Fundamenta Informaticae 53(2), 2002, 185-198

Characterization Results for Time-Varying Codes

F.L. Ţiplea [a] E. Mäkinen [b] D. Trincă [a]
C. Enea [a]

a) Faculty of Computer Science
"Al.I.Cuza" University of Iaşi
6600 Iaşi, Romania
E-mail: fltiplea,dragost,cenea@infoiasi.ro

b) Department of Computer and Information Sciences
University of Tampere
P.O. Box 607, FIN-33014
Tampere, Finland
E-mail: em@cs.uta.fi

Abstract

Time-varying codes associate variable length code words to letters being encoded depending on their positions in the input string. These codes have been introduced in [8] as a proper extension of L-codes.

This paper is devoted to a further study of time-varying codes. First, we show that adaptive Huffman encodings are special cases of encodings by time-varying codes. Then, we focus on three kinds of characterization results: characterization results based on decompositions over families of sets of words, a Schützenberger like criterion, and a Sardinas-Patterson like characterization theorem. All of them extend the corresponding characterization results known for classical variable length codes.

1 Introduction and Preliminaries

Originated in the Shannon's information theory in the 1950s, the *theory of codes* has developed in several directions. Among them, the theory of *variable length codes*, strongly related to combinatorics on words, automata theory, formal languages, and the theory of semigroups, has produced a number of beautiful results applicable to various fields. Intuitively, a variable length code is a set of words such that any product of these words can be uniquely "decoded".

In this paper we deal with *time-varying codes*, a special class of variable length codes. These codes have been introduced in [8] as a proper extension of L-codes [4]. The connection with gsm-codes and SE-codes has been also discussed in [8]. With a time-varying code, the code word associated to a letter depends on the position where the letter occurs in the input string. Adaptive Huffman encodings are special cases of encodings by time-varying codes (Section 2). In Section 3 we focus on characterization results for time-varying codes. First, we discuss decompositions over families of sets of words, and characterize them by power-series. Then, two main characterization results are presented: a Schützenberger like criterion and a Sardinas-Patterson like characterization theorem for time-varying codes. All these results generalize the ones for classical codes.

In the rest of this section we recall a few basic concepts and notations on formal languages and codes (for further details the reader is referred to [1, 5, 6]).

$A \subseteq B$ denotes the *inclusion* of A into B, and $|A|$ is the *cardinality* of the set A; the *empty set* is denoted by $\emptyset$. The set of natural numbers is denoted by $\mathbf{N}$. By $\mathbf{N}^*$ we denote the set $\mathbf{N} - \{0\}$.

An *alphabet* is any nonempty set. For an alphabet Δ, Δ^* is the free monoid generated by Δ under the *catenation operation*, and λ is its unity (the *empty word*). As usual, Δ^+ stands for $\Delta^* - \{\lambda\}$. For a set $X \subseteq \Delta^+$, X^+ is the set $\bigcup_{n\geq 1} X^n$, where $X^1 = X$ and $X^{n+1} = X^n X$, for all $n \geq 1$ (for two sets of words A and B, AB is the set of all words uv, where $u \in A$ and $v \in B$). Given an word w and a set X of words, by wX (Xw, resp.) we denote the set of all words wx (xw, resp.), where $x \in X$.

Let X be a nonempty subset of Δ^+, and $w \in \Delta^+$. A *decomposition of w over X* is any sequence of words $u_1, \ldots, u_s \in X$ such that $w = u_1 \cdots u_s$. A *code* over Δ is any nonempty subset $C \subseteq \Delta^+$ such that each word $w \in \Delta^+$ has at most one decomposition over C. Alternatively, one can say that C is a code over Δ if there is an alphabet Σ and a function $h : \Sigma \to \Delta^+$ such that $f(\Sigma) = C$ and the unique homomorphic extension $\bar{h} : \Sigma^* \to \Delta^*$ of h defined by $\bar{h}(\lambda) = \lambda$ and

$$\bar{h}(\sigma_1 \cdots \sigma_n) = h(\sigma_1) \cdots h(\sigma_n),$$

for all $\sigma_1 \cdots \sigma_n \in \Sigma^+$, is injective. When a code is given by a function f as above we say that $\sigma \in \Sigma$ is encoded by $f(\sigma)$.

A *prefix code* is any code C such that no word in C is a proper prefix of another word in C.

A useful graphic representation of a finite code $C \subseteq \Delta^+$ consists of a tree with vertices labelled by symbols in Δ such that the code words are exactly the sequences of labels collected from the root to leaves. For example, the tree in Figure 1 is the graphic representation of the prefix code $\{01, 110, 101\}$, where a is encoded by 01, b by 101, and c by 110.

2 Time-Varying Codes

Time-Varying Codes (TV-codes, for short) have been introduced in [8] as a

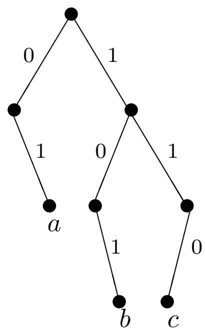

Figure 1: Tree representation of the code $\{01, 101, 110\}$

proper extension of L-codes [4]. The connection to gsm-codes and SE-codes has been also discussed in [8]. In this section we recall the concept of a TV-code and show that adaptive Huffman encodings are special cases of encodings by TV-codes.

A TV-code over an alphabet Δ is any function $h : \Sigma \times \mathbf{N}^* \to \Delta^+$, where Σ is an alphabet, such that the function $\bar{h} : \Sigma^* \to \Delta^*$ given by $\bar{h}(\lambda) = \lambda$ and

$$\bar{h}(\sigma_1 \cdots \sigma_n) = h(\sigma_1, 1) \cdots h(\sigma_n, n),$$

for all $\sigma_1 \cdots \sigma_n \in \Sigma^+$, is injective. Hence, an input string w over Σ is encoded, by a TV-code, letter by letter, taking into consideration the letters' positions in w. That is, if $\sigma \in \Sigma$ occurs on a position i in w, then the occurrence is encoded by $h(\sigma, i)$.

Let $h : \Sigma \times \mathbf{N}^* \to \Delta^+$ be a TV-code. We denote by H_i the set $H_i = \{h(\sigma, i) | \sigma \in \Sigma\}$, for all $i \geq 1$, and by H the family $H = (H_i | i \geq 1)$. The set H_i will be called the i^{th} *section* of h. When Σ is at most countable we can view h as an infinite matrix

$\Sigma \setminus \mathbf{N}^*$	1	2	3	$\cdots$
σ_1	$h(\sigma_1, 1)$	$h(\sigma_1, 2)$	$h(\sigma_1, 3)$	$\cdots$
σ_2	$h(\sigma_2, 1)$	$h(\sigma_2, 2)$	$h(\sigma_2, 3)$	$\cdots$
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$

where $\sigma_1, \sigma_2, \ldots$ is an arbitrary total ordering of the elements of Σ.

In order to show that the adaptive Huffman encodings are special cases of encodings by TV-codes, we recall first the concept of a Huffman code [7].

Let Σ be a finite alphabet. A *Huffman code* for Σ w.r.t. a probability distribution $(p_\sigma | \sigma \in \Sigma)$ on Σ is any prefix code $h : \Sigma \to \{0, 1\}^+$ such that its average length is minimised. A tree associated to a Huffman code is usually called a *Huffman tree.*

The design of a Huffman encoding for an input $w \in \Sigma^+$ requires two steps:

- determine the frequency of occurrences of each letter σ in w;

- design a Huffman code for Σ w.r.t. the probability distribution

$$p_\sigma = \frac{\text{frequency of } \sigma \text{ in } w}{|w|}.$$

  Then, encode w by this Huffman code.

Because this procedure requires two parsings of the input, it is time-consuming for large inputs (although the compression rate by such an encoding is optimal). In practice, an alternative method which requires only one parsing of the input is used. It is called the *adaptive Huffman encoding*. The encoding of an input w by adaptive Huffman is based on the construction of a sequence of Huffman trees as follows (for details the reader is referred to [7]):

- start initially with a Huffman tree A_0 associated to the alphabet Σ (each symbol of Σ has frequency 1);

- if A_n is the current Huffman tree and the current input symbol is σ (that is, $w = u\sigma v$ and u has been already processed), then output the code of σ in A_n (this code is denoted by $code(\sigma, A_n)$) and update the tree A_n getting a new tree A_{n+1} as follows:

  – increment the frequency of σ;

  – apply the *sibling transformation*.

The sibling transformation consists of:

1. compare σ to its successors in the tree (from left to right and from bottom to top). If the immediate successor has frequency $k+1$ or greater, the nodes are still in sorted order and there is no need to change anything. Otherwise, σ should be swapped with the last successor which has frequency k or smaller (except that σ should not be swapped with its parent);

2. increment the frequency of σ (from k to $k+1$);

3. if σ is the root, the loop halts; otherwise, the loop repeats with the parent of σ.

The adaptive Huffman encoding of w defines naturally a TV-code by

$$h_w(\sigma, i) = code(\sigma, A_{i-1}),$$

for all $\sigma \in \Sigma$, $i \geq 1$, and taking $A_i = A_{|w|}$, for all $i > |w|$.

It is quite easy to prove that h_w is a TV-code. Indeed, let us consider two distinct words over Σ, $u\sigma u'$ and $u\sigma' u''$, where $\sigma, \sigma' \in \Sigma$, $u, u', u'' \in \Sigma^*$, and $\sigma \neq \sigma'$. Then,

$$\bar{h}_w(u\sigma u') = \bar{h}_w(u) h_w(\sigma, |u|+1) \bar{h}_w(u') = \bar{h}_w(u) code(\sigma, A_{|u|}) \bar{h}_w(u')$$

and

$$\bar{h}_w(u\sigma' u'') = \bar{h}_w(u)h_w(\sigma', |u|+1)\bar{h}_w(u'') = \bar{h}_w(u)code(\sigma', A_{|u|})\bar{h}_w(u'').$$

Because A_i is a prefix code for any i, it follows that neither $code(\sigma, A_{|u|})$ is a prefix of $code(\sigma', A_{|u|})$ nor $code(\sigma', A_{|u|})$ is a prefix of $code(\sigma, A_{|u|})$. Therefore, $\bar{h}_w(u\sigma u') \neq \bar{h}_w(u\sigma' u'')$, proving that h_w is a TV-code.

We have proved the following.

Proposition 2.1 Adaptive Huffman encodings are special cases of encodings by TV-codes.

Example 2.1 In Figure 2, the sequence of Huffman trees needed to encode the string *dcd* over the alphabet $\{a, b, c, d\}$, is given. The first Huffman tree A_0 is associated to the alphabet. When the first letter d of the input string is read, it is encoded by $code(A_0, d)$, and a new Huffman tree A_1 is generated. This procedure is iterated until the last letter of the input string is processed. The

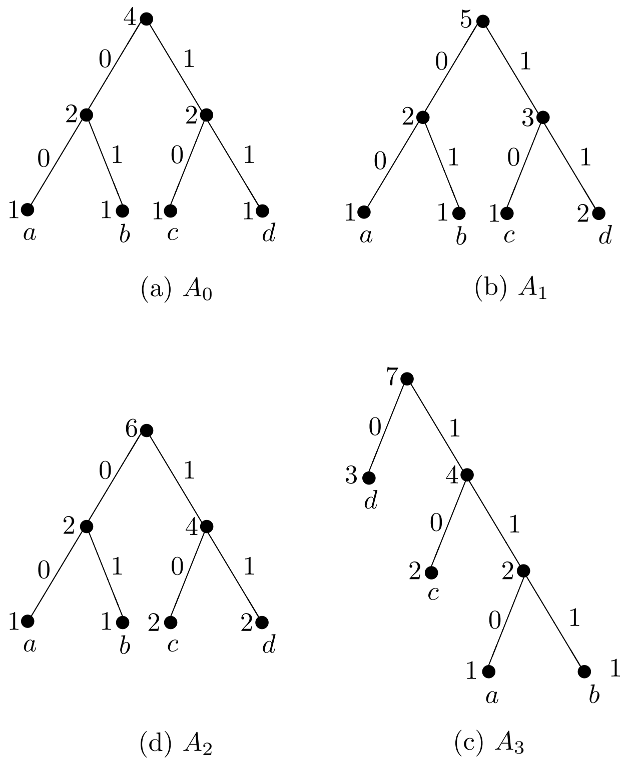

Figure 2: Huffman trees

TV-code induced by this adaptive Huffman encoding in given in the table below.

$\Sigma \setminus \mathbf{N}^*$	1	2	3	4	5	$\cdots$
a	00	00	00	110	110	$\cdots$
b	01	01	01	111	111	$\cdots$
c	10	10	10	10	10	
d	11	11	11	0	0	$\cdots$

3 Characterization Results

The aim of this section is to present several characterization results for TV-codes. First, we characterize the TV-code property by means of decompositions over families of sets of words, and then, a Schützenberger criterion and a Sardinas-Patterson characterization theorem are presented. All these results extend the corresponding characterization results known for classical codes.

In order to avoid trivial but annoying analysis cases, the alphabet Σ is assumed to be of cardinality at least 2 throughout this section.

3.1 Decompositions over Families of Sets of Words

Let $A = (A_i|i \geq 1)$ be a family of subsets of Δ^+. Denote by $A^{\geq i}$ the set $A^{\geq i} = \bigcup_{j \geq i} A_i \cdots A_j$. A *decomposition of a word* $w \in \Delta^+$ *over the family* A is any sequence of words $u_1, \ldots, u_k$ such that $w = u_1 \cdots u_k$ and $u_i \in A_i$, for all $1 \leq i \leq k$.

Definition 3.1 A function $h : \Sigma \times \mathbf{N}^* \to \Delta^+$ is called *regular* or *injective on sections* if the following property holds true:

$$(\forall i \geq 1)(\forall \sigma, \sigma' \in \Sigma)(\sigma \neq \sigma' \ \Rightarrow \ h(\sigma, i) \neq h(\sigma', i)).$$

h is called *regular of base* C, where C is a nonempty subset of Δ^+, if it is regular and $H_i = C$, for all $i \geq 1$.

When Σ is finite, h is regular iff $|H_i| = |\Sigma|$, for all $i \geq 1$.

Remark 3.1 Any TV-code h must be regular. Indeed, if we assume that h is a TV-code and there are $\sigma, \sigma' \in \Sigma$ such that $\sigma \neq \sigma'$ and $h(\sigma, i) = h(\sigma', i)$ for some $i \geq 1$, then

$$\underbrace{\sigma \cdots \sigma\sigma}_{i \text{ times}} \neq \underbrace{\sigma \cdots \sigma\sigma'}_{i \text{ times}}$$

and $\bar{h}(\sigma \cdots \sigma\sigma) = \bar{h}(\sigma \cdots \sigma\sigma')$. Hence, h is not a TV-code; a contradiction.

Directly from definitions we have:

Proposition 3.1 A function $h : \Sigma \times \mathbf{N}^* \to \Delta^+$ is a TV-code iff it is regular and any word $w \in \Delta^+$ has at most one decomposition over its sections $H = (H_i|i \geq 1)$.

Remark 3.2 Without the regularity requirement, the statement in Proposition 3.1 may be false. For example, the function h given in the table below

$\Sigma \setminus \mathbf{N}^*$	1	2	3	$\cdots$
σ_1	a	c	c	$\cdots$
σ_2	a	d	d	$\cdots$

is not regular but any word $w \in \Delta^+$ has at most one decomposition over its family of sections. Moreover, h is not a TV-code because $\bar{h}(\sigma_1) = \bar{h}(\sigma_2)$.

Corollary 3.1 A regular function $h : \Sigma \times \mathbf{N}^* \to \Delta^+$ of base C is a TV-code iff C is a code over Δ.

Proof Let $h : \Sigma \times \mathbf{N}^* \to \Delta^+$ be a regular function of base C. Then, h is a TV-code iff any word $w \in \Delta^+$ has at most one decomposition over $H = (H_i | i \geq 1)$ iff any word $w \in \Delta^+$ has at most one decomposition over C iff C is a code over Δ. $\square$

Remark 3.3 (1) A section of a TV-code is not necessarily a code. Indeed, let us consider the function h given in the table below.

$\Sigma \setminus \mathbf{N}^*$	1	2	3	$\cdots$
σ_1	a	c	c	$\cdots$
σ_2	ab	d	d	$\cdots$
σ_3	ba	e	e	$\cdots$

It is easily seen that h is a TV-code, but H_1 is not a code.

(2) If all sections of a regular function $h : \Sigma \times \mathbf{N}^* \to \Delta^+$ are prefix codes, then h is a TV-code. The proof of this fact is similar to the one in Section 2 showing that adaptive Huffman encodings are special cases of encodings by TV-codes.

The code property of all sections of h does not alone guarantee the TV-code property of h. Indeed, the function h below is regular and each H_i is a code. However, h is not a TV-code because $\bar{h}(\sigma_1\sigma_1) = \bar{h}(\sigma_2\sigma_2)$.

$\Sigma \setminus \mathbf{N}^*$	1	2	3	4	$\cdots$
σ_1	a	ba	c	c	$\cdots$
σ_2	ab	a	d	d	$\cdots$

The unique decomposition property over a family of sets of words can be characterized by formal power-series (for details on semirings and formal power-series the reader is referred to [2, 3]).

Let Δ be a set and R a semiring. A (formal) *power-series* over R with indeterminates in Δ is defined as a function $f : \Delta^* \to R$. An alternative notation for f is $\sum_{w \in \Delta^*} f(w)w$. Let $R[[A]]$ be the set of all power-series over R with indeterminates in Δ.

The *sum* and *product* of two power-series $f, g \in R[[\Delta]]$ are defined by

$$(f + g)(w) = f(w) + g(w),$$

$$(f \cdot g)(w) = \sum_{uv=w} f(u)g(v),$$

for all $w \in \Delta^*$ (the symbol "$\sum_{uv=w}$" in the right hand side of the second equation indicates summation in the semiring R, over all factorizations uv of w, and $f(u)g(v)$ is the product of $f(u)$ and $g(v)$ in R). These operations are associative and $R[[\Delta]]$ under them forms a semiring.

In what follows we work only with power-series over the semiring $\mathbf{N}$ of natural numbers with addition and multiplication. Let Δ be an alphabet and $X \subseteq \Delta^*$. The *characteristic power-series* of X, denoted χ_X, is defined by $\chi_X(w) = 1$ if $w \in X$, and $\chi_X(w) = 0$, otherwise.

Let $A_1, \ldots, A_n$ be subsets of Δ^+, where $n \geq 2$. The product $A_1 \cdots A_n$ is called *unambiguous* if any word $w \in A_1 \cdots A_n$ has only one decomposition $w = u_1 \cdots u_n$ with $u_i \in A_i$, for all $1 \leq i \leq n$.

The unambiguity property can be easily characterized by power-series as follows [1]: the product $A_1 \cdots A_n$ is *unambiguous* iff

$$\chi_{A_1 \cdots A_n} = \chi_{A_1} \cdots \chi_{A_n}.$$

Let $h : \Sigma \times \mathbf{N}^* \to \Delta^+$ be a function. The family $(\chi_{H_1} \cdots \chi_{H_i} | i \geq 1)$ of formal power-series is *locally finite*, that is, the following property holds true

$$(\forall w \in \Delta^*)(|\{i \geq 1 | (\chi_{H_1} \cdots \chi_{H_i})(w) \neq 0\}| \in \mathbf{N}).$$

Therefore, a power-series $\sum_{i \geq 1} \chi_{H_1} \cdots \chi_{H_i}$ can be defined by

$$(\sum_{i \geq 1} \chi_{H_1} \cdots \chi_{H_i})(w) = \sum_{\{i \geq 1 | (\chi_{H_1} \cdots \chi_{H_i})(w) \neq 0\}} (\chi_{H_1} \cdots \chi_{H_i})(w),$$

for all $w \in \Delta^*$ (the right hand side of the equality is a finite sum of natural numbers).

Proposition 3.2 Let $h : \Sigma \times \mathbf{N}^* \to \Delta^+$ be a regular function. h is a TV-code iff $\chi_{H^{\geq 1}} = \sum_{i \geq 1} \chi_{H_1} \cdots \chi_{H_i}$.

Proof Assume that h is a TV-code. Then, the following two properties hold true:

- for any $i \geq 1$, the product $H_1 \cdots H_i$ is unambiguous;
- for any $w \in \Delta^*$, if $w \in H^{\geq 1}$ then there is exactly one $i \geq 1$ such that $w \in H_1 \cdots H_i$.

These two properties lead to the following equivalences:

$$\begin{array}{rcl}\chi_{H^{\geq 1}}(w) = 1 & \Leftrightarrow & w \in H^{\geq 1} \\ & \Leftrightarrow & w \in H_1 \cdots H_i, \text{ for exactly one } i \geq 1 \\ & \Leftrightarrow & \chi_{H_1 \cdots H_i}(w) = 1, \text{ for exactly one } i \geq 1 \\ & \Leftrightarrow & (\chi_{H_1} \cdots \chi_{H_i})(w) = 1, \text{ for exactly one } i \geq 1 \\ & \Leftrightarrow & (\sum_{i \geq 1} \chi_{H_1} \cdots \chi_{H_i})(w) = 1,\end{array}$$

for any $w \in \Delta^*$.

Conversely, assume that $\chi_{H^{\geq 1}} = \sum_{i\geq 1} \chi_{H_1} \cdots \chi_{H_i}$, but h is not a TV-code. Then, there is a word $w \in \Delta^+$ having at least two distinct decompositions over H. That is, there are $i \geq 1$ and $j \geq 1$ such that $w \in H_1 \cdots H_i \cap H_1 \cdots H_j$ (in the case $i = j$ we assume that w has two distinct decompositions over $H_1 \cdots H_i$). Then:

– $(\chi_{H_1} \cdots \chi_{H_i})(w) \geq 1$ and $(\chi_{H_1} \cdots \chi_{H_j})(w) \geq 1$, if $j \neq i$, and

– $(\chi_{H_1} \cdots \chi_{H_i})(w) \geq 2$, if $j = i$.

Therefore, $(\sum_{i\geq 1} \chi_{H_1} \cdots \chi_{H_i})(w) \geq 2 > 1 = \chi_{H^{\geq 1}}(w)$; a contradiction. □

Remark 3.4 Let $h : \Sigma \times \mathbf{N}^* \to \Delta^+$ be a regular function of base C. By Corollary 3.1 and Proposition 3.1 we obtain that C is a code iff $\chi_{C^+} = \sum_{i\geq 1}(\chi_C)^i$. This is a well-known characterization result for classical codes [3].

3.2 Schützenberger Criterion for TV-Codes

Definition 3.2 A family $A = (A_i | i \geq 1)$ of subsets of Δ^+ is called *catenatively independent* if the following property holds

$$(\forall i \geq 1)(\forall j > i)(A_i \cap A_i \cdots A_j = \emptyset).$$

Now, we can prove the following characterization result for TV-codes.

Theorem 3.1 (Schützenberger criterion for TV-codes)
A function $h : \Sigma \times \mathbf{N}^* \to \Delta^+$ is a TV-code iff the following properties hold:

(1) h is regular;

(2) H is catenatively independent;

(3) for any $w \in \Delta^+$ and $i, j \geq 1$, if $w \notin H^{\geq i+1}$ then $H_1 \cdots H_i w \cap H_1 \cdots H_j = \emptyset$ or $wH^{\geq j+1} \cap H^{\geq i+1} = \emptyset$.

Proof Let us assume that h is a TV-code. Then, clearly, h is regular and H is catenatively independent.

Let $w \in \Delta^+$ and $i, j \geq 1$ such that $w \notin H^{\geq i+1}$. To derive a contradiction, suppose that $H_1 \cdots H_i w \cap H_1 \cdots H_j \neq \emptyset$ and $wH^{\geq j+1} \cap H^{\geq i+1} \neq \emptyset$. Let $x \in H_1 \cdots H_i$ and $y \in H^{\geq j+1}$ such that $xw \in H_1 \cdots H_j$ and $wy \in H^{\geq i+1}$. Then, the word xwy has at least two decompositions over H, one of the form $x(wy)$, beginning by a decomposition of $x \in H_1 \cdots H_i$, and another one of the form $(xw)y$, beginning by a decomposition of $xw \in H_1 \cdots H_j$. The decomposition of x in the word xw is different than the decomposition of $x \in H_1 \cdots H_i$ because w is not a member of $H^{\geq i+1}$. Therefore, xwy has at least two distinct decompositions over H, contradicting the fact that h is a TV-code.

Conversely, suppose that (1), (2), and (3) hold true but h is not a TV-code. Then, there are two distinct words $\sigma_1 \cdots \sigma_n, \theta_1 \cdots \theta_m \in \Sigma^+$ such that

$$h(\sigma_1, 1) \cdots h(\sigma_n, n) = h(\theta_1, 1) \cdots h(\theta_m, m).$$

Let i be the least index such that $\sigma_i \neq \theta_i$. By (1), we obtain $h(\sigma_i, i) \neq h(\theta_i, i)$. Moreover, either $h(\sigma_i, i)$ is a proper prefix of $h(\theta_i, i)$, or vice versa. Let us suppose that $h(\theta_i, i) = h(\sigma_i, i)w$, where w is non-empty. Since H is catenatively independent, $w \notin H^{\geq i+1}$.

The equality

$$h(\theta_1, 1) \cdots h(\theta_i, i) = h(\sigma_1, 1) \cdots h(\sigma_i, i)w$$

shows that $H_1 \cdots H_i w \cap H_1 \cdots H_i \neq \emptyset$. Similarly, the equality

$$h(\sigma_{i+1}, i+1) \cdots h(\sigma_n, n) = wh(\theta_{i+1}, i+1) \cdots h(\theta_m, m)$$

shows that $wH^{\geq i+1} \cap H^{\geq i+1} \neq \emptyset$. We have now a contradiction with (2). $\square$

Remark 3.5 Let C be a nonempty subset of Δ^+. Corollary 3.1 and Theorem 3.1 lead to the following conclusion: C is a code iff the following two properties hold

1. C is catenatively independent;
2. for any $w \in \Delta^+$, if $w \notin C^+$ then $C^+w \cap C^+ = \emptyset$ or $wC^+ \cap C^+ = \emptyset$.

This is the Schützenberger criterion for codes [1].

3.3 Sardinas-Patterson Criterion for TV-Codes

We are going now to develope a Sardinas-Patterson characterization theorem for TV-codes. We shall analyse first all the cases where we can get the same encoding of two different input strings. The encodings will be viewed as two strings growing successively by catenating new code words.

Case 1: We start by growing both strings by different code words, as in Figure 3. Define the set $H_{1,1} = \{x \in \Delta^+ | \exists \sigma, \sigma' \in \Sigma : \ h(\sigma, 1)x = h(\sigma', 1)\}$. The word w in Figure 3 is a member of this set. Clearly, if $H_{1,1} \cap H_2 \neq \emptyset$ then h is not a TV-code because it is not catenatively independent.

Case 2: Starting with the configuration in Figure 3 we catenate a new code word to the first (second) string. Assume that we have the case in Figure 4. Define the set $H_{2,1} = \{x \in \Delta^+ | \exists \sigma \in \Sigma : \ h(\sigma, 2)x \in H_{1,1}\}$. The word w' in

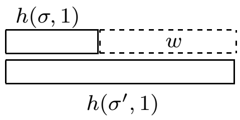

Figure 3: Generating two distinct code sequences: case 1

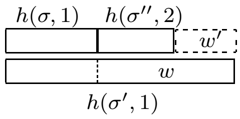

Figure 4: Generating two distinct code sequences: case 2

Figure 4 is a member of $H_{2,1}$. Clearly, if $H_{2,1} \cap H_3 \neq \emptyset$ then H is not a TV-code.

Case 3: Starting with the configuration in Figure 3 we catenate a new code word to the first (second) string. Assume that we have the case in Figure 5. Define the set $H_{1,2} = \{x \in \Delta^+ | \exists y \in H_{1,1} : \ yx \in H_2\}$. The word w' in Figure 5

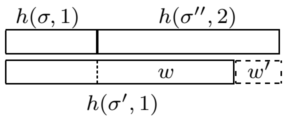

Figure 5: Generating two distinct code sequences: case 3

is a member of $H_{1,2}$. Clearly, if $H_{1,2} \cap H_2 \neq \emptyset$ then H is not a TV-code.

The discussion above leads us to consider, for any $k \geq 1$, the family of sets $\mathcal{H}_k = \{H_{i,j} | i, j \geq k\}$ defined as follows:

- $H_{k,k} = \{x \in \Delta^+ | H_k x \cap H_k \neq \emptyset\}$;
- $H_{i,j} = \{x \in \Delta^+ | (i > k \wedge H_i x \cap H_{i-1,j} \neq \emptyset) \vee (j > k \wedge H_{j-1,i} x \cap H_j \neq \emptyset)\}$, for all $i, j \geq k$, but at least one of them greater than k.

Example 3.1 Let h be the function given in the table below.

$\Sigma \setminus \mathbf{N}^*$	1	2	3	4	5	$\cdots$
σ_1	a	baa	ab	ab	ab	$\cdots$
σ_2	baa	b	aab	aab	aab	$\cdots$
σ_3	aba	bab	a	a	a	$\cdots$

Then, the following sets are members of the family $\mathcal{H}_1$:

– $H_{1,1} = \{ba\}$;

– $H_{2,1} = \{a\}$, $H_{1,2} = \{a, b\}$;

– $H_{3,1} = \emptyset$, $H_{2,2} = \{aa, ab\}$, $H_{1,3} = \{b, ab\}$;

– $H_{4,1} = \emptyset$, $H_{3,2} = \{a, b\}$, $H_{2,3} = \{b\}$, $H_{1,4} = \emptyset$.

Proposition 3.3 Let $h : \Sigma \times \mathbf{N}^* \to \Delta^+$ be a function. For any $k \geq 1$, any $i, j \geq k$, and any $x \in H_{i,j}$, there are $\sigma_1 \cdots \sigma_{i-k+1}, \sigma'_1 \cdots \sigma'_{j-k+1} \in \Sigma^+$ such that $\sigma_1 \neq \sigma'_1$ and

$$h(\sigma_1, k) \cdots h(\sigma_{i-k+1}, i)x = h(\sigma'_1, k) \cdots h(\sigma'_{j-k+1}, j).$$

Proof We prove the proposition by induction on $i + j$, where $i, j \geq k$.

Induction basis: Let $x \in H_{k,k}$. There are $\sigma_1, \sigma'_1 \in \Sigma$ such that $h(\sigma_1, k)x = h(\sigma'_1, k)$. Because x is non-empty, σ_1 and σ'_1 should be distinct.

Induction hypothesis: Assume the statement in proposition true for all $p, q \geq k$ such that $p + q = i + j$, where where $i, j \geq k$.

Induction step: Let $p, q \geq k$ such that $p + q = i + j + 1$, and let $x \in H^1_{p,q}$. We have to consider two cases.

Case 1 $H_p x \cap H_{p-1,q} \neq \emptyset$. Let $y \in H_p x \cap H_{p-1,q}$. Because $y \in H_{p-1,q}$, by the induction hypothesis there are $\sigma_1 \cdots \sigma_{p-k}, \sigma'_1 \cdots \sigma'_{q-k+1} \in \Sigma^+$ such that $\sigma_1 \neq \sigma'_1$ and

$$h(\sigma_1, k) \cdots h(\sigma_{p-k}, p-1)y = h(\sigma'_1, k) \cdots h(\sigma'_{q-k+1}, q).$$

Because $y \in H_p x$, there is $\sigma_{p-k+1} \in \Sigma$ such that $y = h(\sigma_{p-k+1}, p)x$. Combining the above results we have

$$h(\sigma_1, k) \cdots h(\sigma_{p-k}, p-1)h(\sigma_{p-k+1}, p)x = h(\sigma'_1, k) \cdots h(\sigma'_{q-k+1}, q)$$

which proves the proposition in this case.

Case 2 $H_{q-1,p} x \cap H_q \neq \emptyset$. Let $y \in H_{q-1,p} x \cap H_q$. Then, there is $y' \in H_{q-1,p}$ such that $y = y'x$. By the induction hypothesis, there are $\sigma'_1 \cdots \sigma'_{q-k}$ and $\sigma_1 \cdots \sigma_{p-k+1}$ in Σ^+ such that $\sigma'_1 \neq \sigma_1$ and

$$h(\sigma'_1, k) \cdots h(\sigma'_{q-k}, q-1)y' = h(\sigma_1, k) \cdots h(\sigma_{p-k+1}, p).$$

Further, we have

$$h(\sigma'_1, k) \cdots h(\sigma'_{q-k}, q-1)y'x = h(\sigma_1, k) \cdots h(\sigma_{p-k+1}, p)x.$$

Because $y \in H_q$, there is $\sigma'_{q-k+1}\Sigma$ such that $y = h(\sigma'_{q-k+1}, q)$. Therefore, we obtain

$$h(\sigma_1, k) \cdots h(\sigma_{p-k+1}, p)x = h(\sigma'_1, k) \cdots h(\sigma'_{q-k}, q-1)h(\sigma'_{q-k+1}, q),$$

which proves the proposition in this case. □

Now we can prove the following result.

Theorem 3.2 (Sardinas-Patterson criterion for TV-codes)
A function $h : \Sigma \times \mathbf{N}^* \to \Delta^+$ is a TV-code iff the following properties hold:

(1) h is regular;

(2) $(\forall k \geq 1)(\forall i, j \geq k)(H_{i,j} \cap H_{i+1} = \emptyset)$.

Proof Let us assume that h is a TV-code. Clearly, h is regular. In order to prove (2) consider $k \geq 1$ and $i, j \geq k$.

To derive a contradiction, assume that $H_{i,j} ap H_{i+1} \neq \emptyset$, and let $x \in H_{i,j} \cap H_{i+1}$. By Proposition 3.3, there are $\sigma_1 \cdots \sigma_{i-k+1}$ and $\sigma'_1 \cdots \sigma'_{j-k+1}$ in Σ^+ such that $\sigma_1 \neq \sigma'_1$ and

$$h(\sigma_1, k) \cdots h(\sigma_{i-k+1}, i)x = h(\sigma'_1, k) \cdots h(\sigma'_{j-k+1}, j).$$

Because $x \in H_{i+1}$, there is $\sigma \in \Sigma$ such that $x = h(\sigma, i+1)$. Let $\theta \in \Sigma$ be an arbitrary element. Then,

$$h(\theta, 1) \cdots h(\theta, k-1)h(\sigma_1, k) \cdots h(\sigma_{i-k+1}, i)h(\sigma, i+1) =$$

$$h(\theta, 1) \cdots h(\theta, k-1)h(\sigma'_1, k) \cdots h(\sigma'_{j-k+1}, j),$$

which shows that $\bar{h}$ is not injective because $\sigma_1 \neq \sigma'_1$ contradicting the fact that h is a TV-code.

Conversely, suppose that (1) and (2) hold true but h is not a TV-code. There are two distinct words $\sigma_1 \cdots \sigma_n, \sigma'_1 \cdots \sigma'_m \in \Sigma^+$ such that

$$(*) \quad \bar{h}(\sigma_1 \cdots \sigma_n) = \bar{h}(\sigma'_1 \cdots \sigma'_m).$$

Let k be the least number such that $\sigma_k \neq \sigma'_k$. Then, $(*)$ is equivalent to

$$(**) \quad h(\sigma_k, k) \cdots h(\sigma_n, n) = h(\sigma'_k, k) \cdots h(\sigma'_m, m).$$

Without loss of the generality we may assume that there are no $k \leq i < n$ and $k \leq j < m$ such that

$$h(\sigma_k, k) \cdots h(\sigma_i, i) = h(\sigma'_k, k) \cdots h(\sigma'_j, j).$$

From (1) it follows that $h(\sigma_k, k) \neq h(\sigma'_k, k)$, and from $(**)$ it follows that $h(\sigma_k, k)$ is a proper prefix of $h(\sigma'_k, k)$, or vice versa. Let us assume that $h(\sigma_k, k)$ is a prefix of $h(\sigma'_k, k)$, and let x be such that $h(\sigma_k, k)x = h(\sigma'_k, k)$. Then, $x \in H_{k,k}$, and $x \notin H_{k+1}$ because $K_{k,k} \cap H_{k+1} = \emptyset$.

The relation $(**)$ leads to

$$(***) \quad h(\sigma_{k+1}, k+1) \cdots h(\sigma_n, n) = xh(\sigma'_k, k) \cdots h(\sigma'_m, m).$$

Then, $h(\sigma_{k+1}, k+1)$ is a proper prefix of x, or vice versa. Combining this with the theorem's hypothesis, there is a word y such that $y \in H_{k+1,k} - H_{k+2}$, if the first case holds, or $y \in H_{k,k+1} - H_{k+1}$, if the second case holds. Both strings in $(***)$ can now be simplified to the left by $h(\sigma_{k+1}, k+1)$ resulting

$$h(\sigma_{k+2}, k+2) \cdots h(\sigma_n, n) = yh(\sigma'_k, k) \cdots h(\sigma'_m, m)$$

or

$$yh(\sigma_{k+2}, k+2)\cdots h(\sigma_n, n) = h(\sigma'_k, k)\cdots h(\sigma'_m, m).$$

Continuing this process a finite number of times, we get a word z such that $h(\sigma_n, n) = z$ and $z \in H_{n-1,m} - H_n$, or $z = h(\sigma'_m, m)$ and $z \in H_{m-1,n} - H_m$. Both cases lead to a contradiction. Therefore, h is a TV-code. □

Example 3.2 Consider the function h in Example 3.1. We have $H_{3,2} \cap H_4 \neq \emptyset$, which shows that h is not a TV-code. In fact, it is easy to see that

$$h(\sigma_1, 1)h(\sigma_2, 2)h(\sigma_1, 3)h(\sigma_1, 4) = h(\sigma_3, 1)h(\sigma_2, 2).$$

Remark 3.6 We show that the Sardinas-Patterson characterization theorem for classical codes is a special case of Theorem 3.2.

Let $C \subseteq \Delta^+$ be a non-empty set. Define

- $C_1 = \{x \in \Delta^+ | Cx \cap C \neq \emptyset\}$;
- $C_{i+1} = \{x \in \Delta^+ | Cx \cap C_i \neq \emptyset \ \wedge \ C_i x \cap C \neq \emptyset\}$, for all $i \geq 1$.

The Sardinas-Patterson characterization theorem states that C is a code iff $C \cap C_i = \emptyset$, for all $i \geq 1$.

Assume now that $h : \Sigma \times \mathbf{N}^* \to \Delta^+$ is a regular function of base C. Then, $H_{k,k} = H_{1,1} = C_1$, for all $k \geq 1$. By induction on $i \geq 2$, we can prove that

$$C_i = \bigcup_{j=1}^{i} H_{i-j+1,j}.$$

If we assume that $C_{i-1} = \bigcup_{j=1}^{i-1} H_{i-j+1,j}$, then

$$\begin{array}{rcl} C_i & = & \{x \in \Delta^+ | Cx \cap C_{i-1} \neq \emptyset \ \vee \ C_{i-1}x \cap C \neq \emptyset\} \\ & = & \{x \in \Delta^+ | Cx \cap H_{i-1,1} \neq \emptyset \ \vee \\ & & \qquad Cx \cap H_{i-2,2} \neq \emptyset \ \vee \ H_{1,i-1}x \cap C \neq \emptyset \ \vee \\ & & \qquad \cdots \\ & & \qquad Cx \cap H_{1,i-1} \neq \emptyset \ \vee \ H_{i-2,2}x \cap C \neq \emptyset \ \vee \\ & & \qquad Cx \cap H_{i-1,1} \neq \emptyset\} \\ & = & \bigcup_{j=1}^{i} H_{i-j+1,j} \end{array}$$

Since h is regular and $H_{k,k} = H_{1,1} = C_1$, for all $k \geq 1$, the condition

$$(\forall k \geq 1)(\forall i, j \geq 1)(H_{i,j} \cap H_{i+1} = \emptyset)$$

is equivalent to

$$(\forall i \geq 1)(C_i \cap C = \emptyset).$$

Therefore, under the assumption of regularity, Theorem 3.2 is the Sardinas-Patterson characterization theorem for codes.

Conclusions

Time-varying codes associate variable length code words to letters being encoded depending on their positions in the input string. These codes have been introduced in [8] as a proper extension of L-codes.

In this paper we have continued the study of time-varying codes. First, we have shown that adaptive Huffman encodings are special cases of encodings by time-varying codes. Then, we have provided three kinds of characterization results: characterization results based on decompositions over families of sets of words, a Schützenberger like criterion, and a Sardinas-Patterson like characterization theorem. All of them extend the corresponding characterization results known for classical variable length codes.

References

[1] J. Berstel, D. Perrin: *Theory of Codes*, Academic Press, 1985.

[2] W. Kuich, A. Salomaa. *Semirings, Automata, and Languages*, Springer-Verlag, 1986.

[3] G. Lallement. *Semigroups and Combinatorial Applications*, John-Wiley & Sons, 1979.

[4] H.A. Maurer, A. Salomaa, D. Wood. *L-Codes and Number Systems*, Theoretical Computer Science 22, 1983, 331–346.

[5] G. Rozenberg, A. Salomaa (eds.): *Handbook of Formal Languages*, vol. 1, Springer-Verlag, 1997.

[6] A. Salomaa: *Jewels of Formal Language Theory*, Computer Science Press, 1981.

[7] D. Salomon. *Data Compression. The Complete Reference*, Springer-Verlag, 1998.

[8] F.L. Ţiplea, E. Mäkinen, C. Enea. *SE-Systems, Timing Mechanisms, and Time-Varying Codes*, International Journal of Computer Mathematics (to appear).