# Some classes of strong codes

## Yoshiyuki Kunimochi Shizuoka Institute of Science and Technology

**abstract** Deletion and insertion are interesting and common operations which often appear in text editing. A language  $L \subset A^*$  closed under the both operations forms a free submonoid of  $A^*$ . Its base C is called a strong code, which is a kind of bifix code. If strong code C is regular(resp. maximal), then its syntactic monoid Syn(C) is finite (resp. group). The class of hyper-strong codes is a subclass of strong codes. A hyper-strong code C and its syntactic monoid Syn(C) is commutative.

## **1** Preliminaries

Let A be a finite nonempty set of *letters*, called an *alphabet* and let  $A^*$  be the free monoid generated by A under the operation of catenation with the identity called the *empty word*, denoted by 1. We call an element of  $A^*$  a word over A. The free semigroup  $A^* \setminus \{1\}$  generated by A is denoted by  $A^+$ . The catenation of two words x and y is denoted by xy. The *length* |w|of a word  $w = a_1a_2...a_n$  with  $a_i \in A$  is the number n of occurrences of letters in w. Clearly, |1| = 0. For a letter a in A, we let  $|w|_a$  denote the number of occurrences of a in w. We denote  $\{a \in A \mid xay \in L, x, y \in A^*\}$  by alph(L).

A word  $u \in A^*$  is a *prefix*(resp. *suffix*) of a word  $w \in A^*$  if there is a word  $x \in A^*$  such that w = ux(resp. w = xu). A word  $u \in A^*$  is a *factor* of a word  $w \in A^*$  if there exist words  $x, y \in A^*$  such that w = xuy. Then a prefix (a suffix or a factor) u of w is called *proper* if  $w \neq u$ .

A subset of  $A^*$  is called a *language* over A. A nonempty language C which is the set of free generators of some submonoid M of  $A^*$  is called a *code* over A. Then C is called the *base* of M and coincides with the minimal set  $Min(M) = (M \setminus 1) \setminus (M \setminus 1)^2$  of generators of M. A nonempty language C is called a *prefix* (or *suffix*) code if  $u, uv \in C$  (resp. $u, vu \in C$ ) implies v = 1. C is called a *bifix* code if C is both a prefix code and a suffix code. A nonempty language C is called a *bifix* code if  $u_1u_2 \ldots u_nu_{n+1}, u_1v_1u_2v_2 \ldots u_nv_nu_{n+1} \in C$  implies  $v_1v_2 \ldots v_n = 1$ . The language  $A^n = \{w \in A^* \mid |w| = n\}$  with  $n \ge 1$  is called a *full uniform* code over A. A code C is called *maximal* if  $C \cup \{w\}$  is not a code for any  $w \in A^* \setminus C$ . A nonempty subset of  $A^n$  is called a *uniform* code over A. The symbols  $\subset$  and  $\subsetneq$  are used for a subset and a proper subset respectively.

A language L over A is called reflexive (resp. commutative) if  $uv \in L$  implies  $vu \in L$  (resp.  $xuvy \in L$  implies  $xvuy \in L$ ). The conjugacy class cl(w) of a word w is the set  $\{vu|w = uv\}$  and  $w' \in cl(w)$  is called a conjugate of w.

Let N be a submonoid of a monoid M. N is right unitary (in M) if  $u, uv \in N$  implies  $v \in N$ . Left unitary is defined in a symmetric way. The submonoid N of M is biunitary if it is both left and right unitary. Especially when  $M = A^*$ , a submonoid N of  $A^*$  is right unitary (resp. left unitary, biunitary) if and only if the minimal set  $N_0 = (N \setminus 1) \setminus (N \setminus 1)^2$  of generators of N, namely the base of N, is a prefix code (resp. a suffix code, a bifix code) ([1] p.46). Let L be a subset of a monoid M, the congruence  $P_L = \{(u, v) | \text{ for all } x, y \in M, xuy \in L \iff xvy \in L\}$  on M is called the *principal congruence*(or *syntactic congruence*) of L. We write  $u \equiv v$  ( $P_L$ ) instead of  $(u, v) \in P_L$ . The monoid  $M/P_L$  is called the *syntactic monoid* of L, denoted by Syn(L). The morphism  $\sigma_L$  of M onto Syn(L) is called the *syntactic morphism* of L.  $\sigma_L(w)$  is denoted by  $\overline{w}_L$ . In particular when  $M = A^*$ , a language  $L \subset A^*$  is regular if and only if Syn(L) is finite([1] p.46).

### 2 Strong Codes

A strong code C is the base of the identity  $\overline{1}_L$  in the syntactic monoid Syn(L) of some language L. Then we state some properties of strong codes.

#### 2.1 definitions

At first, we give the definition of strong codes.

**DEFINITION 2.1** [13] A code  $C \subset A^+ \setminus \{\emptyset\}$  is called a *strong* code if

(i) 
$$x, y_1 y_2 \in C^* \implies y_1 x y_2 \in C^*$$
  
(ii)  $x, y_1 x y_2 \in C^* \implies y_1 y_2 \in C^*$ 

Here extractable codes and insertable codes are introduced below.

**DEFINITION 2.2** Let  $C \subset A^+ \setminus \{\emptyset\}$  be a code. Then, C is called an insertable (or extractable) code if C satisfies the condition (i)( or (ii)).

Note that when C satisfies the condition (ii), we can easily check that  $C^*$  is biunitary(and thus free). Indeed,  $uv = 1uv, u \in C^*$  implies  $v = 1v \in C^*$  and  $uv = uv1, v \in C^*$  implies  $u = 1u \in C^*$ . Then the minimal set  $C = (C^* \setminus 1) \setminus (C^* \setminus 1)^2$  of generators of  $C^*$  becomes a bifix code. Therefore both strong codes and extractable codes are necessarily bifix codes.

Remark that an insertable submonoid M of  $A^*$ , the minimal set of generators of M is not necessarily a code. For example, If  $C = \{a^2, a^3\}$ , then the submonoid  $C^*$  is insertable but its minimal set C of generators is not necessarily a code.

A strong code C is described as the base of the identity  $P_L$ -class  $\overline{1}_L = \{w \in A^* \mid w \equiv 1(P_L)\}$ of the syntactic monoids Syn(L) of some language L.

**PROPOSITION 2.1** [13] Let  $L \subset A^*$ . Then  $C = (\overline{1}_L \setminus 1) \setminus (\overline{1}_L \setminus 1)^2$  is a strong code if it is not empty. Conversely, if  $C \subset A^+$  is a strong code, then there exists a language  $L \subset A^*$  such that  $\overline{1}_L = C^*$ .

Moreover if a strong code C is finite, the following proposition holds.

**PROPOSITION 2.2** [13] Let C be a finite strong code over A and B = alph(C). Then,  $C = B^n$  for some positive integer n, that is, C is a full uniform code over B.

**EXAMPLE 2.1** (1) A singleton  $\{w\}$  with  $w \in \{a\}^+$  is a strong code.  $\{w\}$  with  $w \in A^+ \setminus \bigcup_{a \in A} \{a\}^+$  is not a strong code but it is an extractable code. Therefore there exist finite extractable codes which are not full uniform codes.

- (2) The conjugacy class cl(ab) of ab is an extractable code but not a strong code.
- (3)  $\{a^n b^n \mid n \text{ is an integer}\}\$  is an (context-free) extractable code but not a strong code.
- (4)  $a^*b$  and  $ba^*$  are (regular) insertable codes but not strong codes.

**PROPOSITION 2.3** [18] Let C be a code over A. Then the following conditions are equivalent:

(1)  $C^*$  is reflexive;

(2) C is a maximal strong code over A;

(3)  $C^*$  is a  $P_{C^*}$ -class,  $Syn(C^*)$  is a group.

Note that the condition (2) is equivalent to the following condition (2'):

(2') C is a strong code over A and alph(C) = A.

Indeed, if  $a \in A \setminus alph(C)$ , then  $C \cup \{a\}$  is a code. This contradicts to the condition (2). Hence alph(C) = A. Conversely, suppose the condition (2'), that is A = alph(C). We show that  $C \cup \{w\}$  with any  $w = a_1a_2...a_k \notin C(a_i \in A, 1 \le i \le k)$  cannot be a code. For any  $a_i \in A$ ,  $a_iy_i \in C^*$  for some  $y_i \in A^*$  because  $C^*$  is reflexive. Therefore  $w(y_k...y_2y_1) = a_1a_2...a_ky_k...y_2y_1 = c_1c_2...c_m \in C^*$  for some  $c_j \in C(1 \le j \le m)$ . Since  $C^*$  is reflexive again,  $(y_k...y_2y_1)w = c'_1c'_2...c'_n \in C^*$  for some  $c'_j \in C(1 \le j \le n)$ . Therefore  $c_1c_2...c_mw = wc'_1c'_2...c'_n \in C^*$ . This proves that  $C \cup \{w\}$  is not a code.

#### 2.2 Insertion and Deletion

Let L be a language over A. A language L is called ins-closed if  $u = u_1u_2 \in L$  and  $v \in L$ imply  $u_1vu_2 \in L$ . A language L is called del-closed if  $u = u_1vu_2 \in L$  and  $v \in L$  imply  $u_1u_2 \in L$  [4].

Let L be a del-closed language. Then, Since L is biunitary, the minimal set C = min(L) of generators of L is a bifix code and  $L = C^*$ .

Let L be an ins-closed language. Then,  $1 \in L$  and  $L^2 \subset L$  implies Since L is a submonoid of  $A^*$ .

**PROPOSITION 2.4** Let  $L \neq \emptyset$  be an ins-closed and del-closed language over A. Then  $L = C^*$  for some strong code C.

Proof) As we stated above, L is a submonoid of  $A^*$  and its minimal set C of generators is a (bifix) code. C satisfies the conditions of a strong code.

#### 2.3 Roots of Strong Codes

Let L be a strong code over A. We define a relation  $\rho$  on the free submonoid  $C^*$  of  $A^*$  as follows:

 $u\rho v$  if and only if there exist  $m \in C^+$   $x_1, x_2 \in A^*$  such that  $u = x_1 x_2$  and  $v = x_1 m x_2$ .

Let  $\overline{\rho}$  the reflexive and transitive closure of  $\rho$ .

**DEFINITION 2.3** [18] Let C be a strong code over A. The root of C is the set:

$$R(C) = \{ c \in C^+ | \forall c_1 \in C^+(c_1 \overline{\rho} c) \to c_1 = c \}.$$

**PROPOSITION 2.5** [18] Let C be a strong code over A. Then the following conditions are equivalent:

(1) C is maximal;

(2) R(C) is reflexive;

**EXAMPLE 2.2** Let  $\Sigma$  be an alphabet and let  $\overline{\Sigma} = {\overline{a} \mid a \in \Sigma}$  be its copy. The Dyck language  $D_{\Sigma}$  over  $\Sigma$  is generated by the context-free grammar  $({S}, \Sigma \cup \overline{\Sigma}, P, S)$ , where

$$S \to \varepsilon, S \to aS\bar{a}S \ (a \in \Sigma).$$

 $D_{\Sigma}$  is a free submonoid of  $(\Sigma \cup \overline{\Sigma})^*$  and its base  $DP_{\Sigma}$  is a strong code over  $\Sigma \cup \overline{\Sigma}$ . If  $|\Sigma| = n$ , then  $D_{\Sigma}$ (resp.  $DP_{\Sigma}$ ) is often denoted by  $D_n$ (resp.  $DP_n$ ).

 $DP_n$  is not a regular language. The root of  $DP_n$  is the set  $R(DP_n) = \{a\overline{a} \mid a \in \Sigma\}$ 

**PROPOSITION 2.6** [18] Let C be a strong code over A. If the root R(C) is finite, the there exist a Dyck language  $D_k \subset (A_1)^*$  and a homomorphism  $f : (A_1)^* \to A^*$  such that  $C^* = f(D_k)$ 

The following corollary and proposition give a necessary condition and a sufficient condition that a strong code has a finite root, respectively.

**COROLLARY 2.1** [18] Let C be a strong code over A. If the root R(C) is finite, then  $C^*$  is context-free.

**PROPOSITION 2.7** [18] Let C be a strong code over A. If C is regular, then the root R(C) is finite.

Zhang conjectured that a strong code has a finite root if and only if it is a simple language. Whereas Harging-Smith[2] proved the following theorem in 1973. In the theorem, Let  $\pi = \langle A; R \rangle$  be a finitely generated presentation of a group G, and  $\Sigma = A \cup A^{-1}$  be the set of generators and their inverses. The word problem  $WP(\pi)$  of  $\pi$  is the set of all words on  $\Sigma$  which are equal to the identity. The reduced word problem  $WP_0(\pi)$  of  $\pi$  is the set  $WP(\pi) \setminus WP(\pi)\Sigma^+$ . The set  $W(\pi)$  of irreducible words is the set  $WP(\pi) \setminus \Sigma^+WP(\pi)\Sigma^+$ 

**DEFINITION 2.4** A context-free grammar  $G = (V, \Sigma, P, S)$  in Greibach normal form is said to be a simple grammar if for all  $A \in N$ ,  $a \in \Sigma$ , and  $\alpha, \beta \in V^*$ ,

 $A \to a\alpha$ , and  $A \to a\beta$  imlpy  $\alpha = \beta$ .

A simple language is a language generated by a simple grammar.

**THEOREM 2.1** [2] The reduced word problem  $WP_0(\pi)$  of a finitely generated group presentation  $\pi$  is a simple language if and only if the set of irreducible words  $W(\pi)$  is finite.

**EXAMPLE 2.3** The language  $L = \{w \mid |w|_a = |w|_b\}$  over  $A = \{a, b\}$  is ins-closed and delclosed. L is a free submonoid of  $A^*$ . Its base C = min(L) is a maximal strong code of even length over A. The root R(C) of C is the set  $R(C) = \{ab, ba\}$ 

## 3 Hyper-strong codes

A hyper-strong code is referred in the literature [18], but its definition is not described. Here we give the definition of hyper-strong codes below.

**DEFINITION 3.1** [18] Let *n* be a positive integer. A code  $C \subset A^+ \setminus \{\emptyset\}$  is called a *n*-strong code if

(i)  $x_1x_2...x_n, y_1y_2...y_ny_{n+1} \in C^* \Longrightarrow y_1x_1y_2x_2...y_nx_ny_{n+1} \in C^*$ (ii)  $x_1x_2...x_n, y_1x_1y_2x_2...y_nx_ny_{n+1} \in C^* \Longrightarrow y_1y_2...y_ny_{n+1} \in C^*$ 

A 1-strong code is a strong code, and vice versa. An (n + 1)-strong code is an *n*-strong code. *C* is called a *hyper-strong* code if *C* is *n*-strong code for each integer n > 0.

**PROPOSITION 3.1** [18] Let C be a code over A = alph(C). Then the following conditions are equivalent:

(1) C is a maximal hyper-strong code over A;

(2)  $C^*$  is commutative;

(3)  $C^*$  is a  $P_{C^*}$ -class,  $Syn(C^*)$  is a commutative group. (4) R(C) is commutative;

**PROPOSITION 3.2** Every commutative code is a hypercode.

**PROPOSITION 3.3** [3] Every hypercode is finite.

**COROLLARY 3.1** Let C be a maximal hyper-strong code over A. R(C) is finite.

The following is an example which is a hyper-strong code but is not a strong code.

**EXAMPLE 3.1** Let  $\Sigma$  be an alphabet and let  $\overline{\Sigma} = {\overline{a} | a \in \Sigma}$  be its copy. The semi-Dyck language  $D'_{\Sigma}$  over  $\Sigma$  is generated by the context-free grammar  $({S}, \Sigma \cup \overline{\Sigma}, P, S)$ , where

$$S \to \varepsilon, S \to aS\bar{a}S, S \to \bar{a}SaS \ (a \in \Sigma).$$

 $D'_{\Sigma}$  is a free submonoid of  $(\Sigma \cup \overline{\Sigma})^*$  and its base  $DP'_{\Sigma}$  is a hyper-strong code over  $\Sigma \cup \overline{\Sigma}$ . If  $|\Sigma| = n$ , then  $D'_{\Sigma}$  (resp.  $DP'_{\Sigma}$ ) is often denoted by  $D'_n$ (resp.  $DP'_n$ ).

 $DP'_n$  is a hyper-strong code and not a regular language. The root of  $DP'_n$  is the set  $R(DP'_n) = \{a\overline{a}, \overline{a}a \mid a \in \Sigma\}$  and is a commutative code.

## References

- [1] J. Berstel and D. Perrin. *Theory of Codes*. Pure and Applied Mathematics. Academic Press, 1985.
- [2] G. H. Haring-Smith. Groups and simple languages, volume 239. 9 1983.
- [3] M. A. Harrison. Introduction to Formal Language Theory. Addison-Wesley Series in Computer Science, Addison-Wesley, 1978.
- [4] M. Ito, L. Kari, and G. Thierrin. Insertion and deletion closure of languages. *Theoretical Computer Science*, 183:3–19, 1997.

- [5] J.M.Howie. Fundamentals of Semigroup Theory. London Mathematical Society Monographs New Series 12. Oxford University Press, 1995.
- [6] Y. Kunimochi. Some properties of extractable codes and insertable codes. *International Journal of Foundations of Computer Science*, 27(3):327–342, 2016.
- [7] G. Lallement. Semigroups and combinatorial applications. John Wiley & Sons, Inc., 1979.
- [8] D. Long. On the structure of some group codes. 45:38–44, 1992.
- [9] M. Lothaire. *Combinatorics on Words*, volume 17 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, 1983.
- [10] T. Moriya and I. Kataoka. Syntactic congruences of codes. *IEICE TRANSACTIONS on Information and Systems*, E84-D(3):415–418, 2001.
- [11] M.Petrich and G.Thierrin. The syntactic monoid of an infix code. *Proceedings of the American Mathematical Society*, 109(4):865–873, 1990.
- [12] G. Rozenberg and A. Salomaa. Handbook of Formal Languages, Vol.1 WORD, LANGUAGE, GRAMMAR. Springer, 1997.
- [13] H.J.Shyr. Strong codes. Soochow J. of Math. and Nat. Sciences, 3:9–16, 1977.
- [14] H.J.Shyr. *Free monoids and Languages*. Lecture Notes. Hon Min book Company, Taichung, Taiwan, 1991.
- [15] G. Tanaka, Y. Kunimochi, and M. Katsura. Remarks on extractable submonoids. *Technical Report kokyuroku, RIMS, Kyoto University*, 1655:106–110, 6 2009.
- [16] S. Yu. A characterization of intercodes. *International Journal of Computer Mathematics*, 36(1-2):39–45, 1990.
- [17] S.-S. Yu. Languages and Codes. Tsang Hai Book Publishing Company, Taiwan, 2005.
- [18] L. Zhang. Rational strong codes and structure of rational group languages. 35(1):181–193, 1987.
- [19] L. Zhang and W. Qiu. Decompositions of recognizable strong maximal codes. 108:173–183, 1993.
- [20] L. Zhang and W. Qiu. On group codes. 163:259–267, 1996.

Yoshiyuki Kunimochi

Shizuoka Institute of Science and Technology

Toyosawa 2200-2, Fukuroi-shi, Shizuoka 437-8555,

JAPAN

Email: kunimochi.yoshiyuki@sist.ac.jp