ARTICLE IN PRESS

Theoretical Computer Science ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at SciVerse ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

Relativized codes

Mark Daley [a], Helmut Jürgensen [a,*], Lila Kari [a], Kalpana Mahalingam [b]

[a] Department of Computer Science, The University of Western Ontario, London, Ontario, Canada, N6A 5B7

[b] Department of Mathematics, IIT Madras, Chennai, 600 036, India

ABSTRACT

A code C over an alphabet Σ is a set of words such that every word in C^+ has a unique factorization over C, that is, a unique C-decoding. When not all words in C^+ appear as messages, a weaker notion of unique factorization can be used. Thus we consider codes C relative to a given set of messages L, such that each word in L has a unique C-decoding. We extend this idea of relativizing code concepts to restricted message spaces.

In general terms, from a predicate P defining a class of codes, P-codes, we derive a relativized version of such codes, P-codes relative to a given language L. In essence, $C \subseteq \Sigma^+$ is a P-code relative to $L \subseteq \Sigma^+$ if P is true on its domain restricted to L. This systematic approach leads to the relativization of the definitions of many classes of codes, including prefix, suffix, bifix and solid codes. It can also be applied to certain classes of languages, like overlap-free languages, which are not codes, but which can be defined using a similar logical framework.

In this paper, we explore the mechanism of this relativization and compare it to other existing methods for relativizing code properties to restricted message spaces.

1. Introduction

Codes are formal languages with special combinatorial and structural properties which are exploited in information processing and information transmission. The well-known model of information transmission consists of a source $\mathcal{S}$ with source alphabet Σ sending information to a recipient $\mathcal{R}$ with receiver alphabet Δ via a channel by encoding the messages with an encoder γ. In the sequel, by encoding we mean homomorphic encoding. The set $C = \gamma(\Sigma)$ is a set of codewords called the code of γ. The code C is usually required to have the property of unique decodability, that is, every word in C^+ should be uniquely decodable as a concatenation of words in C. Abstracting from this scenario, any uniquely decodable non-empty set C of non-empty words over Δ is called a code. Beyond the basic requirement of unique decodability, other properties of codes are relevant depending on the situation at hand: decoding delay, synchronization delay, error-resistance, error-recovery, error-correction, etc. A survey regarding the hierarchy of classes of codes is presented in [9]. In addition to [9], we refer to [1,13,15] for relevant information on codes.

Usually, not every word over the receiver alphabet will occur as an output of $\mathcal{S}$. For example, if $\mathcal{S}$ is an author writing English prose, the word $(xyz)^{1000000}$ is not likely to be one of the best-selling novels. For simplification, messages with a very low probability of being issued by $\mathcal{S}$ are considered as impossible in the sequel, leaving the set L of possible output messages to be considered. If only messages in L are encoded, then properties like unique decodability etc. need only be considered with respect to L. Thus, code properties can be *relativized to* L and preserved, even if the original set C is itself not a code. To our knowledge, this idea was first introduced in [4] and pursued further in [5–7]. In particular, the potential of such codes

* Corresponding author.
E-mail addresses: daley@csd.uwo.ca (M. Daley), hjj@csd.uwo.ca (H. Jürgensen), lila@csd.uwo.ca (L. Kari), kmahalingam@iitm.ac.in (K. Mahalingam).

0304-3975/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2011.12.024

in the context of DNA encodings was investigated in [12]. Another approach to dealing with the issue of decodability in the case of C not being a code uses coding partitions [2].

Many natural classes of codes can be defined in terms of some independence properties [9]. A subset H of Σ^* is called an *independent set with respect to a partial order* $\leq$ if every pair of distinct words in H is incomparable with respect to the partial order $\leq$. If, for example, $\leq = \leq_{\mathrm{p}}$, the prefix order defined by $u \leq_{\mathrm{p}} v$ if the word u is a prefix of the word v, then H is an independent set with respect to $\leq_{\mathrm{p}}$ if and only if H is a prefix code. Thus, the family of all independent sets with respect to the prefix order is exactly the family of all prefix codes over Σ.

In Section 3, we relativize the general definition of a code, starting from a relation or a predicate that defines it. Given a non-empty set $C \subseteq \Sigma^+$, not necessarily a code, and a predicate P on the set of finite subsets of Σ^*, a word $q \in C^+$ is called P-admissible for C if the following condition is satisfied: if $q = xuy = x'u'y'$, with $u, u' \in C$ and $x, x', y, y' \in C^*$, then $P(\{u, u'\}) = 1$. The predicates P to which this definition applies include those defining prefix, suffix, infix and outfix codes. For example, the predicate P_{p} defines prefix-freeness as follows: for $u, u' \in \Sigma^+$, $P_{\mathrm{p}}(\{u, u'\}) = 1$, if and only if u is not a proper prefix of u' and u' is not a proper prefix of u. Then a word $q \in C^+$ is prefix-admissible (P_{p}-admissible) for C if no two words $u, u' \in C$ appearing in any decoding of q over C are strict prefixes of each other. We observe that, for several standard predicates P (prefix-free, suffix-free, infix-free, outfix-free, etc.), if a word is P-admissible then it is uniquely decodable over C. Thus, based on the notion of P-admissibility, we define the notion of a P-code relative to a language as follows: a set C is called a P-code relative to a language $L \subseteq C^+$ if every word q in L is P-admissible for C. When $L = C^+$ and P is one of the standard predicates, then the notion of P-code relative to $L = C^+$ coincides with the respective classical notion of prefix code, suffix code, infix code, outfix code, etc.

We compare our method of relativizing the notion of code with other approaches: *coding partitions* (Section 3.1); *relative solid codes* (Section 3.2); *joins of a code* (Section 3.3).

In Section 4, we derive several general properties of relativized codes and also of special classes of such codes. We show for example that, for the predicates mentioned above, the relativized codes form a hierarchy similar to the one for classical codes and that, for any given word q which is P-admissible for C, we can extract a subset $C_q \subseteq C$ such that $q \in C_q^+$, C_q is a P-code, and C_q is minimal with respect to this property.

2. Notation and basic notions

The sets of positive integers and of non-negative integers are $\mathbb{N}$ and $\mathbb{N}_0$, respectively. An alphabet is a non-empty set. To avoid trivial special cases, we assume that an alphabet has at least two elements. Throughout this paper Σ is an arbitrary, but fixed, alphabet. When required we add the assumption that Σ is finite. A word over Σ is a finite sequence of symbols from Σ; the set Σ^* of all words over Σ, including the empty word λ, is a free monoid generated by Σ with concatenation of words as multiplication. The set of non-empty words is Σ^+, that is, $\Sigma^+ = \Sigma^* \setminus \{\lambda\}$. A language over Σ is a subset of Σ^*. For a language $L \subseteq \Sigma^*$ and $n \in \mathbb{N}_0$ let

$$L^n = \begin{cases} \{\lambda\}, & \textit{if}\ \mathrm{n{=}0}, \\ L, & \textit{if}\ \mathrm{n{=}1}, \\ \left\{ w \;\middle|\; \exists u \in L\, \exists v \in L^{n-1} : w = uv \right\}, & \textit{if}\ \mathrm{n{>}1}. \end{cases}$$

Moreover, let

$$L^* = \bigcup_{n \in \mathbb{N}_0} L^n \text{ and } L^+ = \bigcup_{n \in \mathbb{N}} L^n.$$

If P is a property of languages, then $\mathcal{L}_P(\Sigma)$ is the set of languages L over Σ for which $P(L) = 1$, that is, $P(L)$ is true. We write $\mathcal{L}_P$ instead of $\mathcal{L}_P(\Sigma)$ when Σ is understood. *In the remainder of this paper, unless explicitly stated otherwise, all languages are assumed to be non-empty.*

Many classes of codes and related languages can be defined systematically in terms of relations on the free monoid Σ^+ or in terms of abstract dependence systems. See [9,10,14,15] for details. In the present paper only the following relations between words $u, v \in \Sigma^+$ are considered:

Property	*Definition*	*Notation*
u is a prefix of v:	$v \in u\Sigma^*$	$u \leq_{\mathrm{p}} v$
u is a proper prefix of v:	$v \in u\Sigma^+$	$u <_{\mathrm{p}} v$
u is a suffix of v:	$v \in \Sigma^* u$	$u \leq_{\mathrm{s}} v$
u is a proper suffix of v:	$v \in \Sigma^+ u$	$u <_{\mathrm{s}} v$
u is an infix of v:	$v \in \Sigma^* u \Sigma^*$	$u \leq_{\mathrm{i}} v$
u is a proper infix of v:	$(u \leq_{\mathrm{i}} v) \wedge (u \neq v)$	$u <_{\mathrm{i}} v$
u is an outfix of v:	$\exists u_1, u_2\ (u = u_1 u_2 \wedge v \in u_1 \Sigma^* u_2)$	$u\ \omega_{\mathrm{o}}\ v$
u is a proper outfix of v:	$(u\ \omega_{\mathrm{o}}\ v) \wedge (u \neq v)$	

We say that u is a *scattered subword* of v, and we write $u\ \omega_{\mathrm{h}}\ v$, if, for some $n \in \mathbb{N}$, there are $u_1, u_2, \ldots, u_n \in \Sigma^*$ and $v_1, v_2, \ldots, v_{n+1} \in \Sigma^*$ such that $u = u_1 u_2 \cdots u_n$ and $v = v_1 u_1 v_2 u_2 \cdots u_n v_{n+1}$. We say that u and v *overlap*, and we write $u\ \omega_{\mathrm{ol}}\ v$, if there is $q \in \Sigma^+$ such that $q <_{\mathrm{p}} u$ and $q <_{\mathrm{s}} v$ or vice versa. The relation ω_{ol} is symmetric.

A binary relation ω on Σ^+ defines the property (predicate) P_ω of languages $L \subseteq \Sigma^+$ as follows: $P_\omega(L) = 1$ if and only if, for all $u, v \in L$, one has $u \not\omega\, v$ and $v \not\omega\, u$. Clearly, if $P_\omega(L) = 1$ and $L' \subseteq L$, then $P_\omega(L') = 1$. Thus $P_\omega(L) = 1$ if and only if $P_\omega(\{u, v\}) = 1$ for all $u, v \in L$. Here the words u and v need not be distinct. This is important for the case of ω_{ol} for instance. Obviously, when ω is reflexive one has $P_\omega(L) = 0$ for every non-empty language L.

When $\omega = <_{\mathrm{p}}$ we write P_{p} instead of $P_{<_{\mathrm{p}}}$. Similarly, when $\omega = \omega_{\mathrm{ol}}$ we write P_{ol} instead of $P_{\omega_{\mathrm{ol}}}$. The predicates P_{s}, P_{i} and P_{o} are defined analogously starting from $<_{\mathrm{s}}$, $<_{\mathrm{i}}$ and ω_{o}, respectively.

For a set S, $\mathfrak{P}(S)$ is the set of all subsets of S and $\mathfrak{P}_{\mathrm{fin}}(S)$ is the set of all finite subsets of S. For $n \in \mathbb{N}$, let

$$\mathfrak{P}_{\le n}(S) = \{T \mid T \in \mathfrak{P}(S), |T| \le n\}$$

and

$$\mathfrak{P}_{= n}(S) = \{T \mid T \in \mathfrak{P}(S), |T| = n\}.$$

In [9] the hierarchy of classes of codes is introduced using the systematic framework of abstract dependence systems. For the purposes of the present paper, the following simplified concepts suffice.

For the remainder of this section, we refer to [9,15] and to sources cited there.

Let $C \subseteq \Sigma^+$. The language C is *uniquely decodable* if C^+ is a free subsemigroup of Σ^+ which is freely generated by C. A less abstract, but equivalent definition reads as follows:

Definition 2.1. Let $C \subseteq \Sigma^+$ be a language over Σ, and let $w \in \Sigma^+$.

1. The word w is *C-decodable* if there are $n \in \mathbb{N}$ and words $u_1, u_2, \ldots, u_n \in C$ such that $u_1 u_2 \cdots u_n = w$. In this case, the pair $(n, (u_1, u_2, \ldots, u_n))$ is called a *C-decoding* of w.
2. The language C is *uniquely decodable* if every word in Σ^+ has at most one C-decoding.

Thus a language C is uniquely decodable, if and only if every word in C^+ has a unique C-decoding. We omit the reference to C when C is understood from the context. In the following we sometimes use parentheses to describe various C-decodings of a word. For example, if $C = \{a, ab, ba\}$, then $w = aba = (a)(ba) = (ab)(a)$ has two different C-decodings.

As every word in C^+ involves only finitely many elements of C, the language C is uniquely decodable if and only if every language in $\mathfrak{P}_{\mathrm{fin}}(C)$ is uniquely decodable.

In the literature one finds the term "code" used in two different ways: (1) *a non-empty language not containing the empty word;* (2) *a uniquely decodable non-empty language not containing the empty word*. For the rest of this paper we adopt the second meaning. By $\mathcal{L}_{\mathrm{code}}$ we denote the set of codes over Σ. For a regular language $C \subseteq \Sigma^+$ it is decidable whether $C \in \mathcal{L}_{\mathrm{code}}$; for linear languages the code property is undecidable.

We now introduce some important classes of languages or codes. Further classes will be defined when they are needed. Let $C \subseteq \Sigma^+$.

For $n \in \mathbb{N}$ with $n > 1$, C is an *n-code* if every language in $\mathfrak{P}_{\le n}(C)$ is a code. In general, an n-code is not necessarily a code. By $\mathcal{L}_{n\text{-code}}$ we denote the set of n-codes over Σ. For regular C it is decidable whether $C \in \mathcal{L}_{2\text{-code}}$. For $\mathcal{L}_{3\text{-code}}$ the corresponding problem is open. The n-codes form an infinite descending hierarchy with $\mathcal{L}_{\mathrm{code}}$ as its lower bound.

The language C is a *prefix code* if, for all $u, v \in C$, $u \not<_{\mathrm{p}} v$. It is a *suffix code* if, for all $u, v \in C$, $u \not<_{\mathrm{s}} v$. It is a *bifix code* if it is both a prefix code and a suffix code. It is an *infix code* if, for all $u, v \in C$, $u \not<_{\mathrm{i}} v$. It is an *outfix code* if, for all distinct $u, v \in C$, $u \not\omega_{\mathrm{o}}\, v$. It is a *solid code* if it is an infix code and if, for all $u, v \in C$ not necessarily distinct, u and v do not overlap. The language C is a *hypercode* if, for all distinct $u, v \in C$, $u \not\omega_{\mathrm{h}}\, v$.

By $\mathcal{L}_{\mathrm{p}}$, $\mathcal{L}_{\mathrm{s}}$, $\mathcal{L}_{\mathrm{b}}$, $\mathcal{L}_{\mathrm{i}}$, $\mathcal{L}_{\mathrm{o}}$, $\mathcal{L}_{\mathrm{h}}$, and $\mathcal{L}_{\mathrm{solid}}$ we denote the sets of prefix codes, suffix codes, bifix codes, infix codes, outfix codes, hypercodes, and solid codes, respectively. The first six of these classes of codes are defined by predicates P_{p}, P_{s}, P_{b}, P_{i}, P_{o} and P_{h} on $\mathfrak{P}_{=2}(C)$. For $\mathcal{L}_{\mathrm{solid}}$ we need $P_{\mathrm{solid}} = P_{\mathrm{i}} \wedge P_{\mathrm{ol}}$ on $\mathfrak{P}_{\le 2}(C)$.

For $n \in \mathbb{N}$, the language C is an *intercode of index n* if, $\Sigma^+ C^n \Sigma^+ \cap C^{n+1} = \emptyset$. The class $\mathcal{L}_{\mathrm{inter}_n}$ of intercodes of index n is defined by a predicate P_{inter_n} on $\mathfrak{P}_{\le 2n+1}(C)$ derivable from P_{i}. The set $\mathcal{L}_{\mathrm{inter}_1}$ of intercodes of index 1 is exactly the set $\mathcal{L}_{\text{comma-free}}$ of *comma-free codes.* The languages in $\mathcal{L}_{\mathrm{inter}} = \bigcup_{n \in \mathbb{N}} \mathcal{L}_{\mathrm{inter}_n}$ are called *intercodes.*

Lemma 2.1 (*See [9,15]*). *The following inclusions hold:*

$$\mathcal{L}_{\mathrm{p}} \cup \mathcal{L}_{\mathrm{s}} \subsetneq \mathcal{L}_{\mathrm{code}}, \;\; \mathcal{L}_{\mathrm{i}} \cup \mathcal{L}_{\mathrm{o}} \subsetneq \mathcal{L}_{\mathrm{b}} = \mathcal{L}_{\mathrm{p}} \cap \mathcal{L}_{\mathrm{s}},$$
$$\forall n\; \mathcal{L}_{\mathrm{inter}_n} \subsetneq \mathcal{L}_{\mathrm{inter}_{n+1}} \subsetneq \mathcal{L}_{\mathrm{inter}} \subsetneq \mathcal{L}_{\mathrm{b}}, \;\; \mathcal{L}_{\mathrm{h}} \cap \mathcal{L}_{\mathrm{solid}} \subsetneq \mathcal{L}_{\mathrm{h}} \subsetneq \mathcal{L}_{\mathrm{i}} \cap \mathcal{L}_{\mathrm{o}}$$

and

$$\mathcal{L}_{\mathrm{h}} \cap \mathcal{L}_{\mathrm{solid}} \subsetneq \mathcal{L}_{\mathrm{solid}} \subsetneq \mathcal{L}_{\text{comma-free}} \subsetneq \mathcal{L}_{\mathrm{i}}.$$

It will simplify the notation significantly and also open the prospects of considering a different set of problems if we weaken the definitions as follows: for

$$\varrho \in \{\mathrm{p}, \mathrm{s}, \mathrm{b}, \mathrm{i}, \mathrm{o}, \mathrm{h}, \mathrm{solid}, \mathrm{ol}, \mathrm{inter}_n, n\text{-code}, \text{comma-free}\}$$

and potentially other types ϱ of language properties, P_ϱ is a predicate on $\mathfrak{P}_{\mathrm{fin}}(C)$ in the following sense: a language $L \subseteq C$ has the property ϱ if and only if $P_\varrho(L)$ holds true, that is, $P_\varrho(L) = 1$; for $\varrho \in \{\mathrm{p}, \mathrm{s}, \mathrm{b}, \mathrm{i}, \mathrm{o}, \mathrm{solid}, \mathrm{ol}\}$ we are mainly interested in situations when $|L| \le 2$ as this leads to manageable decision properties. As a warning to the reader – we have seen this misread before – the set $\{u, v\}$ is equal to $\{u\}$ when $u = v$, that is, $\{u, v\}$ is not a pair, but a set.

3. Relativizing codes

Consider a language $C \subseteq \Sigma^+$ which is going to be used for encoding messages. If there are no restrictions as to which messages to expect, one could encounter any word in C^+ as an encoded message. If, however, the set of messages to be encoded is restricted, then one may only encounter encoded messages in a set $L \subsetneq C^+$. We show that, by restricting the set of expected encoded messages to a language $L \subseteq C^+$, one can weaken the conditions C must satisfy while still preserving useful properties for all the words in L. For example, if we want the words in L to be uniquely C-decodable, then we may weaken the condition of unique decodability for C; in such a case, C need not be a code. If $L \subseteq C^+$ is a set of (encoded) messages which are uniquely C-decodable, then C is said to be *a code relative to L*.

Definition 3.1 (*[6]*). A set $C \subseteq \Sigma^+$ is said to be a *code relative to* $L \subseteq C^+$ if every word $q \in L$ has a unique C-decoding.

A language $C \subseteq \Sigma^+$ is a code if and only if it is a code relative to C^+.

In view of Definition 2.1 above and Proposition 3.2 below we rephrase Definition 3.1 as follows.

Remark 3.1. Let $L, C \subseteq \Sigma^+$. The language C is a code relative to L if every $q \in L$ has at most one C-decoding.

Example 3.1. Consider the set $C = \{a, ab, ba\}$ over the binary alphabet $\Sigma = \{a, b\}$. The given set C is not a code since the word $w = aba = (a)(ba) = (ab)(a)$ has two different C-decodings. Let

$$\begin{aligned} L &= \{ab, ba\}^+ \cup \{a^m(ab)^n \mid m \geq 0, n \geq 1\} \\ &\quad \cup \{(ba)^m a^n \mid m \geq 1, n \geq 0\} \cup \{a^m \mid m \geq 1\}. \end{aligned}$$

Then, for all $q \in L$, q has a unique C-decoding and, hence, C is a code relative to L.

Proposition 3.1. *Let $C \subseteq \Sigma^+$ and let $\mathcal{L}$ be the set of languages in C^+ such that C is a code relative to L for all $L \in \mathcal{L}$. Let*

$$L_{\max}(C) = \{q \mid q \text{ has a unique } C\text{-decoding}\}.$$

The set $\mathcal{L}$ is closed under arbitrary unions and

$$L_{\max}(C) = \bigcup_{L \in \mathcal{L}} L.$$

The set $\mathcal{L}$ is also closed under arbitrary intersections.[1]

Proof. Let $\mathcal{L}' \subset \mathcal{L}$ and $q \in \bigcup_{L \in \mathcal{L}'} L$. Then $q \in L$ for some $L \in \mathcal{L}$, and q has a unique C-decoding. Hence

$$\bigcup_{L \in \mathcal{L}'} L \in \mathcal{L}.$$

Let $\hat{L} = \bigcup_{L \in \mathcal{L}} L$. Thus, $\hat{L} \in \mathcal{L}$. As $L_{\max}(C) \in \mathcal{L}$, one has $L_{\max}(C) \subseteq \hat{L}$, but also $\hat{L} \subseteq L_{\max}(C)$.

The claim for the intersection follows from the fact that $\bigcup_{L \in \mathcal{L}'} \subseteq L_{\max}(C)$ for all $\mathcal{L}' \subseteq \mathcal{L}$. $\square$

Note that, even for an infinite language C, the set $L_{\max}(C)$ of all uniquely C-decodable messages over C can be finite.

Example 3.2. Let $C = \{(ab)^n \mid n \geq 1\}$. Then C is not a code, but C is a code relative to $L_{\max}(C) = \{ab\}$.

In the following definition, we establish a mechanism for relativizing code properties given by predicates or binary relations.

Definition 3.2. Let C be a subset of Σ^+ and let P be a predicate on $\mathfrak{P}_{\mathrm{fin}}(C)$. A word $q \in C^+$ is said to be *P-admissible for C* if the following condition is satisfied: if $q = xuy = x'u'y'$, with $u, u' \in C$ and $x, x', y, y' \in C^*$ then $P(\{u, u'\}) = 1$.

This means that a word $q \in C^+$ is P-admissible if every two words $u, u' \in C$ appearing in C-decodings of q, together satisfy the property P. For example, for $P = P_{\mathrm{p}}$, a word q is *prefix-admissible,* if no two words $u, u' \in C$ appearing in C-decodings of q are strict prefixes of each other. There is a subtle point: suppose that u' is a proper prefix of u. For a word q two different situations need to be considered:

1. The word q has a C-decoding of the form

   $$\cdots (u') \cdots (u) \cdots$$

   or

   $$\cdots (u) \cdots (u') \cdots.$$

   This is the situation considered in Definition 3.2.
2. The word q has two C-decodings of the form $q_1(u')v'q_2$ and $q_1(u)q_2$ with $u = u'v'$.

[1] In contrast to our general convention for this paper, here we include the case of a language being empty.

The difference between these situations becomes apparent in our discussion of relativized solid codes below. Thus, Definition 3.2 applies to any occurrences of u and u', not just to those situations in which u and u' start at the same position in q, and also not just to occurrences of u and u' in the same C-decoding of q. Thus, if u and u' are distinct and occur in any C-decodings of $q \in L$, prefix-admissible for C, then the set $\{u, u'\}$ must be a prefix code.

Similarly, a word $q \in C^+$ is *overlap-admissible* if no two words $u, u' \in C$, not necessarily distinct and appearing in any C-decodings of q, overlap. In particular, if $u \in C$ and u occurs in a C-decoding of q, then u must not overlap itself.

Using the setting of Example 3.1, note that the word $q = a(ba)(ba) = (ab)a(ba) = (ab)(ab)a$ is not prefix-admissible because, taking the first and third C-decodings of q, $u = a$ is a prefix of $u' = ab$. The word q is also not suffix-admissible because, taking the first and second C-decodings of q, $u = a$ is a suffix of $u' = ba$. On the other hand, the word $q' = (ba)(ba)a$ is prefix-admissible, but not suffix-admissible, because a is a suffix of ba. The word $q'' = a(ab)(ab)$ is suffix-admissible, but not prefix-admissible, because a is a prefix of ab.

Definition 3.3. Let C be a subset of Σ^+, let $L \subseteq C^+$ and let P be a predicate on $\mathfrak{P}_{\text{fin}}(C)$. Then C is said to *satisfy P relative to L* if every $q \in L$ is P-admissible for C.

Definition 3.4. When C satisfies P relative to L we say that C is a *P-code relative to L*.

Example 3.3. Consider $C = \{a, ab, ba\}$ as in Example 3.1. The language C is not a prefix code as $a <_{\text{p}} ab$. The set C is a P_{p}-code relative to $L_1 = \{ab, ba\}^+ \cup \{(ba)^n a^m \mid n, m \geq 1\}$. Note that C is not a P_{s}-code relative to L_1. The set C is a P_{s}-code relative to $L_2 = \{a^m(ab)^n \mid m \geq 0, n \geq 1\} \cup \{a^m \mid m \geq 1\}$.

The idea of decoding messages over a set C which is not a code, is not new. Several authors [2,4,6] have used different ideas to weaken the condition on C, so that messages could still be uniquely decodable using C in some sense, even when C is not a code. In the remainder of this section, we describe three such approaches, as follows. The use of coding partitions for decoding messages over a set which is not a code, from [2], is described in Section 3.1. The idea of relativizing solid codes, from [4,8], is discussed in Section 3.2, and that of relativizing comma-free codes, from [6,12], is described in Section 3.3. Moreover, we briefly outline the connection between relativized codes and free subsemigroups of free semigroups in Section 3.4.

3.1. Using coding partitions

In [2], a method to decode messages over C by partitioning the given set C, even when C is not a code, is proposed as follows:

Definition 3.5. Let $F = \{X_1, X_2, \ldots, X_i, \ldots\}$ be a partition of C such that $X_i^+ \cap X_j^+ = \emptyset$ for $i \neq j$. For a word $q \in C^+$ a *decoding over F* is an n-tuple $(z_1, z_2, \ldots, z_n)$ for some n such that the following conditions are met:

1. $q = z_1 z_2 \cdots z_n$;
2. for every $i = 1, 2, \ldots, n$ there is j_i with $z_i \in X_{j_i}^+$;
3. for $i = 1, 2, \ldots, n-1$, with j_i as before, one has $z_{i+1} \notin X_{j_i}^+$.

The set F is called a *coding partition,* if every $q \in C^+$ has a unique decoding over F.

Example 3.4 (*[2]*)**.** Let $\Sigma = \{a, b\}$ and $C = \{aa, aaa, bb, bbb\}$. Let $F = \{X_1, X_2\}$ where $X_1 = \{aa, aaa\}$ and $X_2 = \{bb, bbb\}$. The set F is a partition of C with $X_1^+ \cap X_2^+ = \emptyset$. One computes that

$$C^+ = (aa^+bb^+)^+ \cup (aa^+bb^+)^* aa^+ \cup (bb^+aa^+)^+ \cup (bb^+aa^+)^* bb^+.$$

Therefore, F is a coding partition. As neither X_1 nor X_2 is a code, the set C is not a code either.

This approach differs from our approach in two ways:

1. If F is a coding partition for C, then every word $q \in C^+$ can be uniquely decoded over F, provided a nontrivial partition exists [2]. However such a word q need not have a unique C-decoding. For instance, the word $q = a^2b^5a^3 \in C^+$ with C as in Example 3.4 has a unique decoding over the partition F into $z_1 = a^2$, $z_2 = b^5$ and $z_3 = a^3$, but has two C-decodings $(aa)(bb)(bbb)(aaa)$ and $(aa)(bbb)(bb)(aaa)$. By comparison, as shown in Proposition 4.3, every word that is P_{p}-admissible or P_{s}-admissible for C is uniquely C-decodable.
2. In contrast to our approach, the coding partition method does not always identify the words with unique C-decodings. Indeed, in some cases a nontrivial coding partition may not exist for a given set C, and hence none of the words in C^+ could be considered as possible uniquely decodable messages according to this method. In such cases, words over C that have a unique C-decoding may be left out. Consider the set $C = \{a, ab, ba\}$. According to [2], the only coding partition for this set is the trivial partition, that is, the set C itself. However, as seen in Example 3.1, there exists an infinite language L such that C is a code relative to L, that is, the infinitely many words in L all have unique C-decodings. These words would be identified by our approach as being uniquely C-decodable, since the first and third components of L consist of prefix-admissible words, while the first and the second components of L consist of suffix-admissible words.

3.2. Relative solid codes

In [4] the notion of relativized solid code is defined as follows.

Definition 3.6 (*[4]*). Let C be a subset of Σ^+ and let $L \subset \Sigma^+$. The set C is a *solid code relative to L* if it satisfies the following conditions for all words $q \in L$:

1. if $q = xszty$ with $x, y, s, t \in \Sigma^*$ such that $z, szt \in C$, then $st = \lambda$;
2. if $q = xszty$ with $x, y, s, t \in \Sigma^*$ such that $sz, zt \in C$ and $z \in \Sigma^+$ then $st = \lambda$.

The first condition states that, for $u, v \in C$, if $u <_{\mathrm{i}} v$, then, for all $q \in L$, $v \not\leq_{\mathrm{i}} q$. The second condition states that if $u, v \in C$, and u and v overlap as $u = sz$ and $v = zt$ with $z \in \Sigma^+$, then, for all $q \in L$, $szt \not\leq_{\mathrm{i}} q$.

Definition 3.6 is one possible relativization of the notion of solid code, which is different from the notion of P_{solid}-code relative to a language as introduced in Definition 3.4. Our definition above is based on a different, but equivalent, definition of non-relativized solid codes as overlap-free infix codes.

Note that, if C is a solid code relative to L then C is a P_{i}-code relative to $L \cap C^+$. Indeed, let q in $L \cap C^+$. If $u \in C$ occurs in a C-decoding of q, $v \in C$ and $u <_{\mathrm{i}} v$, then $v \not<_{\mathrm{i}} q$. Hence v does not occur in a C-decoding of q.

For (unrelativized) solid codes there is also a definition based on decompositions of messages (see [9]): let C be a subset of Σ^+ and $q \in \Sigma^+$. A *C-decomposition of q* consists of two sequences $u_0, u_1, \ldots, u_n \in \Sigma^*$ and $v_1, v_2, \ldots, v_n \in C$ for some $n \in \mathbb{N}$, such that $q = u_0 v_1 u_1 v_2 u_2 \cdots v_n u_n$ and $v \not\leq_{\mathrm{i}} u_i$ for all $v \in C$ and $i = 0, 1, \ldots, n$. Every word $q \in \Sigma^+$ has at least one C-decomposition. Note that every C-decomposition of a word in C^+ can be considered as a C-decoding as follows:

$$u_0 = u_1 = \cdots = u_n = \lambda$$

and the C-decoding is

$$(n, (v_1, v_2, \ldots, v_n)).$$

The set C is a solid code if and only if every word in Σ^+ has a unique C-decomposition. In [8], a relativization of the notion of solid code is proposed, based on the uniqueness of C-decompositions, and this notion turns out to be equivalent to the one of Definition 3.6.

Proposition 3.2 (*[8]*). *Let $L \subseteq \Sigma^+$. A language $C \subseteq \Sigma^+$ is a solid code relative to L if and only if every word $q \in L$ has a unique C-decomposition.*

The difference between these equivalent concepts and our approach to relativizing solid codes is illustrated by the following example.

Example 3.5 (*[6]*). Let $\Sigma = \{a, b, c\}$ and $C = \{ab, c, ba\}$. The set C is not overlap-free, hence not a solid code. By Definition 3.6, C is a solid code relative to the language $L = (\{abc\} \bigcup \{cba\})^*$. However, the set C is not a P_{solid}-code relative to L, as $q = abccba \in L$ has the C-decoding $(ab)(c)(c)(ba)$ and is thus not P_{solid}-admissible since $ab <_{\mathrm{ol}} ba$.

3.3. Decoding messages over sets that are not comma-free

Another type of relativization, for sets which are not comma-free, was suggested by Head in [6]. For a given set C that is not comma-free, one decodes the messages over C pretending that C is comma-free. Using coding properties relative to a language [4], a sequence of subsets of C is constructed, which are comma-free. The codes of this sequence are called *join codes.*

Definition 3.7 (*[6]*). A word $w \in \Sigma^*$ is said to be a *join relative to a language L*, if, for some $u, v \in \Sigma^*$ with $uwv \in L$, then both u and v are also in L. A word $w \in C$ is said to be a join in C, if it is a join relative to C^*.

It is shown in [6] that, for a given code C, the set $J(C)$ of its joins is indeed a comma-free code. Similarly, with $C_1 = C \setminus J(C)$, let $J(C_1)$ be the set of joins of C_1. In general, if C_i is not empty, let $J(C_i)$ be the set of joins of C_i and let $C_{i+1} = C_i \setminus J(C_i)$. Thus a sequence of comma-free codes can be obtained from C, itself not comma-free, and messages over C are decoded using the sequence of joins. If C is such that $C = \bigcup_{i \geq 0} J(C_i)$, with $C_0 = C$, then C is called a *split code*.

Example 3.6 (*[6]*). Let $C = \{a, bab, cbabc\}$ be a code over the alphabet $\{a, b, c\}$. Then $J(C) = \{cbabc\}$, $C_1 = C \setminus J(C) = \{a, bab\}$ and $J(C_1) = \{bab\}$ and $C_2 = J(C_2) = \{a\}$ with $C_3 = \emptyset$. One can verify that $J(C)$, $J(C_1)$ and $J(C_2)$ are comma-free and C is indeed a split code. Fig. 1 shows how the word *acbabcbabcbabcaababacbabcabab* in C^+ can be decoded in steps, using joins.

Observe that the set C in Example 3.6 is not comma-free, but any message over C^+ can be decoded using $J(C)$ first and then $J(C_1)$ and hence $J(C_2)$. However, this procedure is not applicable for all sets C because the join of C or one of the sets C_i could be empty. For a set C, every word over C can be decoded using the join codes as if the set C were a comma-free code only if C is a *split code*.

However, even in problematic cases, our approach can provide words in the set C^+ that can be decoded as if the set C were comma-free. For example, consider the set $X = \{abb, aab, aba, bab\}$ over the alphabet $\{a, b\}$. Observe that

1. $ab(bab)b \in X^*$ with $ab, b \notin X$,

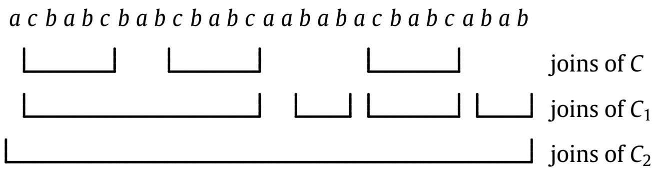

Fig. 1. For $C = \{a, bab, cbabc\}$, the word *acbabcbabcbabcaababacbabcabab* in C^+ can be decoded in steps, using the joins $J(C) = \{cbabc\}$, $J(C_1) = \{bab\}$, and $J(C_2) = \{a\}$, which are comma-free.

2. $a(abb)ab \in X^*$ with $a, ab \notin X$,
3. $ab(aab)b \in X^*$ with $ab, b \notin X$ and
4. $b(aba)ab \in X^*$ with $b, ab \notin X$.

Thus, for all $w \in X$, there exist words of the form $uwv \in X^*$ where neither u nor v belong to X. Therefore, by Definition 3.7, $J(X)$, the join of X, is empty and hence no word over X^+ can be decoded using a comma-free set. However, $q = abbaab \in X^+$ is comma-free-admissible for X, because $q = (abb)(aab)$, and q has a unique X-decoding. Moreover, $q \in C_q^+$, and q has a unique C_q-decoding for $C_q = \{abb, aab\}$, where C_q is a comma-free code. The general method for constructing C_q is given in Proposition 4.5 below. Thus even when the join of a set X is empty, our method could still identify words over X as messages which can be uniquely decoded using a comma-free subset of X.

3.4. *Free subsemigroups of free semigroups*

In this section we summarize related work on free subsemigroups of free semigroups. As a general source we refer to [3]. The literature cited there should be consulted as well.

The set Σ^+ is a free semigroup freely generated by Σ. A subset $C \subseteq \Sigma^+$ is a code, if and only if the semigroup C^+ generated by C is free with C as the free set of generators. If C^+ is not freely generated by C, that is, C is not a code, our relativization considers languages the words of which are uniquely C-decodable. One could turn the question around as follows: given a language $L \subseteq \Sigma^+$, find a language $C \subseteq \Sigma^+$ such that all words in L have a unique C-decoding.

In a more restricted framework this question is studied in [3] to determine properties of the 'minimal' embedding of non-free subsemigroups of Σ^+ into free subsemigroups of Σ^+. Consider a finite set $G \subseteq \Sigma^+$, and let L be the subsemigroup of Σ^+ generated by G. The *free envelope* $H_{\mathrm{f}}(L)$ of L is the smallest free subsemigroup of Σ^+ containing L. The *unique factorization extension* $H_{\mathrm{u}}(L)$ of L is the smallest subsemigroup of Σ^+ in which the elements of L can be factored uniquely.[2] Both $H_{\mathrm{f}}(L)$ and $H_{\mathrm{u}}(L)$ are finitely generated. In general, $H_{\mathrm{u}}(L) \subseteq H_{\mathrm{f}}(L)$.

Now consider minimal sets of generators G_{f} and G_{u} for $H_{\mathrm{f}}(L)$ and $H_{\mathrm{u}}(L)$, respectively. These are unique. If L is not freely generated by G, then $|G_{\mathrm{f}}| < |G|$ and $|G_{\mathrm{u}}| < |G|$.

Either one of G_{f} or G_{u} would take the rôle of the relativized code C in our setting.

4. Properties of relativized codes

In this section we derive several properties of relativized codes and relativized P-codes. We focus on the particular case of predicates emphasized in Section 2: prefix-, suffix-, infix-, outfix-, overlap-free, etc. We show that, for several of these predicates, a hierarchy exists for the corresponding relativized codes, similar to that for classical codes and that, for any given word $q \in \Sigma^+$ that is P-admissible for $C \subseteq \Sigma^+$, one can extract a subset $C_q \subseteq C$ such that $q \in C_q^+$, C_q is a code and C_q is minimal with respect to this property.

Observe that, if C satisfies P relative to L then C satisfies P relative to L' for all $L' \subseteq L$.

The following statement generalizes Proposition 3.1.

Proposition 4.1. *Let $C \subseteq \Sigma^+$, let $P \in \mathfrak{P}_{\mathrm{fin}}(C)$, and let $\mathcal{L}$ be the set of languages in Σ^+ such that C satisfies P relative to L for all $L \in \mathcal{L}$. Let*

$$L_{\max}(C, P) = \{q \mid q \in C^+, q \text{ is } P\text{-admissible for } C\}.$$

The set $\mathcal{L}$ is closed under arbitrary unions and

$$L_{\max}(C) = \bigcup_{L \in \mathcal{L}} L.$$

The set $\mathcal{L}$ is also closed under arbitrary intersections.[3]

Proof. The proof is similar to the proof of Proposition 3.1. □

[2] This means, that they have unique G_{u}-decodings, where G_{u} is the unique minimal set of generators of $H_{\mathrm{u}}(L)$.

[3] As in Proposition 3.1, in this statement a language is permitted to be empty.

Note that, for any word in $L_{\max}(C, P)$, the set C behaves like a code satisfying P. For example, if $P = P_{\mathrm{p}}$ and C is a P_{p}- code relative to L, any message in $L_{\max}(C, P)$ can be decoded as though C were a prefix code.

Proposition 4.2. *Let $C \subseteq \Sigma^+$, and P be a predicate $\mathfrak{P}_{\mathrm{fin}}(C)$, The following are equivalent.*

1. *C satisfies P relative to C^2.*
2. *C satisfies P relative to C^+.*
3. *$L_{\max}(C, P) = C^+$.*

Proof. $(2) \Rightarrow (1)$: If C satisfies P relative to C^+, then C satisfies P also relative to C^2 as $C^2 \subseteq C^+$.

$(1) \Rightarrow (2)$: Consider $q = uv \in C^2$, with $u, v \in C$. By assumption q is P-admissible for C, hence $P(\{u, v\}) = 1$. This holds true for all words $q \in C^2$, hence for all words $u, v \in C$. Therefore, C satisfies P relative to C^+.

$(3) \Rightarrow (2)$: C satisfies P relative to $L_{\max}(C, P)$ which is equal to C^+.

$(2) \Rightarrow (3)$: $L_{\max}(C, P) \subseteq C^+$ by definition and $C^+ \subseteq L_{\max}(C, P)$ by assumption (2). □

We consider closure properties of the set $\mathcal{L}$ of languages relative to which a given language $C \subseteq \Sigma^+$ is a P-code. In addition to union, intersection, concatenation and complement, we also include the mirror image, insertion, deletion and shuffle.

For $w = a_1 a_2 \cdots a_{n-1} a_n$ with $a_1, a_2, \ldots, a_n \in \Sigma$, denote by $\mathrm{mi}(w)$ the *mirror image* of w, that is, $\mathrm{mi}(w) = a_n a_{n-1} \cdots a_2 a_1$. For $L \subseteq \Sigma^*$, $\mathrm{mi}(L) = \{\mathrm{mi}(w) \mid w \in L\}$. For two words $u, v \in \Sigma^*$, the *insertion* of v into u is defined in [11] as

$$u \longleftarrow v = \{u_1 v u_2 \mid u_1 u_2 = u \text{ with } u_1, u_2 \in \Sigma^*\},$$

and the *deletion* of v from u, is defined in [11] as

$$u \longrightarrow v = \{u_1 u_2 \mid u = u_1 v u_2 \text{ with } u_1, u_2 \in \Sigma^*\}.$$

The *shuffle* of u and v is the set

$$u \sqcup\!\sqcup v = \left\{ w \;\middle|\; \begin{array}{l} w \in \Sigma^*, \exists n \in \mathbb{N}\, \exists u_0, u_1, \ldots, u_n \in \Sigma^* \\ \exists v_1, v_2, \ldots, v_n \in \Sigma^* :\ u = u_0 u_1 \cdots u_n, \\ v = v_1 v_2 \cdots v_n,\ w = u_0 v_1 u_1 v_2 u_2 \cdots v_n u_n \end{array} \right\}.$$

Remark 4.1. Let $C \subseteq \Sigma^+$, let P be a predicate $\mathfrak{P}_{\mathrm{fin}}(C)$, and let $\mathcal{L}$ be the set of languages relative to which C is a P-code.

1. $\mathcal{L}$ is closed under arbitrary unions and intersections.
2. $\mathcal{L}$ is not necessarily closed under concatenation.
3. $\mathcal{L}$ is not necessarily closed under complement, mirror image, insertion, deletion or shuffle.

Proof. Statement (1) is proved in Proposition 4.1.

For Statement (2), consider $P = P_{\mathrm{p}}$ and $C = \{a, ab, ba\}$. Let $L_1 = \{babaa\}$ and $L_2 = \{ba\}$. Then $L_1 L_2 = \{babaaba\}$. The word $babaaba$ has a factorization $(ba)(ba)(ab)(a)$. As $a <_{\mathrm{p}} ab$, this word is not P_{p}-admissible for C.

Statement (3) follows from the fact that each of these operations can result in words which are not in C^+. □

As shown in the proof, the non-closure in Statements (2) and (3) is fundamentally different.

In the remainder of this section we consider properties of P-codes relative to a language for the particular cases of predicates defined in Section 2.

We first show that for $\varrho \in \{\mathrm{p}, \mathrm{s}, \mathrm{b}, \mathrm{i}, \mathrm{o}, \mathrm{h}, \mathrm{solid}, \mathrm{inter}_n\}$, if a word $q \in C^+$ is P_ϱ admissible for C, then q has a unique C-decoding.

Proposition 4.3. *Let C be a subset of Σ^+ and let $q \in C^+$. If q is P_{p}-admissible or P_{s}-admissible for C, then q is uniquely C-decodable.*

Proof. Suppose that q has two different C-decodings

$$q = x_1 x_2 \cdots x_n = y_1 y_2 \cdots y_m.$$

There is a smallest k, $1 \le k \le n$, such that $x_k \neq y_k$. For this k, without loss of generality, $x_k <_{\mathrm{p}} y_k$. Hence, $P_{\mathrm{p}}(\{x_k, y_k\})$ is false. By left–right duality there is an l, $1 \le l \le n$, such that $P_{\mathrm{s}}(\{x_l, y_l\})$ is false. Thus, q is neither P_{p}-admissible nor P_{s}-admissible. □

The following Lemma states the general structure of the proof of Proposition 4.3.

Lemma 4.1. *Let P and P' be predicates in $\mathfrak{P}_{\mathrm{fin}}(C)$ with the following properties:*

1. *For all $x, y \in \Sigma^+$, $P(\{x, y\}) = 1$ implies $P'(\{x, y\}) = 1$.*
2. *For all sets $C \subseteq \Sigma^+$ and all $q \in C^+$, P'-admissibility of q for C implies that q is uniquely C-decodable.*

Then, for all sets $C \subseteq \Sigma^+$ and all $q \in C^+$, if q is P-admissible for C then q is uniquely C-decodable.

Corollary 4.1. *For all $\varrho \in \{\mathrm{p}, \mathrm{s}, \mathrm{b}, \mathrm{i}, \mathrm{o}, \mathrm{h}, \mathrm{solid}, \mathrm{inter}_n\}$, for all sets $C \subseteq \Sigma^+$ and for all $q \in C^+$, if q is P_ϱ-admissible for C then q is uniquely C-decodable. Moreover, if C is a P_ϱ-code relative to L then C is a code relative to L.*

Proof. By Lemma 2.1, for every $\varrho \in \{\mathrm{p, s, b, i, o, h, solid, inter}_n\}$, the predicate P_ϱ implies $P_\mathrm{p} \vee P_\mathrm{s}$. The statement thus follows from Proposition 4.3 and Lemma 4.1. □

For $\varrho \in \{\mathrm{p, s, b, i, o, h, solid, inter}_n\}$ and related ones, P_ϱ-admissibility implies unique decodability. The converse implication is not true in general. In Example 3.1, the word $q = (a)(ab)(ab)$ is uniquely C-decodable, but not P_p-admissible for C.

Remark 4.2. If C is a P-code relative to C^+ then $P(\{x, y\}) = 1$ for all $x, y \in C$. In particular, if $\varrho \in \{\mathrm{p, s, b, i, o, h, solid, inter}_n\}$ and C is a P_ϱ-code relative to C^+, then C is a ϱ-code in the usual sense.

Proof. For every $x, y \in C$ there is a word $q \in C^+$ such that both x and y occur in a C-decoding of q. As q is P-admissible for C, one has $P(\{x, y\}) = 1$. For $\varrho \in \{\mathrm{p, s, b, i, o, h, solid, inter}_n\}$, C is a ϱ-code if and only if $P_\varrho(\{x, y\}) = 1$ holds for all $x, y \in C$. □

According to Remark 4.1, concatenations of P-admissible words need not be P-admissible. The following proposition gives a condition under which powers of P-admissible words are P-admissible.

Proposition 4.4. *If $C \subseteq \Sigma^+$ is a code then, for $q \in C^+$, we have that q is P-admissible for C if and only if q^n is P-admissible for C for all $n \geq 2$.*

Proof. Given q is P-admissible for C and let $q^n = xuy = x'u'y'$ such that $x, y, x', y' \in C^*$, $u, u' \in C$. Given that C is a code and hence q^n has a unique C-decoding. Thus $q^n = q^k x_1 u y_1 q^l = q^i x'_1 u' y'_1 q^j$ where $q = x_1 u y_1 = x'_1 u' y'_1$ with $x_1, y_1, x'_1 . y'_1 \in C^*$ and $k + l = i + j = n - 1$. Since q is P-admissible for C, $P(\{u, u'\}) = 1$. Thus q^n is P-admissible for C. For the other implication assume that q^n is P-admissible for C for some $n \geq 2$. Let $q = xuy = x'u'y'$ with $x, y, x', y' \in C^*$ and $u, u' \in C$. Then $q^n = q^{n-1}q = q^{n-1}xuy = qq^{n-1} = x'u'y'q^{n-1}$ with $q^{n-1}x, y, x', y'q^{n-1} \in C^*$ and hence $P(\{u, u'\}) = 1$. Thus q is P-admissible for C. □

Observe that if q is P-admissible for C, then for an arbitrary $w \in \mathrm{Sub}(q) \cap C^+$, where $\mathrm{Sub}(q)$ denotes the set of all subwords of q, w need not be P-admissible for C. Let $C = \{a, ba, ab\}$, $q = bababa$ is P_p-admissible for C but $aba \in \mathrm{Sub}(q) \cap C^+$ is not P_p-admissible for C since $a <_\mathrm{p} ab$. Note that C is not a code. The statement does not hold even if C is a code. For example consider $C_1 = \{aba, ba\}$. The word $q = bababa$ is P_s-admissible for C_1 but $ababa \in \mathrm{Sub}(q) \cap C_1^+$ is not P_s-admissible for C_1 since $\{aba, ba\} \notin P_\mathrm{s}$.

The following observation shows that, for several standard predicates, a hierarchy of the corresponding relativized codes exists, similar to the one for unrelativized codes.

Lemma 4.2. *Let $\varrho_1, \varrho_2 \in \{\mathrm{p, s, b, i, o, h, solid, inter}_n\}$. If C is a P_{ϱ_1}-code relative to L for some language $L \subseteq C^+$ and if $P_{\varrho_1}(\{x, y\}) = 1$ implies $P_{\varrho_2}(\{x, y\}) = 1$ for all $x, y \in \Sigma^*$, then C is a P_{ϱ_2}-code relative to L.*

Proposition 4.5. *For all $\varrho \in \{\mathrm{p, s, b, i, o, h, solid, inter}_n\}$, let $q \in C^+$ be P_ϱ-admissible for C. Then there exists a unique set $C_q \subseteq C$ such that $q \in C_q^+$, C_q is a P_ϱ-code, and C_q is minimal with respect to this property.*

Proof. Since $q \in C^+$ there exist $x_1, x_2, .., x_n \in C$ such that $q = x_1 x_2 \cdots x_n$ and by assumption q is P_ϱ-admissible for C. Let $C_q = \bigcup_{i=1}^n \{x_i\}$ and for all $x_i, x_j \in C_q$, $P_\varrho(\{x_i, x_j\}) = 1$. Thus C_q is a ϱ-code and C_q is minimal by construction. Now we show the uniqueness of C. Suppose there exists an $L = \bigcup_{i=1}^m \{p_i\} \subseteq C$ such that $q = p_1 p_2 \cdots p_m$. Then $q = x_1 x_2 \cdots x_n = p_1 p_2 \cdots p_m$ and since q is P_ϱ-admissible for C, by Corollary 4.1, q has a unique C-decoding. Thus $x_i = p_i$ for all i and $m = n$ and hence $L = C_q$. □

Thus for $\varrho \in \{\mathrm{p, s, b, i, o, h, solid, inter}_n\}$, if a word q is P_ϱ-admissible for C, then there exists a minimal set $C_q \subseteq C$ such that C_q is a ϱ-code and $q \in C_q^+$. For other words in C_q^+, however, the situation is different. If $w \in C_q^+$ such that $w \neq q$, then w will be P_ϱ-admissible for C_q, but it need not be P_ϱ-admissible for C. For example, let $C = \{a, ab, ba\}$ and let $q = babaa$ be a P_p-admissible word for C. Then $C_q = \{ba, a\}$ and C_q is a prefix code, i.e., for all $w \in C_q^+$, w is P_p-admissible for C_q. However, a word $w \in C_q^+$ need not be P_p-admissible for C. Indeed, the word $w = ababa = (a)(ba)(ba)$ has a C_q-decoding and is P_p-admissible for C_q, but $w = (a)(ba)(ba) = (ab)(ab)(a)$ has two different decodings over C and also $a <_\mathrm{p} ab$. Hence w is not P_p-admissible for C. In the following corollary we show that the words over C_q will be P_ϱ-admissible for C if we impose an additional constraint on C.

Corollary 4.2. *Let $q \in C^+$ be P-admissible for C. If C is a code, then there exists a unique set $C_q \subseteq C$ as follows:*

1. *$q \in C_q^+$, C_q satisfies P relative to C_q^+, and C_q is the minimal set with respect to these properties;*
2. *for all $w \in C_q^+$, w is P-admissible for C.*

As shown in Remark 4.1, the concatenation of two words, each P-admissible for C, is not always P-admissible for C. To obtain closure under concatenation, we need an additional constraint. Under this constraint, we now state a necessary and sufficient condition for the concatenation of two P-admissible words to be P-admissible.

Proposition 4.6. *Let $C \subseteq \Sigma^+$ be a code and let $q, y \in C^+$ be P-admissible for C. Let C_q and C_y be given as in Corollary 4.2. Every word $z \in \{q, y\}^+$ is P-admissible for C, if and only if $C_q \cup C_y$ satisfies P relative to $(C_q \cup C_y)^+$.*

Proof. The implication "$\Longleftarrow$" is straightforward. Since $C_q \cup C_y$ satisfies P relative to $(C_q \cup C_y)^+$ and C is a code, for all $z \in \{q, y\}^+$, z has a unique C-decoding and hence also a unique decoding over $C_q \cup C_y$. Thus z is P-admissible for C.

For the implication "$\Longrightarrow$", suppose that every $z \in \{q, y\}^+$ is P-admissible for C. Let $w \in (C_q \cup C_y)^+$ such that $w = \alpha u \beta = \alpha' u' \beta'$ with $\alpha, \beta, \alpha', \beta' \in C^*$ and $u, u' \in C$. Observe that, since C is a code, w has a unique C-decoding and hence $\alpha, \beta, \alpha', \beta' \in (C_q \cup C_y)^*$ and $u, u' \in (C_q \cup C_y)$. Then we have the following cases: (1) $u, u' \in C_q$, (2) $u, u' \in C_y$ and (3) $u \in C_q$, $u' \in C_y$. The first two cases imply that $P(\{u, u'\}) = 1$ since both C_q and C_y satisfy P relative to C_q^+ and C_y^+, respectively. Suppose for $u \in C_q$, $u' \in C_y$ and $P(\{u, u'\}) = 0$. Then for $z = qy \in \{q, y\}^+$ one has $z = suty = qs'u't'$ where $q = sut$, $y = s'u't'$ with $s, t, s', t' \in C^*$. Hence $z = qy$ is not P-admissible for C, a contradiction. □

In the following, we provide a necessary and sufficient condition for all words $w \in C$ to be P_ϱ-admissible for C where $\varrho \in \{\mathrm{p, s, b, i, o, h, solid, inter}_n\}$.

Proposition 4.7. *For all $\varrho \in \{\mathrm{p, s, b, i, o, h, solid, inter}_n\}$, $C \subseteq L_{\max}(C, P_\varrho)$ if and only if $C \cap C^n = \emptyset$ for all $n \geq 2$.*

Proof. Assume $C \subseteq L_{\max}(C, P_\varrho)$. Suppose there exists a $q \in C$ such that $q = x_1 x_2 \cdots x_n$ for some $n \geq 2$ and $x_1, x_2, \ldots, x_n \in C$. Then $x_1 <_{\mathrm{p}} q$ and $x_n <_{\mathrm{s}} q$. By Lemma 2.1, for every $\varrho \in \{\mathrm{p, s, b, i, o, h, solid, inter}_n\}$, the predicate P_ϱ implies $P_{\mathrm{p}} \vee P_{\mathrm{s}}$. Thus for $\varrho \in \{\mathrm{p, s, b, i, o, h, solid, inter}_n\}$, $P_\varrho(\{x_i, q\}) = 0$ for some i, a contradiction since q is P_ϱ-admissible for C. Hence $C \cap C^n = \emptyset$ for all $n \geq 2$. Conversely let $C \cap C^n = \emptyset$ for all $n \geq 2$. Let $q \in C$ such that $q \notin L_{\max}(C, P_\varrho)$, then q is not P_ϱ-admissible for C which implies that $q = xuy = x'u'y'$ with $x, x', y, y' \in C^*$, $u, u' \in C$ and $P_\varrho(\{u, u'\}) = 0$. Thus either $u <_\varrho u'$ or $u' <_\varrho u$ which implies that $u \neq u'$ and hence $q \in C^n$ for some $n \geq 2$ which is a contradiction. Therefore $C \subseteq L_{\max}(C, P_\varrho)$. □

Proposition 4.8. *Let C_1 and C_2 be such that $C_1 \cap C_1^n = \emptyset$ and $C_2 \cap C_2^n = \emptyset$ for all $n \geq 2$. Then, for all $\varrho \in \{\mathrm{p, s, b, i, o, h, solid, inter}_n\}$, $C_1 = C_2$ if and only if $L_{\max}(C_1, P_\varrho) = L_{\max}(C_2, P_\varrho)$.*

Proof. Obviously, if $C_1 = C_2$, then $L_{\max}(C_1, P_\varrho) = L_{\max}(C_2, P_\varrho)$. Conversely, let $L_{\max}(C_1, P_\varrho) = L_{\max}(C_2, P_\varrho)$. We show that $C_1 \subseteq C_2$. Let $w \in C_1$. If $w \notin C_2$, then $w \in C_2^+$ since, by Proposition 4.7, $C_1 \subseteq L_{\max}(C_1, P_\varrho) = L_{\max}(C_2, P_\varrho)$ and hence $w = y_1 y_2 \cdots y_n$ for $y_i \in C_2$ for all $1 \leq i \leq n$. Given that $C_2 \cap C_2^n = \emptyset$ for all $n \geq 2$, by Proposition 4.7 we have $C_2 \subseteq L_{\max}(C_2, P_\varrho) = L_{\max}(C_1, P_\varrho)$, $y_i \in L_{\max}(C_1, P_\varrho)$ for all $1 \leq i \leq n$. Then either $y_i \in C_1$ or $y_i \in C_1^+$. In both cases, $w \in C_1^m$ for some $m \geq 2$, a contradiction. Hence $w \in C_2$. Similarly one shows that $C_2 \subseteq C_1$. □

According to Remark 4.1, if a word w is P-admissible for C for some predicate P in $\mathfrak{P}_{\mathrm{fin}}(C)$, then its mirror image need not be P-admissible for C. As stated there, this is mainly due to the fact that C^+ need not be closed under mirror images. The following lemma can serve as a pattern for closure results which go beyond Remark 4.1.

Proposition 4.9. *Let $C \subseteq \Sigma^*$ be such that $\mathrm{mi}(C) = C$. Then,*

1. *C is P_{p}-code relative L if and only if C is P_{s}-code relative to $\mathrm{mi}(L)$.*
2. *C is P_ϱ-code relative L if and only if C is P_ϱ-code relative to $\mathrm{mi}(L)$ for all $\varrho \in \{\mathrm{b, i, o, ol, solid, inter_n}\}$.*

Proof. (1) Suppose that C is P_{p}-code relative to L. Then, for all $w \in L$, w is P_{p}-admissible for C. Suppose that $\mathrm{mi}(w)$ is not P_{s}-admissible for C. Then there exist $\alpha, \beta, p, q \in C^*$ and $u, v \in C$ such that $\mathrm{mi}(w) = \alpha u \beta = pvq$ and $P_{\mathrm{s}}(\{u, v\}) = 0$. This implies that

$$w = \mathrm{mi}(\beta)\mathrm{mi}(u)\mathrm{mi}(\alpha) = \mathrm{mi}(q)\mathrm{mi}(v)\mathrm{mi}(p)$$

with

$$\mathrm{mi}(\beta), \mathrm{mi}(\alpha), \mathrm{mi}(q), \mathrm{mi}(p) \in C^*$$

and

$$\mathrm{mi}(u), \mathrm{mi}(v) \in C.$$

Since $P_{\mathrm{s}}(\{u, v\}) = 0$ also $P_{\mathrm{p}}(\{\mathrm{mi}(u), \mathrm{mi}(v)\}) = 0$, a contradiction. The converse follows by left–right duality.

(2) For all $u, v \in \Sigma^*$, $P_\varrho(\{u, v\}) = 1$ if and only if $P_\varrho(\{\mathrm{mi}(u), \mathrm{mi}(v)\}) = 1$. Thus $\mathrm{mi}(w)$ is P_ϱ-admissible for C, if and only if w is P_ϱ-admissible for C. □

5. Conclusions

In this paper we propose a uniform approach to relativizing the notion of code classes to given message spaces, in order to deal with situations when the set C of codewords is not actually a code in the given class, but nevertheless all words in the message space can be decoded uniquely according to the requirements of the respective class of codes. We derive several basic properties of such relativized codes.

Acknowledgements

The research reported in this paper was supported by a Canada Research Chair Award to Lila Kari and grants from the Natural Sciences and Engineering Research Council of Canada.

The authors thank the referees for their very careful and in-depth recommendations.

References

[1] J. Berstel, D. Perrin, C. Reutenauer, Codes and Automata, in: Encyclopedia of Mathematics and Its Applications, vol. 129, Cambridge University Press, Camridge, 2010.
[2] F. Burderi, A. Restivo, Coding partitions, Discrete Math. Theor. Comput. Sci. (DMTCS) 9 (2) (2007) 227–240.
[3] T. Harju, J. Karhumäki, On the defect theorem and simplifiability, Semigroup Forum 33 (1986) 199–217.
[4] T. Head, Unique decipherability relative to a language, Tamkang J. Math. 11 (1980) 59–66.
[5] T. Head, Deciding the immutability of regular codes and languages und finite transductions, Info. Processing Letters 31 (1989) 239–241.
[6] T. Head, Relativized code concepts and multi-tube DNA dictionaries, in: C. S. Calude, Gh. Păun (Eds.), Finite versus Infinite – Contributions to an Eternal Dilemma, Springer-Verlag, London, 2000, pp. 175–186.
[7] T. Head, Draft of notes for Form. Lang. Sem. Thurs. Sept. 12, 2002, 3 pp. Personal Communication.
[8] H. Jürgensen, Markers and deterministic acceptors for non-deterministic languages, J. of Automata, Languages and Combinatorics 14 (2009) 33–62.
[9] H. Jürgensen, S. Konstantinidis, Codes, in: G. Rozenberg, A. Salomaa (Eds.), in: Handbook of Formal Languages, vol. 1, Springer-Verlag, Berlin, 1997, pp. 511–607.
[10] H. Jürgensen, S. S. Yu, Relations on free monoids, their independent sets, and codes, Internat. J. Comput. Math. 40 (1991) 17–46.
[11] L. Kari, On insertion and deletion in formal languages, Ph.D. Thesis, University of Turku, 1991.
[12] K. Mahalingam, Involution codes: with application to DNA strand design, Ph.D. Thesis, University of South Florida, 2004.
[13] H.J. Shyr, Free Monoids and Languages, third ed., Hon Min Book Company, Taichung, 2001.
[14] H.J. Shyr, G. Thierrin, Codes and binary relations, in: M.P. Malliavin (Ed.), Séminaire d'algèbre Paul Dubreil, Paris 1975–1976, (29ème Année), in: Lecture Notes in Computer Science, vol. 586, Springer-Verlag, Berlin, 1977, pp. 180–188.
[15] S.-S. Yu, Languages and Codes, Tsang Hai Book Publishing Co, Taichung, Taiwan, 2005.

Please cite this article in press as: M. Daley, et al., Relativized codes, Theoretical Computer Science (2012), doi:10.1016/j.tcs.2011.12.024