^{k_2}} = \frac{(\sigma^t - 1) k_1 \sigma^{k_2}}{\sigma^{k_1} \sigma^{k_2}} = \frac{1}{\sigma^t}.
\]

Hence

\[
\lim_{n \to \infty} \frac{\alpha_n(C^*)}{\sigma^n} = 0
\]

which contradicts the fact that \( C \) is rich.

Consequently \( C \) must be dense and the result holds. \( \Box \)

To put some of the dependencies we have demonstrated in a better perspective we provide now the following result.
Theorem 6. There exists a maximal code which is not rich.

Proof.

Consider the family of all full binary trees in which leaves are labelled by \( a \) and all inner nodes are labelled by \( b \). Consider now all postfix notations for these trees - in this way we get the language \( P \subseteq \{a, b\}^+ \). It is well known that \( P \) is a code (every forest of full binary trees has a unique representation in the postfix notation).

Consider an arbitrary word \( z \in \{a, b\}^+ - P \). Clearly \( a^{|z|+1}z \in P^+ \) (we parse \( a^{|z|+1}z \) from right to left assigning \(+1\) to \( a \) and \(-1\) to \( b \); then each subword yielding by summation weight \(+1\) is a tree corresponding to an element of \( P \)). Hence \( P \cup \{z\} \) is not a code, because \( a^{|z|+1}z \) would have two different representations in \( P^+ \). Thus \( P \) is a maximal code.

On the other hand it is known (see, e.g., [F], Ch. III, Sect.3) that
\[
\lim_{n \to \infty} \frac{\alpha_n(P^+)}{2^n} = 0.
\]
(Here one considers random walks on the line of positive integers where \( a \) represents a "step up" and \( b \) represents a "step down". It turns out that the probability of starting in 0 and not returning to 1 in up to \( n \) steps equals 1 in the limit).

Hence \( P \) is not rich and the theorem holds. ♦

Perhaps the most significant open question in the area of "extending codes to their maximal counterparts" is (see [P2]): can every biprefix regular code be extended to a maximal biprefix regular code? An answer to this question will certainly make the picture of the whole area clearer.
ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support of NSF grant MCS 79-03838. The authors are indebted to J. Karhumaki and D. Perrin for their comments on the first version of this paper.
REFERENCES


EACH REGULAR CODE IS INCLUDED
IN A MAXIMAL REGULAR CODE

by

A. Ehrenfeucht** and G. Rozenberg***

CU-CS-236-82 September, 1982

**University of Colorado, Department of Computer Science, Boulder, Colorado

*Institute of Applied Mathematics and Computer Science, University of Leiden, Leiden, The Netherlands and University of Colorado, Department of Computer Science, Boulder, Colorado
ANY OPINIONS, FINDINGS, AND CONCLUSIONS OR RECOMMENDATIONS EXPRESSED IN THIS PUBLICATION ARE THOSE OF THE AUTHOR AND DO NOT NECESSARILY REFLECT THE VIEWS OF THE NATIONAL SCIENCE FOUNDATION.